**WEEK 6 – ASSIGNMENT**

**EXERCISE 1:**

**Step-by-Step Implementation**

**Installed Node.js and NPM from:**

<https://nodejs.org/en/download/>

**Installed create-react-app globally using:**

Npm install -g create-react-app

**Created a new React app named myfirstreact:**

Npx create-react-app myfirstreact

**Navigated into the project directory:**

Cd myfirstreact

Opened the project in Visual Studio Code and modified App.js as follows:

**App.js Content:**

**Jsx**

Import React from ‘react’;

Function App() {

Return (

<div>

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

</div>

);

}

Export default App;

**Ran the React application:**

Npm start

**Opened browser and visited:**

<http://localhost:3000>

The heading “Welcome to the first session of React” displayed successfully.

**EXERCISE 2:**

**Step-by-Step Implementation**

**Create React App**

Open Visual Studio Code terminal and run:

Npx create-react-app StudentApp

**Cd StudentApp**

Project Structure

**Pgsql**

StudentApp/

├── node\_modules/

├── public/

├── src/

│ ├── Components/

│ │ ├── Home.js

│ │ ├── About.js

│ │ └── Contact.js

│ ├── App.js

│ └── index.js

├── package.json

└── README.md

**1. Home Component**

**src/Components/Home.js**

**Jsx**

Import React from ‘react’;

Function Home() {

Return (

<div>

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

</div>

);

}

Export default Home;

**2. About Component**

**src/Components/About.js**

**Jsx**

Import React from ‘react’;

Function About() {

Return (

<div>

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

</div>

);

}

Export default About;

**3. Contact Component**

**src/Components/Contact.js**

**Jsx**

Import React from ‘react’;

Function Contact() {

Return (

<div>

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

</div>

);

}

Export default Contact;

**4. Modify App.js to Render All Components**

**src/App.js**

**Jsx**

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”>

<Home />

<About />

<Contact />

</div>

);

}

Export default App;

**Run the Application In the terminal:**

Npm start

In your browser:

Visit <http://localhost:3000>

**Output :**

**Css**

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

**EXERCISE 3:**

**Step-by-Step Implementation**

**Step 1:** Create React App

Open VS Code terminal and run:

Npx create-react-app scorecalculatorapp

Cd scorecalculatorapp

**Step 2:** Create Components Folder and File

Create a folder:

/src/Components/

Add a file named:

CalculateScore.js

**Step 3:** Create the CalculateScore Functional Component

**src/Components/CalculateScore.js**

**Jsx**

Import React from ‘react’;

Import ‘../Stylesheets/mystyle.css’;

Function CalculateScore() {

Const student = {

Name: ‘Alice’,

School: ‘Springfield High School’,

Total: 450,

Goal: 500

};

Const average = (student.total / student.goal) \* 100;

Return (

<div className=”score-container”>

<h2>Student Score Report</h2>

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

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

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

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

<p><strong>Average Score:</strong> {average.toFixed(2)}%</p>

</div>

);

}

Export default CalculateScore;

**Step 4:** Create Stylesheet

Create a folder:

/src/Stylesheets/

Add a file named:

**Mystyle.css**

**src/Stylesheets/mystyle.css**

.score-container {

Background-color: #f0f8ff;

Padding: 20px;

Border: 2px solid #0077cc;

Border-radius: 10px;

Margin: 20px auto;

Width: 400px;

Font-family: ‘Segoe UI’, Tahoma, Geneva, Verdana, sans-serif;

Box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

}

.score-container h2 {

Color: #0077cc;

Text-align: center;

}

.score-container p {

Font-size: 16px;

Line-height: 1.6;

}

**Step 5:** Edit App.js to Render Component

**src/App.js**

**Jsx**

Import React from ‘react’;

Import ‘./App.css’;

Import CalculateScore from ‘./Components/CalculateScore’;

Function App() {

Return (

<div className=”App”>

<CalculateScore />

</div>

);

}

Export default App;

**Step 6:** Run the Application

In the terminal, inside the project folder, run:

Npm start

Open browser and go to:

<http://localhost:3000>

**Output :**

Student Score Report

Name: Alice

School: Springfield High School

Total Marks: 450

Goal: 500

Average Score: 90.00%

**EXERCISE 4:**

**Step-by-Step Implementation**

**1. Create React App**

Open VS Code terminal and run:

Npx create-react-app blogapp

Cd blogapp

**2. Create Post.js**

**src/Post.js**

**Jsx**

Import React from ‘react’;

Function Post({ title, body }) {

Return (

<div style={{ border: ‘1px solid #ddd’, padding: ‘10px’, marginBottom: ‘10px’ }}>

<h3>{title}</h3>

<p>{body}</p>

</div>

);

}

Export default Post;

**3. Create Class Component Posts.js**

**src/Posts.js**

**Jsx**

Import React, { Component } from ‘react’;

Import Post from ‘./Post’;

Class Posts extends Component {

Constructor(props) {

Super(props);

This.state = {

Posts: [],

Error: null

};

}

// Lifecycle hook to fetch data

componentDidMount() {

this.loadPosts();

}

// Method to fetch posts from API

loadPosts() {

fetch(‘https://jsonplaceholder.typicode.com/posts’)

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

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

.catch(error => this.setState({ error }));

}

// Lifecycle hook to catch rendering errors

componentDidCatch(error, info) {

alert(‘Something went wrong while rendering posts.’);

console.error(“Error:”, error);

console.error(“Info:”, info);

}

Render() {

Const { posts, error } = this.state;

If (error) {

Return <p>Error loading posts.</p>;

}

Return (

<div>

<h2>Blog Posts</h2>

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

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

))}

</div>

);

}

}

Export default Posts;

**4. Edit App.js to Use the Posts Component**

**src/App.js**

**Jsx**

Import React from ‘react’;

Import ‘./App.css’;

Import Posts from ‘./Posts’;

Function App() {

Return (

<div className=”App”>

<h1>Welcome to BlogApp</h1>

<Posts />

</div>

);

}

Export default App;

**5. Run the Application**

Npm start

Open browser and visit:

<http://localhost:3000>

**Output :**

Welcome to BlogApp

Blog Posts

Post Title 1

Post Body 1

Post Title 2

Post Body 2

...

**EXERCISE 5:**

**Step-by-Step Implementation**

**Step 1:** Setup the Project

Unzip the provided React app to a folder (e.g., CohortDashboard).

Open a terminal (Command Prompt or VS Code Terminal).

Navigate to the project directory:

Cd CohortDashboard

**Restore project dependencies:**

Npm install

**Step 2:** Create CSS Module File

Create a file under the same folder as the component:

**src/CohortDetails.module.css**

**Css**

.box {

Width: 300px;

Display: inline-block;

Margin: 10px;

Padding: 10px 20px;

Border: 1px solid black;

Border-radius: 10px;

}

Dt {

Font-weight: 500;

}

**Step 3:** Modify the Component — CohortDetails.js

**src/CohortDetails.js**

**Jsx**

Import React from ‘react’;

Import styles from ‘./CohortDetails.module.css’;

Function CohortDetails({ cohort }) {

Const headerStyle = {

Color: cohort.status === ‘ongoing’ ? ‘green’ : ‘blue’

};

Return (

<div className={styles.box}>

<h3 style={headerStyle}>{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>

);

}

Export default CohortDetails;

**Step 4:** Sample Usage in App.js (Optional Testing)

**src/App.js**

**Jsx**

Import React from ‘react’;

Import CohortDetails from ‘./CohortDetails’;

Function App() {

Const cohorts = [

{ name: “React Bootcamp”, status: “ongoing”, startDate: “2025-07-01”, endDate: “2025-08-01” },

{ name: “Java Sprint”, status: “completed”, startDate: “2025-05-01”, endDate: “2025-06-01” }

];

Return (

<div>

<h1>My Academy Dashboard</h1>

{cohorts.map((cohort, index) => (

<CohortDetails key={index} cohort={cohort} />

))}

</div>

);

}

Export default App;

**Output :**

Each cohort will be displayed in a card-like box:

Ongoing cohorts → Green heading

Completed cohorts → Blue heading

Titles (<dt>) will appear bold

Each cohort detail box will have proper padding, border, and spacing

**EXERCISE 6:**

**Step-by-Step Implementation**

**Step 1:** Create the React App

Open terminal and run:

Npx create-react-app TrainersApp

Cd TrainersApp

Code .

**Step 2:** Install React Router DOM

Install routing library:

Npm install react-router-dom

**Step 3:** Define the Trainer Class

**src/trainer.js**

**Jsx**

Class Trainer {

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

This.trainerId = trainerId;

This.name = name;

This.phone = phone;

This.email = email;

This.technology = technology;

This.skills = skills;

}

}

Export default Trainer;

**Step 4:** Create Mock Trainer Data

**src/TrainersMock.js**

**Jsx**

Import Trainer from ‘./trainer’;

Const TrainersData = [

New Trainer(1, “Alice Johnson”, “1234567890”, [alice@example.com](mailto:alice@example.com), “React”, [“JS”, “React”, “Redux”]),

New Trainer(2, “Bob Smith”, “9876543210”, [bob@example.com](mailto:bob@example.com), “Java”, [“Java”, “Spring”, “Hibernate”]),

New Trainer(3, “Charlie Brown”, “5551234567”, [charlie@example.com](mailto:charlie@example.com), “Python”, [“Python”, “Django”, “Flask”]),

];

Export default TrainersData;

**Step 5:** Create Home.js Component

**src/Home.js**

**Jsx**

Import React from ‘react’;

Function Home() {

Return (

<div>

<h2>Welcome to Cognizant Academy</h2>

<p>Manage your trainer profiles and skills with this SPA.</p>

</div>

);

}

Export default Home;

**Step 6:** Create TrainerList.js Component

**src/TrainerList.js**

**Jsx**

Import React from ‘react’;

Import { Link } from ‘react-router-dom’;

Import TrainersData from ‘./TrainersMock’;

Function TrainerList() {

Return (

<div>

<h2>Trainers List</h2>

<ul>

{TrainersData.map(trainer => (

<li key={trainer.trainerId}>

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

</li>

))}

</ul>

</div>

);

}

Export default TrainerList;

**Step 7:** Create TrainerDetails.js Component

**src/TrainerDetails.js**

**Jsx**

Import React from ‘react’;

Import { useParams } from ‘react-router-dom’;

Import TrainersData from ‘./TrainersMock’;

Function TrainerDetails() {

Const { id } = useParams();

Const trainer = TrainersData.find(t => t.trainerId === parseInt(id));

If (!trainer) return <p>Trainer not found</p>;

Return (

<div>

<h2>Trainer Details</h2>

<p><strong>ID:</strong> {trainer.trainerId}</p>

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

<p><strong>Email:</strong> {trainer.email}</p>

<p><strong>Phone:</strong> {trainer.phone}</p>

<p><strong>Technology:</strong> {trainer.technology}</p>

<p><strong>Skills:</strong> {trainer.skills.join(‘, ‘)}</p>

</div>

);

}

Export default TrainerDetails;

**Step 8:** Update App.js with Routing Logic

**src/App.js**

**Jsx**

Import React from ‘react’;

Import { BrowserRouter, Routes, Route, Link } from ‘react-router-dom’;

Import Home from ‘./Home’;

Import TrainerList from ‘./TrainerList’;

Import TrainerDetails from ‘./TrainerDetails’;

Function App() {

Return (

<BrowserRouter>

<div>

<h1>Trainers Management Portal</h1>

<nav>

<Link to=”/”>Home</Link> | <Link to=”/trainers”>Trainers</Link>

</nav>

<Routes>

<Route path=”/” element={<Home />} />

<Route path=”/trainers” element={<TrainerList />} />

<Route path=”/trainer/:id” element={<TrainerDetails />} />

</Routes>

</div>

</BrowserRouter>

);

}

Export default App;

**Step 9:** Run the Application

Npm start

Open the browser at:

<http://localhost:3000>

**Output :**

Welcome to Cognizant Academy

Manage your trainer profiles and skills with this SPA.

**Trainers List (/trainers)**

Trainers List

- Alice Johnson (Clickable)

- Bob Smith (Clickable)

- Charlie Brown (Clickable)

Trainer Detail (/trainer/1, /trainer/2, etc.)

**Trainer Details**

ID: 1

Name: Alice Johnson

Email: [alice@example.com](mailto:alice@example.com)

Phone: 1234567890

Technology: React

Skills: JS, React, Redux

**EXERCISE 7:**

**Step-by-Step Implementation**

**Step 1:** Create React App

Open the terminal and run:

Npx create-react-app shoppingapp

Cd shoppingapp

**Step 2: Project Structure**

**Css**

Shoppingapp/

├── src/

│ ├── OnlineShopping.js

│ ├── Cart.js

│ └── index.js

**Step 3:** Create the Cart Component (Class + Props)

**src/Cart.js**

**Jsx**

Import React from ‘react’;

Class Cart extends React.Component {

Render() {

Return (

<div style={styles.card}>

<h3>Item: {this.props.itemname}</h3>

<p>Price: ₹{this.props.price}</p>

</div>

);

}

}

// Default Props

Cart.defaultProps = {

Itemname: ‘Unknown Item’,

Price: 0

};

Const styles = {

Card: {

Border: ‘1px solid #ccc’,

Padding: ‘15px’,

borderRadius: ‘10px’,

margin: ‘10px’,

backgroundColor: ‘#f9f9f9’

}

};

Export default Cart;

**Step 4:** Create the OnlineShopping Component

**src/OnlineShopping.js**

**Jsx**

Import React from ‘react’;

Import Cart from ‘./Cart’;

Class OnlineShopping extends React.Component {

Render() {

Const items = [

{ itemname: “Laptop”, price: 50000 },

{ itemname: “Smartphone”, price: 25000 },

{ itemname: “Headphones”, price: 3000 },

{ itemname: “Keyboard”, price: 1500 },

{ itemname: “Monitor”, price: 12000 }

];

Return (

<div>

<h2>🛒 Welcome to Online Shopping Portal</h2>

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

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

))}

</div>

);

}

}

Export default OnlineShopping;

**Step 5:** Update index.js to Render Main Component

**src/index.js**

**Jsx**

Import React from ‘react’;

Import ReactDOM from ‘react-dom/client’;

Import OnlineShopping from ‘./OnlineShopping’;

Const root = ReactDOM.createRoot(document.getElementById(‘root’));

Root.render(<OnlineShopping />);

ReactDOM.render() is used to mount the OnlineShopping component into the root DOM node, making the app functional in the browser.

**Step 6:** Run the Application

Npm start

Visit in browser:

<http://localhost:3000>

**Output :**

Welcome to Online Shopping Portal

Item: Laptop

Price: ₹50000

Item: Smartphone

Price: ₹25000

Item: Headphones

Price: ₹3000

Item: Keyboard

Price: ₹1500

Item: Monitor

Price: ₹12000

**EXERCISE 8:**

**Step-by-Step Implementation**

**Step 1:** Create React App

Open the terminal and run:

Npx create-react-app counterapp

Cd counterapp

Code .

**Step 2:** Create CountPeople.js Component

**src/CountPeople.js**

**Jsx**

Import React from ‘react’;

Class CountPeople extends React.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 style={styles.container}>

<h2>Mall Entry Management</h2>

<p><strong>People Entered:</strong> {this.state.entryCount}</p>

<p><strong>People Exited:</strong> {this.state.exitCount}</p>

<button onClick={this.updateEntry} style={styles.button}>Login</button>

<button onClick={this.updateExit} style={styles.button}>Exit</button>

</div>

);

}

}

Const styles = {

Container: {

textAlign: ‘center’,

marginTop: ‘50px’,

fontFamily: ‘Arial, sans-serif’

},

Button: {

Margin: ‘10px’,

Padding: ‘10px 20px’,

fontSize: ‘16px’

}

};

Export default CountPeople;

**Step 3:** Update App.js to Use the Component

**src/App.js**

**Jsx**

Import React from ‘react’;

Import ‘./App.css’;

Import CountPeople from ‘./CountPeople’;

Function App() {

Return (

<div className=”App”>

<CountPeople />

</div>

);

}

Export default App;

**Step 4:** Run the Application

Npm start

Visit your browser:

<http://localhost:3000>

**Output :**

Mall Entry Management:

People Entered: 3

People Exited: 1

[Login] [Exit]