REACT ROUTER

It is a standard liberary for routing in React.

It enables the navigation among views of various components in a React Application, allows changing the browser URL, and Keeps the UI in sync with the URL without refreshing the page.

=> Installation Command

npm ~~is~~ install react-router-dom

=> Then go to index.js file and wrap BrowserRouter around App i.e.

```
<BrowserRouter>
        <App/>
</BrowserRouter>
```

=> Now create routes in App.js file using

```
<Routes>

</Routes>
```

tag

To create single route we use following tag

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

From the above routing method we have to change the link in order to make changes in the page. i.e to go from Home to ~~an~~ About.

To avoid this problem we will use <NavLink> tag to show link of the various other links.

<NavLink to="/"> Home<NavLink>

By using above tag we can automatically add active class in the link ~~in~~ in which we have clicked.

We can also use nesting in <Route> tag but by default it will not show the childs of the parent <Route> element.

In order to make the visible we will use <Outlet/> tag in MainHeader.js

```
Ex <Routes>                              Parent Route
       <Route path="/" element={<MainHeader/>}>
It makes it   <Route index element={<Home>}>
default route <Route path="/About" element={<Suppor/>}>
       </Route>                                   child route
    </Routes>
```

When you use <Outlet/> tag it show the child content and parent content also, so if you want to remove the parent content from every child content.

Then you have to remove the content from the parent element. From this when you go to child element you won't see the content of the parent element.

But when you click on the parent element you'll show nothing because you have already removed the content.

Parent Route

To avoid this create a MainHeader
~~an~~ route and give path of Home to
it and add the content to the
child Route Home route. Same as we did in the
previous example.

Use Navigate Hook (useNavigate())

This hook helps you to go to specific URL, forward or backward pages in the React Router. (For better understanding implement this)