Experiment 8

<u>Aim:</u> Create React Sidebar Navigation Menu for a webpage. Implement the concept of React Hooks for backward compatibility and faster user experience.

1) In components folder, make Navbar.js, and design the Navbar

```
src > Components > Navbar > JS Navbar.js > ...
  1 v import React, { useState } from "react";
    import * as FaIcons from "react-icons/fa";
    import * as AiIcons from "react-icons/ai";
    import { Link } from "react-router-dom";
    import { SidebarData } from "./SidebarData.js";
     import { IconContext } from "react-icons";
 9 v function Navbar() {
      const [sidebar, setSidebar] = useState(false);
       const showSidebar = () => setSidebar(!sidebar);
          <IconContext.Provider value={{ color: "#fff" }}>
            <div className="navbar">
             <Link to="#" className="menu-bars">
              <FaIcons.FaBars onClick={showSidebar} />
             <span class="rollnumber">21BIT005 : React Sidebar</span>
            <nav className={sidebar ? "nav-menu active" : "nav-menu"}>
              <Link to="#" className="menu-bars">
                   <AiIcons.AiOutlineClose />
                {SidebarData.map((item, index) => {
                   <Link to={item.path}>
                      {item.icon}
                      <span>{item.title}</span>
     export default Navbar;
```

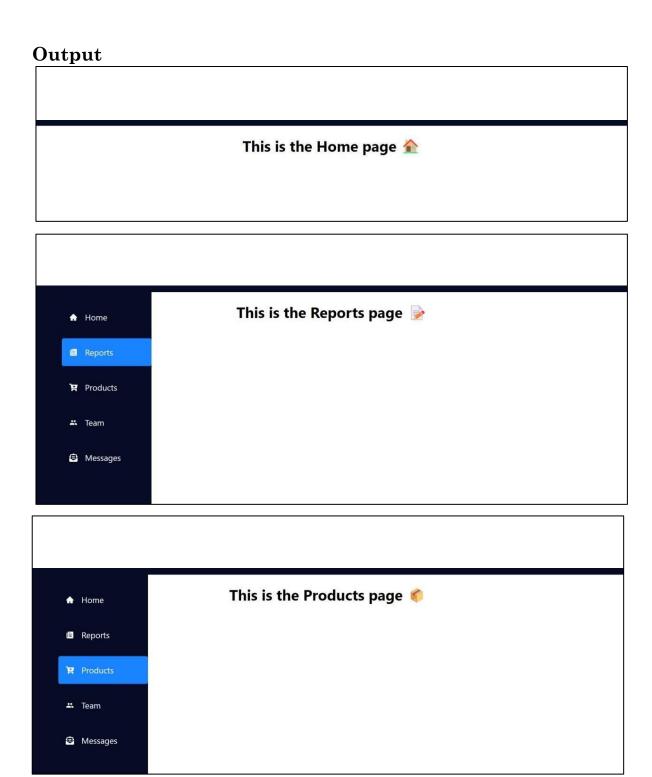
2) Give CSS to Navbar

```
.nav-text a {
.navbar {
                                              text-decoration: none;
   background-color: □#060b26;
                                              color: #f5f5f5;
   height: 80px;
                                              font-size: 18px;
   display: flex;
                                              width: 95%;
   justify-content: start;
                                              height: 100%;
   align-items: center;
                                              display: flex;
                                              align-items: center;
 .menu-bars {
                                              padding: 0 16px;
   margin-left: 2rem;
                                              border-radius: 4px;
   font-size: 2rem;
   background: none;
                                            .nav-text a:hover {
 .nav-menu {
                                              background-color: #1a8
   background-color: □#060b26;
   width: 250px;
                                            .nav-menu-items {
   height: 100vh;
                                             width: 100%;
   display: flex;
   justify-content: center;
                                            .navbar-toggle {
   position: fixed;
                                             background-color: #060
   top: 0;
                                              width: 100%;
   left: -100%;
                                              height: 80px;
   transition: 850ms;
                                              display: flex;
                                              justify-content: start;
                                              align-items: center;
  .nav-menu.active {
   left: 0;
   transition: 350ms;
                                            span {
                                             margin-left: 16px;
 .nav-text {
   display: flex;
                                            .rollnumber{
   justify-content: start;
                                              padding-left: 450px;
   align-items: center;
                                              color: white;
   padding: 8px 0px 8px 16px;
                                              font-weight: bold;
   list-style: none;
                                              font-size: x-large;
   height: 60px;
                                    76
                                            }
```

3) Make a Sidebar.js in components folder, to design the sidebar

4) Make a pages folder, and add the pages of the sidebar

5) Update the App.js file



Discussions:

This experiment demonstrated the creation of a React sidebar navigation menu, enhancing webpage accessibility. Utilizing React Hooks ensures faster user interaction and maintains compatibility across different React versions.