Chapter 10: React Router

1 Introduction

Is chapter mein humne **React Router** ke baare mein seekha, jo React apps mein client-side routing ke liye use hoti hai. Yeh multiple pages aur navigation banane mein help karti hai without page refresh. Humne `BrowserRouter`, `Routes`, `Route`, aur `Link` ka use dekha.

2 10.1 What is React Router?

React Router ek library hai jo single-page applications (SPAs) mein client-side routing enable karti hai. Yani, user page ke beech navigate karta hai toh browser pura refresh nahi hota, sirf components change hote hain.

2.1 Real-World Example

Food delivery app mein pages jaise Home (`/`), Menu (`/menu`), aur Cart (`/cart`) banane ke liye React Router use hota hai.

3 10.2 Setting Up React Router

React Router ke liye `react-router-dom` use karte hain. Key components:

- `BrowserRouter`: App ko routing ke liye wrap karta hai.

- `Routes`: Sabhi routes define karta hai.

- `Route`: Path aur component ko map karta hai.

- `Link`: Navigation links without refresh.

3.1 Code Example: Basic Routing

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Basic Routing</title>
6   <script
       src="https://cdn.jsdelivr.net/npm/react@18.2.0/umd/react.development.js
7   <script
       src="https://cdn.jsdelivr.net/npm/react-dom@18.2.0/umd/react-dom.develop
8   <script
       src="https://cdn.jsdelivr.net/npm/@babel/standalone@7.20.6/babel.min.js
9   <script
       src="https://cdn.jsdelivr.net/npm/react-router-dom@6.3.0/dist/umd/react-
10   <script src="https://cdn.tailwindcss.com"></script>
11 </head>
12 <body>
13   <div id="root"></div>
```

```
14  <script type="text/babel">
15    const { BrowserRouter, Routes, Route, Link } =
        ReactRouterDOM;
16
17    // Home Component
18    function Home() {
19      return (
20        <div className="text-center p-4">
21          <h1 className="text-3xl font-bold
              text-blue-600">Welcome to Food App</h1>
22          <p className="text-lg">Explore our delicious menu!</p>
23        </div>
24      );
25    }
26
27    // Menu Component
28    function Menu() {
29      return (
30        <div className="text-center p-4">
31          <h1 className="text-3xl font-bold
              text-blue-600">Menu</h1>
32          <ul className="list-disc list-inside">
33            <li>Chai - Rs 10</li>
34            <li>Samosa - Rs 20</li>
35            <li>Idli - Rs 30</li>
36          </ul>
37        </div>
38      );
39    }
40
41    // Cart Component
42    function Cart() {
43      return (
44        <div className="text-center p-4">
45          <h1 className="text-3xl font-bold text-blue-600">Your
              Cart</h1>
46          <p className="text-lg">Cart is empty</p>
47        </div>
48      );
49    }
50
51    // Main App Component
52    function App() {
53      return (
54        <BrowserRouter>
55          <div className="p-4">
56            <nav className="flex justify-center space-x-4">
57              <Link to="/" className="text-blue-500
                  hover:underline">Home</Link>
58              <Link to="/menu" className="text-blue-500
                  hover:underline">Menu</Link>
```

```
59          <Link to="/cart" className="text-blue-500
                hover:underline">Cart</Link>
60        </nav>
61        <Routes>
62          <Route path="/" element={<Home />} />
63          <Route path="/menu" element={<Menu />} />
64          <Route path="/cart" element={<Cart />} />
65        </Routes>
66      </div>
67    </BrowserRouter>
68  );
69 }
70
71 const root =
       ReactDOM.createRoot(document.getElementById('root'));
72 root.render(<App />);
73 </script>
74 </body>
75 </html>
```

3.2 Output

Browser mein yeh dikhega:

- **Top Navigation Bar**: "Home", "Menu", "Cart" links (blue text, hover pe underline).

- **URL: /**
  - Heading: "Welcome to Food App" (large, bold, blue).
  - Text: "Explore our delicious menu!" (smaller text).

- **URL: /menu**
  - Heading: "Menu" (large, bold, blue).
  - Bulleted list: "Chai - Rs 10", "Samosa - Rs 20", "Idli - Rs 30".

- **URL: /cart**
  - Heading: "Your Cart" (large, bold, blue).
  - Text: "Cart is empty".

- Links click karne pe page refresh nahi hota, sirf content change hota hai.

3.3 Explanation

- `BrowserRouter`: App ko routing ke liye wrap karta hai.

- `Link`: Navigation links (`/`, `/menu`, `/cart`).

- `Routes` aur `Route`: URL paths ko components se map karta hai.

4 10.3 Dynamic Routing with URL Parameters

Dynamic routes banane ke liye URL parameters (jaise `:id`) aur `useParams` hook use hota hai.

4.1 Code Example: Dynamic Item Details

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Dynamic Routing</title>
  <script
      src="https://cdn.jsdelivr.net/npm/react@18.2.0/umd/react.development.js
  <script
      src="https://cdn.jsdelivr.net/npm/react-dom@18.2.0/umd/react-dom.develop
  <script
      src="https://cdn.jsdelivr.net/npm/@babel/standalone@7.20.6/babel.min.js
  <script
      src="https://cdn.jsdelivr.net/npm/react-router-dom@6.3.0/dist/umd/react-
  <script src="https://cdn.tailwindcss.com"></script>
</head>
<body>
  <div id="root"></div>
  <script type="text/babel">
    const { BrowserRouter, Routes, Route, Link, useParams } =
        ReactRouterDOM;

    // Menu Component
    function Menu() {
      const menuItems = [
        { id: 1, name: "Chai", price: 10 },
        { id: 2, name: "Samosa", price: 20 },
        { id: 3, name: "Idli", price: 30 }
      ];

      return (
        <div className="text-center p-4">
          <h1 className="text-3xl font-bold
              text-blue-600">Menu</h1>
          <ul className="list-disc list-inside">
            {menuItems.map((item) => (
              <li key={item.id}>
                <Link to={`/item/${item.id}`}
                    className="text-blue-500 hover:underline">
                  {item.name} - Rs {item.price}
                </Link>
              </li>
            ))}
          </ul>
        </div>
```

```
38          );
39        }
40
41        // Item Detail Component
42        function ItemDetail() {
43          const { id } = useParams();
44          const menuItems = [
45            { id: 1, name: "Chai", price: 10, description: "Hot
                  spiced tea" },
46            { id: 2, name: "Samosa", price: 20, description: "Crispy
                  fried snack" },
47            { id: 3, name: "Idli", price: 30, description: "Steamed
                  rice cake" }
48          ];
49          const item = menuItems.find((item) => item.id ===
                parseInt(id));
50
51          return (
52            <div className="text-center p-4">
53              <h1 className="text-3xl font-bold text-blue-600">{item
                    ? item.name : 'Item Not Found'}</h1>
54              {item && (
55                <div>
56                  <p className="text-lg">Price: Rs {item.price}</p>
57                  <p className="text-lg">Description:
                      {item.description}</p>
58                  <Link to="/menu" className="text-blue-500
                      hover:underline">Back to Menu</Link>
59                </div>
60              )}
61            </div>
62          );
63        }
64
65        // Main App Component
66        function App() {
67          return (
68            <BrowserRouter>
69              <div className="p-4">
70                <nav className="flex justify-center space-x-4">
71                  <Link to="/" className="text-blue-500
                      hover:underline">Home</Link>
72                  <Link to="/menu" className="text-blue-500
                      hover:underline">Menu</Link>
73                </nav>
74                <Routes>
75                  <Route path="/" element={<h1
                      className="text-center text-3xl font-bold
                      text-blue-600 p-4">Home</h1>} />
76                  <Route path="/menu" element={<Menu />} />
77                  <Route path="/item/:id" element={<ItemDetail />} />
```

```
78          </Routes>
79        </div>
80      </BrowserRouter>
81    );
82  }
83
84  const root =
        ReactDOM.createRoot(document.getElementById('root'));
85  root.render(<App />);
86  </script>
87 </body>
88 </html>
```

4.2 Output

Browser mein yeh dikhega:

- **Top Navigation Bar**: "Home", "Menu" links (blue text, hover pe underline).
- **URL: /**
  - Heading: "Home" (large, bold, blue).
- **URL: /menu**
  - Heading: "Menu" (large, bold, blue).
  - Bulleted list: "Chai - Rs 10", "Samosa - Rs 20", "Idli - Rs 30" (sab clickable links).
- **URL: /item/1**
  - Heading: "Chai" (large, bold, blue).
  - Text: "Price: Rs 10", "Description: Hot spiced tea".
  - Link: "Back to Menu" (blue, hover pe underline).
- **URL: /item/2**
  - Heading: "Samosa", Price: Rs 20, Description: "Crispy fried snack".
- Links click karne pe content instantly change hota hai.

4.3 Explanation

- `useParams`: URL parameter (`:id`) ko access karta hai.
- `Link to="/item/$item.id"`: Dynamic URLs banate hain.
- `ItemDetail`: Item ke details dikhata hai.

5 10.4 Combining with Previous Concepts

React Router ko `useState`, forms, aur lifting state up ke saath combine kar sakte hain.

5.1 Code Example: Cart with Navigation

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Cart with Navigation</title>
  <script
    src="https://cdn.jsdelivr.net/npm/react@18.2.0/umd/react.development.js
  <script
    src="https://cdn.jsdelivr.net/npm/react-dom@18.2.0/umd/react-dom.develop
  <script
    src="https://cdn.jsdelivr.net/npm/@babel/standalone@7.20.6/babel.min.js
  <script
    src="https://cdn.jsdelivr.net/npm/react-router-dom@6.3.0/dist/umd/react
  <script src="https://cdn.tailwindcss.com"></script>
</head>
<body>
  <div id="root"></div>
  <script type="text/babel">
    const { BrowserRouter, Routes, Route, Link } =
      ReactRouterDOM;

    // Cart Component
    function Cart() {
      const [items, setItems] = React.useState([
        { id: 1, name: "Chai", price: 10 }
      ]);

      const addItem = () => {
        const newItem = {
          id: items.length + 1,
          name: `Item ${items.length + 1}`,
          price: 10 * (items.length + 1)
        };
        setItems([...items, newItem]);
      };

      return (
        <div className="text-center p-4">
          <h1 className="text-3xl font-bold text-blue-600">Your
            Cart</h1>
          <button
            className="bg-green-500 text-white px-4 py-2 m-2
              rounded"
            onClick={addItem}
          >
            Add Item
          </button>
          <ul className="list-disc list-inside">
            {items.map((item) => (
              <li key={item.id} className="text-lg">
```

```
44              {item.name} - Rs {item.price}
45            </li>
46          ))}
47        </ul>
48        <Link to="/" className="text-blue-500
             hover:underline">Back to Home</Link>
49      </div>
50    );
51  }
52
53  // Main App Component
54  function App() {
55    return (
56      <BrowserRouter>
57        <div className="p-4">
58          <nav className="flex justify-center space-x-4">
59            <Link to="/" className="text-blue-500
               hover:underline">Home</Link>
60            <Link to="/cart" className="text-blue-500
               hover:underline">Cart</Link>
61          </nav>
62          <Routes>
63            <Route path="/" element={<h1
               className="text-center text-3xl font-bold
               text-blue-600 p-4">Home</h1>} />
64            <Route path="/cart" element={<Cart />} />
65          </Routes>
66        </div>
67      </BrowserRouter>
68    );
69  }
70
71  const root =
72      ReactDOM.createRoot(document.getElementById('root'));
72  root.render(<App />);
73  </script>
74  </body>
75  </html>
```

5.2   Output

Browser mein yeh dikhega:

- **Top Navigation Bar**: "Home", "Cart" links (blue text, hover pe underline).
- **URL: /**
    - Heading: "Home" (large, bold, blue).
- **URL: /cart**
    - Heading: "Your Cart" (large, bold, blue).

- Button: "Add Item" (green button).

- Bulleted list: Initially "Chai - Rs 10".

- Add Item click pe: New items add hote hain (Item 2 - Rs 20, Item 3 - Rs 30, etc.).

- Link: "Back to Home" (blue, hover pe underline).

- Navigation seamless hai, no page refresh.

5.3 Explanation

- `Cart`: `useState` se items manage karta hai.

- `Link`: Home aur Cart ke beech navigation.

- `Routes`: URL paths ko components se map karta hai.

6 Common Mistakes

- **Missing `BrowserRouter`**: App ko wrap karna zaroori hai.

- **Incorrect Path Syntax**: `path="/menu"` exact match hona chahiye.

- **Using `<a>` Instead of `Link`**: `<a>` page refresh karta hai, `Link` nahi.

7 Interview Tips

- **React Router kya hai?**
  - Library jo client-side routing handle karti hai.

- **`Link` vs anchor tag?**
  - `Link` SPA navigation ke liye, no refresh. `<a>` refresh karta hai.

- **Dynamic routing kaise karte hain?**
  - URL parameters (`:id`) aur `useParams` hook se.

8 Assignment: Practice Time

8.1 Task 1: Basic Navigation

- `FoodApp` banao jisme `Home`, `Menu`, aur `About` pages ho.

- `Link` se navigation aur `Routes` se paths define karo.

- Har page mein heading aur paragraph.

- Tailwind se style karo.

8.2   Task 2: Dynamic Menu Details

- `MenuApp` banao jisme `/menu` pe items list ho.

- Har item `Link` ho jo `/item/:id` pe jaye.

- `ItemDetail` mein name, price, description dikhao.

- Tailwind se style karo.

8.3   Task 3: Cart Navigation with State

- `CartApp` banao jisme `/cart` pe dynamic cart items ho (`useState`).

- Button se items add karo, list render karo.

- `/home` aur `/cart` ke beech `Link` se navigation.

- Tailwind se style karo.