

Introduktion till React Router

Vad är React Router?

- React Router är ett bibliotek för att hantera routing i React-applikationer.
- Det låter dig skapa en SPA (Single Page Application) där du kan navigera mellan olika "sidor" utan att ladda om hela sidan.
- Med React Router kan du definiera vilka komponenter som ska visas beroende på URL:en.

Vad används React Router till?

- **Navigering:** Skapa länkar mellan olika vyer i din applikation.
- **Dynamisk rendering:** Visa olika komponenter baserat på URL-parametrar.
- **Historikhantering:** Tillåter användare att navigera bakåt och framåt med webbläsarens historik.
- **Programmatisk navigering:** Med hooks som `useNavigate` kan du styra navigationen via kod.

Grundläggande koncept i React Router

1. BrowserRouter:

- Omsluter hela din applikation och tillhandahåller routing-kontext.

2. Routes & Route:

- Används för att definiera vilka komponenter som ska renderas för specifika URL-vägar.

3. Link:

- En komponent som skapar navigationslänkar utan att ladda om sidan.

4. useNavigate:

- En hook för att navigera programatiskt (t.ex. efter en händelse).

Installation av React Router

- I terminalen, kör:

```
npm install react-router-dom
```

- Detta lägger till React Router i ditt projekt så att du kan importera och använda dess komponenter och hooks.

Använda BrowserRouter

- **BrowserRouter** är den komponent som du använder för att omsluta din applikation.
- Den skapar en kontext som låter andra routing-komponenter (Routes, Route, Link, etc.) fungera korrekt.

```
// Exempel: Omsluta din applikation
import { BrowserRouter } from "react-router-dom";

const App = () => {
  return <BrowserRouter>{/* Din applikationsinnehåll */}</BrowserRouter>;
};

export default App;
```

Definiera rutter med Routes och Route

- **Routes** fungerar som en behållare för alla dina **Route**-komponenter.
- Varje **Route** definierar en specifik URL-path och den komponent som ska renderas när den pathen matchas.

```
// Exempel: Definiera rutter
import { Routes, Route } from "react-router-dom";
import HomePage from "../pages/HomePage";
import LoginPage from "../pages/LoginPage";

const AppRoutes = () => {
  return (
    <Routes>
      <Route path="/" element={<HomePage />} />
      <Route path="/login" element={<LoginPage />} />
    </Routes>
  );
};

export default AppRoutes;
```

Routes direkt i App

```
const App = () => {  
  return (  
    <BrowserRouter>  
      <Routes>  
        <Route path="/" element={<HomePage />} />  
        <Route path="/login" element={<LoginPage />} />  
      </Routes>  
    </BrowserRouter>  
  );  
};
```

Navigera med Link

- **Link:** Används för att skapa klickbara länkar som navigerar utan att ladda om sidan.

```
// Exempel: Använda Link
import { Link } from "react-router-dom";

const Navigation = () => {
  return (
    <nav>
      <Link to="/">Hem</Link> | <Link to="/login">Logga in</Link>
    </nav>
  );
};

export default Navigation;
```


UseNavigate

- **useNavigate**: En hook för programmatisk navigering, t.ex. efter att en inloggning lyckats.

```
// Exempel: Använda useNavigate
import { useNavigate } from "react-router-dom";

const LoginPage = () => {
  const navigate = useNavigate();

  const handleLogin = () => {
    // Efter lyckad inloggning
    navigate("/dashboard");
  };

  return <button onClick={handleLogin}>Logga in</button>;
};

export default LoginPage;
```