**Day 76- react router dom**

**Npm I react router dom**

**App.js**

import { useState } from 'react'

import reactLogo from './assets/react.svg'

import viteLogo from '/vite.svg'

import './App.css'

function App() {

  return (

    <>

    <h2>react router dom tutorial</h2>

    </>

  )

}

export default App

**index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App.jsx";

import "./index.css";

import { BrowserRouter } from "react-router-dom";

ReactDOM.createRoot(document.getElementById("root")).render(

  <React.StrictMode>

    <BrowserRouter>

      <App />

    </BrowserRouter>

  </React.StrictMode>

);

**Create some pages** [**home.js,about.js,products.js**](http://home.js,about.js,products.js) **inside src>pages folder**

**App.js**

import { Routes,Route } from 'react-router-dom'

import { Home } from './pages/Home';

import { About } from './pages/About';

import { Products } from './pages/Products'

function App() {

  return (

    <>

    <h2>react router dom tutorial</h2>

    <Routes>

      <Route path='/' element={<Home />}/>

      <Route path='/about' element={<About />}/>

      <Route path='/products' element={<Products />}/>

    </Routes>

    </>

  );

}

export default App

**now check by gng to /, /about, /products endpoint by executing**

**instead of gng to routes manually can add buttons or link tag like navigation tab**

**src>components>Navbar.jsx**

import React from 'react'

export const Navbar = () => {

  return (

    <ul>

        <li>home</li>

        <li>about</li>

        <li>products</li>

    </ul>

  )

}

**Instead of adding anchor tags – can use link tag**

**Instead of href – we use to attribute**

import { Link } from 'react-router-dom'

export const Navbar = () => {

  return (

    <ul>

        <li>

            <Link to='/'>Home</Link>

        </li>

        <li>

            <Link to='/about'>About</Link>

        </li>

        <li>

            <Link to='/products'>Products</Link>

        </li>

    </ul>

  )

}

**Call navbar component in app.js**

function App() {

  return (

    <>

    <header>

      <Navbar />

    </header>

    <h2>react router dom tutorial</h2>

    <Routes>

      <Route path='/' element={<Home />}/>

      <Route path='/about' element={<About />}/>

      <Route path='/products' element={<Products />}/>

    </Routes>

    </>

  );

}

export default App

**can add styling to ul and li**

**include it in App.css or create and keep in Navbar.css**

ul{

  display: flex;

  justify-content: center;

  align-items: center;

  list-style-type:none;

}

li{

  margin-right: 20px;

  font-size: 20px;

}

**Now check execute and check navigation**

**Can use <NavLink > instead of <Link>, can add classname,id like attributes to NavLink**

**Can style it**

**createBrowserRouter:**

**index.js or main.jsx:**

import { createBrowserRouter,createRoutesFromElements,Route,Router } from "react-router-dom";

// import { BrowserRouter } from "react-router-dom";

import { Home } from './pages/Home';

import { About } from './pages/About';

import { Products } from './pages/Products'

const router = createBrowserRouter(

  createRoutesFromElements(

    <Routes>

      <Route path="/" element={<Home />} />

      <Route path="/about" element={<About />} />

      <Route path="/products" element={<Products />} />

    </Routes>

  )

);

**Or specifying main path index page**

const router = createBrowserRouter(

  createRoutesFromElements(

    <Routes path='/'>

      <Route index element={<Home />} />

      <Route path="about" element={<About />} />

      <Route path="products" element={<Products />} />

    </Routes>

  )

);

ReactDOM.createRoot(document.getElementById("root")).render(

  <React.StrictMode>

      <App />

  </React.StrictMode>

);

**Removing browser router**

**Adding element to apply routes**

    <Routes path='/'  element={<App />}>

**Replace app with router provider**

ReactDOM.createRoot(document.getElementById("root")).render(

  <React.StrictMode>

    <RouterProvider router={router } />

  </React.StrictMode>

);

    <Route path='/'  element={<App />}>

      <Route index element={<Home />} />

      <Route path="about" element={<About />} />

      <Route path="products" element={<Products />} />

    </Route>

**No <routes> tag, replace it with <Route>**

**Now navigation working successful**

**But data of different pages is not getting**

**We called home,about ,products page**

**But where to place it in app.js , declare a place using outlet**

**App.jsx**

import { Outlet } from 'react-router-dom'

function App() {

  return (

    <>

    <header>

      <Navbar />

    </header>

    <h2>react router dom tutorial</h2>

    <Outlet />

</>

**Nested routes :**

**Calling other route inside a route – nested routes**

**To understand it**

**Lets create 2 buttons in products page**

import React from 'react'

export const Products = () => {

  return (

    <div>

        <h2>Products Page</h2>

        <button>call or mail us</button>

        <button>Drop a letter</button>

    </div>

  )

}

**Src>components>Comp1.jsx**

import React from 'react'

export const Comp1 = () => {

  return (

    <div>

        <p>Call us at: +917878798</p>

        <p>Email us at : abc@gmail.com</p>

    </div>

  )

}

**Src>components>Address.jsx**

import React from 'react'

export const Address = () => {

  return (

    <div>

        <p>street abc</p>

        <p>area shramika</p>

        <p>pincode 500000</p>

    </div>

  )

}

**Now on clicking those buttons I want to open this pages**

**In index.js/main.jsx**

**We have main path for products page**

      <Route path="products" element={<Products />} />

import { Address } from "./pages/components/Address.jsx";

import { Comp1 } from "./pages/components/Comp1.jsx";

const router = createBrowserRouter(

  createRoutesFromElements(

    <Route path='/'  element={<App />}>

      <Route index element={<Home />} />

      <Route path="about" element={<About />} />

      <Route path="products" element={<Products />}  >  // /products

          <Route path="address" element={<Address />} /> // /products/address

          <Route path="comp1" element={<Comp1 />}/>    //  /products/comp1

      </Route>

    </Route>

  )

);

Products.jsx

import React from 'react'

import { NavLink } from 'react-router-dom'

export const Products = () => {

  return (

    <div>

        <h2>Products Page</h2>

        <button><NavLink to='comp1' >call or mail us </NavLink></button>

        <button><NavLink to='address' >drop a letter</NavLink></button>

    </div>

  )

}

Now check by clicking buttons

It is gng to routes properly but data is not displaying

Add outlet in products page

import React from 'react'

import { NavLink } from 'react-router-dom'

import { Outlet } from 'react-router-dom'

export const Products = () => {

  return (

    <div>

        <h2>Products Page</h2>

        <button><NavLink to='comp1' >call or mail us </NavLink></button>

        <button><NavLink to='address' >drop a letter</NavLink></button>

        <Outlet />

    </div>

  )

}

Adding 404 not found path

const router = createBrowserRouter(

  createRoutesFromElements(

    <Route path='/'  element={<App />}>

      <Route index element={<Home />} />

      <Route path="about" element={<About />} />

      <Route path="products" element={<Products />}  >  // /products

          <Route path="address" element={<Address />} /> // /products/address

          <Route path="comp1" element={<Comp1 />}/>    //  /products/comp1

      </Route>

      <Route path='\*' element={<h2>page not found</h2>}></Route>

    </Route>

  )

);

**Dynamic routing:**

**Now let say in products page, we have so many products**

export const Products = () => {

  return (

    <div>

        <h2>Products Page</h2>

        {/\* <button><NavLink to='comp1' >call or mail us </NavLink></button>

        <button><NavLink to='address' >drop a letter</NavLink></button>

        <Outlet /> \*/}

        <Link>product 123</Link>

        <Link>product 456</Link>

        <Link>product 789</Link>

    </div>

  )

}

div>

        <h2>Products Page</h2>

        {/\* <button><NavLink to='comp1' >call or mail us </NavLink></button>

        <button><NavLink to='address' >drop a letter</NavLink></button>

        <Outlet /> \*/}

        <Link style={{marginRight: '10px'}}>product 1</Link>

        <Link style={{marginRight: '10px'}}>product 2</Link>

        <Link style={{marginRight: '10px'}}>product 3</Link>

    </div>

  )

**Src>pages>ProductDetails.jsx**

import React from 'react'

export const ProductDetails = () => {

  return (

    <div>ProductDetails</div>

  )

}

**Adding nested routes in main.js**

const router = createBrowserRouter(

  createRoutesFromElements(

    <Route path='/'  element={<App />}>

      <Route index element={<Home />} />

      <Route path="about" element={<About />} />

      <Route path="products" element={<Products />}  >  // /products

          {/\* <Route path="address" element={<Address />} /> // /products/address

          <Route path="comp1" element={<Comp1 />}/>    //  /products/comp1 \*/}

          <Route path=":id" element={<ProductDetails/>} />

      </Route>

      <Route path='\*' element={<h2>page not found</h2>}></Route>

    </Route>

  )

);

**Using useParams in productDetails page to access id**

import React from 'react'

import { useParams } from 'react-router-dom'

export const ProductDetails = () => {

    const params=useParams();

    console.log('params: ' + params)

  return (

    <div>

    </div>

  )

}

**Products.jsx**

export const Products = () => {

  return (

    <div>

        <h2>Products Page</h2>

        {/\* <button><NavLink to='comp1' >call or mail us </NavLink></button>

        <button><NavLink to='address' >drop a letter</NavLink></button>

        <Outlet /> \*/}

        <Link  to='123' style={{marginRight: '10px'}}>product 1</Link>

        <Link  to ='456' style={{marginRight: '10px'}}>product 2</Link>

        <Link to='789' style={{marginRight: '10px'}}>product 3</Link>

    </div>

  )

}

import React from 'react'

import { useParams } from 'react-router-dom'

export const ProductDetails = () => {

    const params=useParams();

    console.log('params: ' , params)

  return (

    <div>

        this is product details page

    </div>

  )

}

Adding outlet in products page

    <div>

        <h2>Products Page</h2>

        <Link  to='123' style={{marginRight: '10px'}}>product 1</Link>

        <Link  to ='456' style={{marginRight: '10px'}}>product 2</Link>

        <Link to='789' style={{marginRight: '10px'}}>product 3</Link>

        <Outlet />

    </div>

In console got params obj with id inside it . clicked id like 123,456,

**Productdetails.jsx**

return (

    <div>

        this is product details page {params.id}

    </div>

  )

**78 diy:**

import { useEffect, useState } from 'react'

import reactLogo from './assets/react.svg'

import viteLogo from '/vite.svg'

import './App.css'

function App() {

  const [data, setData] = useState([])

  const [e,setE] = useState()

  const [out, setOut] = useState()

  useEffect(() => {

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

      .then(result => result.json())

      .then((resp )=> {

        // console.log("result",resp)

        setData(resp)

        console.log(data)

      })

      .catch((err)=>{

        console.log(err)

      })

  }, [])

  // console.log(data)

 function filteration(){

  let arr;

  if(e!=''){

    arr=data.filter(item => item.id == Number(e))

  }

  else{

    arr=data;

  }

  setOut(arr)

  console.log('arr is');

  console.log(out)

  }

  return (

    <>

    <input type="text" placeholder='Enter id' onChange={(ev)=>setE(ev.target.value)} />

    <button onClick={()=>{

        let arr;

        if(e!=''){

          arr=data.filter(item => item.id == Number(e))

        }

        else{

          arr=data;

        }

        setOut(arr)

        console.log('arr is');

        console.log(out)

    }}>Submit</button>

    {out.map((item)=>{

      return  <div>

        <h4>{`${item.id} - ${item.title}`}</h4>

        <div>{item.body}</div>

      </div>

    })}

    </>

  )

}

export default App