6. ReactJS-HOL

**Objectives**

* **Explain the need and benefits of React Router**
  + React Router enables navigation between views without reloading the entire page, giving users a seamless single-page experience.
  + It helps manage complex navigation structures, enhances app performance, and allows bookmarking and deep linking.
* **Identify Components in React Router**
  + BrowserRouter: Uses the HTML5 history API for clean URLs.
  + Routes: Contains all the Route definitions.
  + Route: Maps a URL path to a specific component.
  + Link: Enables navigation without page reload.
  + useParams: React hook to extract URL parameters.
* **List the types of Router Components**
  + BrowserRouter: For traditional web apps with real URLs.
  + HashRouter: Uses URL hash for routing (e.g., /#/path).
  + MemoryRouter: Keeps routing history in memory (useful for testing).
* **Understand parameter passing via URL**
  + Parameters can be passed via the URL path (e.g., /trainer/:id).
  + useParams() hook is used to access these dynamic segments inside the target component.

**Hands-on Lab**

1**. Create React App:**  
  
npx create-react-app TrainersApp  
cd TrainersApp  
code .

2. **Create Trainers.js:**  
  
export default 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;  
 }  
}

3. **Create TrainersMock.js:**  
  
const trainers = [...]; // Sample trainer data as discussed

4. Install React Router:  
  
npm install react-router-dom

5. **Create TrainerList.js:**  
  
import React from 'react';  
import { Link } from 'react-router-dom';  
  
function TrainerList({ data }) {  
 return (  
 <div>  
 <h2>Trainer List</h2>  
 <ul>  
 {data.map(trainer => (  
 <li key={trainer.trainerId}>  
 <Link to={`/trainer/${trainer.trainerId}`}>{trainer.name}</Link>  
 </li>  
 ))}  
 </ul>  
 </div>  
 );  
}  
  
export default TrainerList;

6**. Create Home.js:**  
  
function Home() {  
 return (  
 <div>  
 <h2>Welcome to TrainersApp</h2>  
 <p>Browse the list of trainers and view details.</p>  
 </div>  
 );  
}

7**. Update App.js** with Routing using BrowserRouter, Routes, Route, Link

8. **Create TrainerDetail.js** using useParams to fetch ID from URL and display trainer data

9**. Run the App:**  
  
npm start

# Output

