**npx create react-app .**

this cmd will cretae react app in current folder

then install, react royter dom

**npm install react-router-dom**

**run project**

**npm start**

**remove headeer tag and logo from app.js**

**rename title from public >index,html to E-Commerce**

**Project structure Inseide src**

**Components – 1 Assets**

**2 Navbar – 2.1 Navbar.jsx**

**Pages – use for Ecommercee pages**

**Context – will use for context API**

**Creatig new component using RAFCE**

RAFCE stands for React Arrow Function Component Export. It is a code snippet that can be used to quickly create a React functional component with an export default statement. This can be useful for quickly creating new components or for refactoring existing components.

**Components – 1 Assets**

**2 Navbar – 2.1 Navbar.jsx**

**2.2 Navbar.css for bavbar.jsx**

**Import css in jsx**

import './Navbar.css'

**mount navbar.jsx in app.js**

import './App.css';

import Navbar from './Components/Navbar/Navbar';

function App() {

  return (

    <div >

      <Navbar/>

    </div>

  );

}

export default App;

**copy assets from** [**https://drive.google.com/file/d/1Nw2Rm88hJ3zowWQAzqwoL0O6a0vA41Nf/view**](https://drive.google.com/file/d/1Nw2Rm88hJ3zowWQAzqwoL0O6a0vA41Nf/view)

**copy assets data inside assets folder of project**

**in all\_profucs.js all product data presne tin json format**

**here we have**

{

    id: 1,

    name: "Striped Flutter Sleeve Overlap Collar Peplum Hem Blouse",

    category: "women",

    image: p1\_img,

    new\_price: 50.0,

    old\_price: 80.5,

  },

**Import imahes from assets inside navbar.jsx**

import logo from '../Assets/logo.png'

import cart\_icon from '../Assets/cart\_icon.png'

src={logo} meaning

<img src={logo} alt="Company Logo" />

In the context of web development or programming, **src={logo}** typically represents the source attribute of an HTML or JSX (JavaScript XML) element, where "logo" is a variable or reference pointing to the location or source of an image file.

**jr image import krun img tah mdhe use krachi asel tr tya image la**

import logo from '../Assets/logo.png'

**import krun**

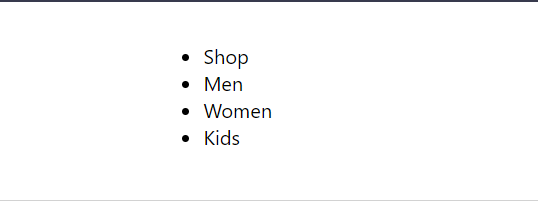
<img src={logo} alt="Company Logo" />

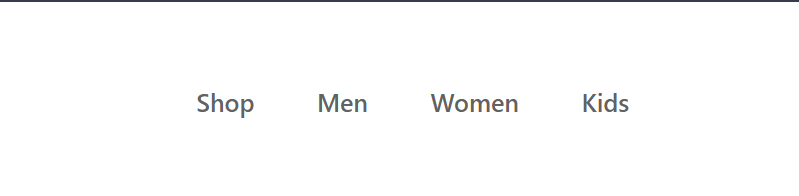
**Curly bracket mdhe use kra**

<img src={logo} alt="" />

            <p>Shopper</p>

**Img ani shopper name side by side sathi**

****

****

.nav-menu{

        display: flex;

        align-items: center;

        list-style: none;

        gap: 50px;

        color: #626262;

        font-size: 20px;

        font-weight: 500;

      }

<ul className="nav-menu">

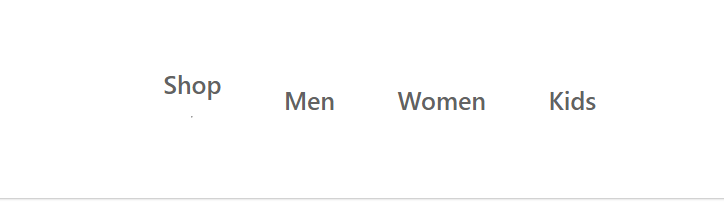
                <li>Shop <hr /></li>

                <li>Men</li>

                <li>Women</li>

                <li>Kids</li>

            </ul>

****

.nav-menu hr{

        border: none;

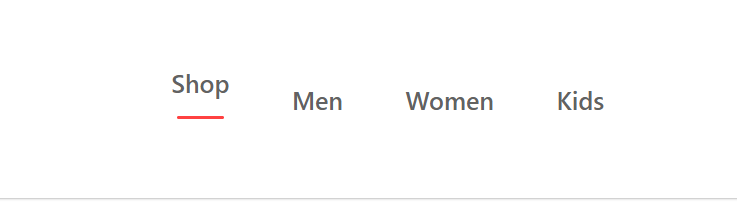
        width: 80%;

        height: 3px;

        border-radius: 10px;

        background: #ff4141;

      }

****

****

.nav-login-cart{

        display: flex;

        align-items: center;

        gap: 45px;

      }

****

 /\* adding counter on the top of logo (which was in the side of logo) \*/

      .nav-cart-count{

        height: 22px;

        width: 22px;

        display: flex;

        justify-content: center;

        align-items: center;

        margin-top: -35px;

        margin-left: -55px;

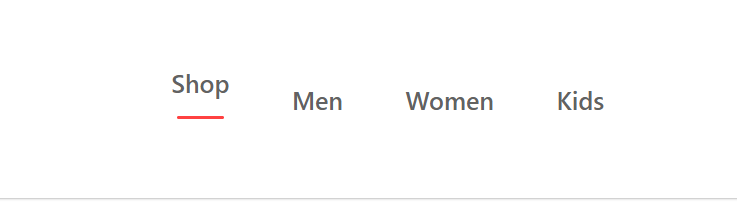
        border-radius: 11px;

        font-size: 14px;

        background: red;

        color: white;

      }

****

**Jewha shop vr click kru tewha underline shop vr anu jewha men vr underline men vr**

usestate in react

In React, useState is a hook that allows functional components to manage and update state. State is a way for a component to keep track of information that may change over time and trigger re-renders when it does. The useState hook is part of the Hooks API introduced in React 16.8.

Here's some basic information about useState:

**Syntax:**

**const [state, setState] = useState(initialState);**

**state: The current state value.**

**setState: A function used to update the state.**

**initialState: The initial value of the state.**

**Example:**

**import React, { useState } from 'react';**

**const Counter = () => {**

**// Using the useState hook to manage state**

**const [count, setCount] = useState(0);**

**return (**

**<div>**

**<p>Count: {count}</p>**

**<button onClick={() => setCount(count + 1)}>Increment</button>**

**</div>**

**);**

**};**

**In this example, the count state is initialized to 0, and the setCount function is used to update its value. When the "Increment" button is clicked, it triggers a re-render with the updated count value.**

**Key Points:**

**Initialization: useState is typically called with an initial state value, which is the value the state will have when the component is first rendered.**

**Returns an Array: The useState function returns an array with two elements: the current state value and a function that can be used to update the state.**

**Immutable Update: State updates are asynchronous, and React merges the updated state with the current state. Always use the function form of the state update to ensure you are working with the latest state.**

**Multiple States: You can use useState multiple times in a component to manage different pieces of state.**

**Functional Updates: setState can accept a function as an argument, allowing you to perform updates based on the previous state.**

**setCount((prevCount) => prevCount + 1);**

he use of **useState** significantly simplifies state management in functional components compared to older class components and encourages the use of functional programming patterns in React applications.

import React , { useState }from 'react'

**onllcik arrown function setmenu() => setmenu mdhe je ahe tyane replace honar**

**shop vr click zalyavr set menu shop honar**

**mens vr click zalyavr set menu mens honar**

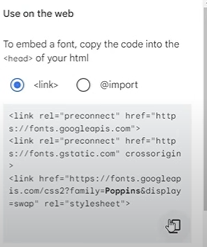
**<li onClick={()=>{setMenu("shops")}}>Shop {menu==="shops"?<hr />:<></>} </li>**

**If menu is shops then use hr tag otherwise empty tag**

**Now set google font poppins**

**Regular font 400**

**Paste it in index.html**

****

<link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap" rel="stylesheet">

**After pasting code**

**Inside index,css**

**Font-famaily:”popins”**

body {

  margin: 0;

  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

    sans-serif;

  -webkit-font-smoothing: antialiased;

  -moz-osx-font-smoothing: grayscale;

  font-family: "poppins";

**poppins family get applied**

**Pages folder**

* 1. **Shop.jsc**
  2. **LoginSignup.jsx**
  3. **ShopCategory.jsx**
  4. **Product.jsx**
  5. **Cart.jsx**

**Use rafce to convert ot onto component**

**Shop page is our home page**

**ShopCategory.jsx ; we will render diff category men womens kids**

**App.js we will set up our routes and we will link that with our navigation bar**

**Inside app.js**

**Import Browser router, routes, route from react router dom**

**We have to wrap all compo using this browser router**

<BrowserRouter>

      <Navbar/>

      </BrowserRouter>

**This nav bar will available in all compo**

**After navbar we will setup our routes**

**In routes we need to set route**

**In route we need to use path**

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

**If someone comes to mens , womens. Kids**

**path then we will render to shop category page**

<Route path='/mens' element={ShopCategory}></Route>

        <Route path='/womens' element={ShopCategory}></Route>

        <Route path='/kids' element={ShopCategory}></Route>

<Route path='/womens' element={<ShopCategory category="women"/>}></Route>

when the user navigates to the '/womens' path in the application, the **ShopCategory** component will be rendered with the category prop set to "women".

**Now we will add some props**

**If someonen cpmes pn path mens then category=”mens”**

<Routes>

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

        <Route path='/mens' element={<ShopCategory category="Men"/>}></Route>

        <Route path='/womens' element={<ShopCategory category="women"/>}></Route>

        <Route path='/kids' element={<ShopCategory category="kid"/>}></Route>

        <Route path='/product' element={<Product/>} ></Route>

        <Route path=':productid' element={<Product/>}></Route>

        <Route path='/cart' element={<Cart/>}></Route>

        <Route path='/login' element={<LoginSignup/>}></Route>

      </Routes>

**Route is done, now we will link this route with navigation bar, we have to set navigation menu with route so we will use link tag**

<ul className="nav-menu">

                <li onClick={()=>{setMenu("Shop")}}> <Link to='/'>Shop</Link> {menu==="Shop"?<hr />:<></>} </li>

                <li onClick={()=>{setMenu("Men")}}><link to='/men'>Men</link> {menu==="Men"?<hr/>:<></>}</li>

                <li onClick={()=>{setMenu("Women")}}> <Link to='/women'>Women</Link> {menu==="Women"?<hr/>:<></>}</li>

                <li onClick={()=>{setMenu("Kids")}}><Link to='/kids'></Link> {menu==="Kids"?<hr/>:<></>}</li>

            </ul>

**Mow add link to login button, then move button tag inside link**

 <div className="nav-login-cart">

            <Link to='/login'><button>Login</button></Link>

            <Link to='/carts'><img src={cart\_icon} alt="" /></Link>

            <div className="nav-cart-count">0</div>

        </div>

****

**After click li get underlined so make styling**

style={{textDecoration:'none'}}

In React, curly braces **{}** are used for embedding JavaScript expressions or code within JSX.

**onClick Event Handling:**

<li onClick={() => {setMenu("shops")}}>

The **onClick** attribute expects a function to be executed when the element is clicked. In this case, an arrow function is used (**() => {setMenu("shops")}**) to create an inline function that calls the **setMenu** function with the argument "shops" when the list item is clicked. The curly braces are used to denote the body of the arrow function.

**style Attribute:**

<Link style={{textDecoration:'none'}} to='/'>

1. The **style** attribute expects a JavaScript object for defining inline styles. In React, the double curly braces (**{{}}**) are used because the outer curly braces indicate that we are embedding JavaScript within JSX, and the inner curly braces create an object literal for the inline styles. For example, **{{textDecoration:'none'}}** sets the **textDecoration** style property to 'none' for the **Link** component.

In both cases, the use of curly braces allows you to incorporate JavaScript logic or expressions within JSX. The **onClick** and **style** attributes are dynamic and can be controlled by the React component's state or other logic, providing flexibility in handling events and styling based on dynamic conditions.

**now we have to create hero component for our shop page**

**Component**

**Hero folder**

**Hero.jsx**

**Rafce convert to component**

**Create hero css file import it in hero.jsx**

**Add hero in shop**

**Create html str for hero element ie hero.jsx**

<div className="hero">

        <div className="hero-left">

          <h2>New Arrivals only</h2>

          <div>

            <div className="hand-hand-icon">

                <p>new</p>

                <img src={hand\_icon} alt="" />

            </div>

            <p>Collection</p>

            <p>for everyone</p>

          </div>

          <div>

            <div className="hero-latest-btn">

                <div>text-latest-collection</div>

                <img src={arrow\_icon} alt="" />

            </div>

          </div>

        </div>

        <div className="hero-right">

            <img src={hero\_image} alt="" />

        </div>

    </div>

**To remove extra space margi zero to all inside index.css**