**React Router**

1. React router DOM is not an essential part of React it is a third party library.
2. To install React router use: **npm i react-router-dom**
3. In react router DOM we have Link which used in place of <a> tag, because when we use <a> tag, it reloads the whole page but link don’t reload the complete page it just show the result.
4. **To make color change on clicking like home, about, contact us, etc we add a call back function in our classes and then we inject the variable with if-else case condition.**
   1. **This is our callback:**



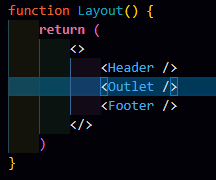
* 1. **This is how we inject the variable:**



1. To import the components we don’t need App.jsx file we can directly call all the components by using main.jsx.
2. To make navigation bar available in all pages many people call the navigation bar in all pages like calling navigation bar in home page, about us page, etc.



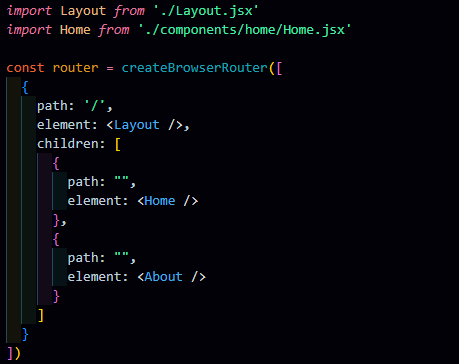
1. To solve this.
2. We create a simple .jsx file and we import **outlet** form **react-router-dom**, then we make the layout as we want like first we keep header, **<outlet />** and footer.



1. **<outlet>** helps to change the things dynamically and **keep the header and footer same** in all the pages and **change only part of <outlet />.**
2. In main.jsx we import RouterProvider and pass the router={router} as parameter:



1. And then we call all these **in the main.jsx file**

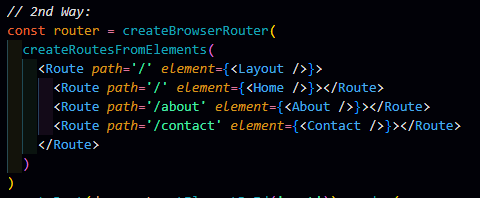


**And we have to use “to” attribute in NavLink to change the pages.**

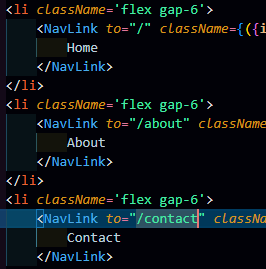
1. Ways to create routes:
   1. 1st Way:



* 1. 2nd Way:



1. If we want more links like home, about, contact then we have to:



1. We can also pass some parameters in the URL, which will change according to the parameter.

* 1. **To achieve this:**
  2. Create user name component and this component to the path and use **:(and pass the name you want like: userName, userID, etc.)**



* 1. Keep in mind to use userName only everywhere don’t use other thing.
  2. Then in user component pass the variable of parameter.



* 1. Import the useParams in user component to use this.
  2. Then create a custom hook to use.

