Routing

import './App.css';

import { BrowserRouter, Routes, Route } from "react-router";

import Contactus from './pages/Contactus';

import About from './pages/About';

import Home from './pages/Home';

import Products from './pages/Products';

function App() {

  return (

    <BrowserRouter>

      <Routes>

        <Route path="/contactus" element={<Contactus />} />

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

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

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

      </Routes>

    </BrowserRouter>

  );

}

export default App;

import React from 'react'

import { Link, useNavigate } from 'react-router'

const Products = () => {

    const navigate = useNavigate();

    const goToHomePage = () => {

        navigate('/');

    };

    return (

        <div>

            <h4>This is Products Page</h4>

            <Link to="/">Go To Home</Link>

            <br />

            <button onClick={goToHomePage}>Go to Home</button>

        </div>

    )

}

export default Products

Add Params

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

import React from 'react'

import { useParams } from 'react-router'

const Singleproducts = () => {

    const params = useParams()

    console.log(params);

    return (

        <div>Singleproducts</div>

    )

}

export default Singleproducts

Not Found

const NotFound = () => <h1>Not Found</h1>;

  <Route path="\*" element={<NotFound />} /

DYNAMIC PARAMS

     <button onClick={() => {

                navigate(`/products/${count}`)

            }}>

                Go to Single Products

            </button>

GET SINGLE PRODUCTS

import React from 'react';

import { Link } from 'react-router';

const ProductsList = () => {

    const products = [

        { id: 1, name: "Product 1", description: "This is Product 1", price: "$10" },

        { id: 2, name: "Product 2", description: "This is Product 2", price: "$20" },

        { id: 3, name: "Product 3", description: "This is Product 3", price: "$30" },

    ];

    return (

        <div>

            <h1>Products List</h1>

            <ul>

                {products.map(product => (

                    <li key={product.id}>

                        <Link to={`/products/${product.id}`}>{product.name}</Link>

                    </li>

                ))}

            </ul>

        </div>

    );

};

export default ProductsList;

import React from 'react';

import { useParams } from 'react-router';

const SingleProduct = () => {

    const products = [

        { id: 1, name: "Product 1", description: "This is Product 1", price: "$10" },

        { id: 2, name: "Product 2", description: "This is Product 2", price: "$20" },

        { id: 3, name: "Product 3", description: "This is Product 3", price: "$30" },

    ];

    const { id } = useParams();

    const product = products.find(p => p.id === parseInt(id));

    return (

        <div>

            {

                product ? <>

                    <h1>{product.name}</h1>

                    <p>{product.description}</p>

                    <p>Price: {product.price}</p>

                </> : <h2>Product not found</h2>

            }

        </div>

    );

};

export default SingleProduct;

PASS TWO PARAMS

 <Route path="/products/:id/:user" element={<Singleproducts />} />

  <Link to={`/products/${product.id}/username`}>

 const { id, user } = useParams();

Nested routes

import { BrowserRouter, Routes, Route, Outlet } from "react-router";

function Profile() {

  return (

    <div>

      <h2>Profile Components</h2>

    </div>

  );

}

function Setting() {

  return (

    <div>

      <h2>Setting Component</h2>

    </div>

  );

}

function Dashboard() {

  return (

    <div>

      <h2>Dashboard Component</h2>

      <Outlet />

    </div>

  );

}

function App() {

  return (

    <BrowserRouter>

      <Routes>

        <Route path="/dashboard" element={<Dashboard />}>

          <Route path="profile" element={<Profile />} />

          <Route path="setting" element={<Setting />} />

        </Route>

      </Routes>

    </BrowserRouter>

  );

}

export default App