

~~28/6/21~~

Date:

* Links → So, in real life we won't be just remembering all the URLs and using them. So for that we will be putting links.

→ So, if the user visits a website by remembering its URL then next task is of us to provide links in order for the user to go ahead and use the website, or the adjacent web pages.

→ So, previously we used '<a>' tags,

Ex -

< >

<div>

 Courses </>

</div>

<div>

 About Us </>

</div>

</>

So, in case of '<a>' tags what

happens is the reloading of the pages and we don't want that to happen and I want it to behave like any single page application only, and there will be no reloading happening and still we will be redirected to other pages.]

- So, to avoid this we use link.
- Link is an another component in react-router which we have.

Ex-

```
import {Link} from 'react-router-dom'
const header = () => {
  return (
    <>
    <div>
      <Link to="/courses"> Courses </Link>
    </div>
    <div>
      <Link to = "/about"> About Us </Link>
    </div>
  );
}
```

→ We can also multiple objects in the Link Component.

Ex →

<Link to={

pathname: "/courses",
hash: "#1234",
search: "?Paid:true".
33 > Courses </Link>

→ So, now when we go ahead we will see all the parameters getting passed in the url.

→ So, in order to access it we use "props".

Ex -

const Courses = (props) => {

 console.log(props)

 return (

 <>

 Courses

 </>

 }

→ When we console the 'props' we will see a lot of objects on the console. for ex → heading, location, Params etc.

* Navlinks → Suppose, if we have a header and if we go ahead and select any particular item, that particular item is either highlighted or is bold or some styling is been applied and that we achieve using CSS.

→ One other way of achieving it in a more easy way is that if we use navlinks.

Ex - <>
<div>
<NavLink to="/about"> About Us </>
</div>
</>

To apply styling →

<>
<NavLink activeStyle={{color: "red", fontWeight: "bold", border: "1px solid red", padding: "5px 10px", margin: "5px 0"},
to="/Courses" > Courses </NavLink>
</>

→ We also have activeClassName in which we can simply use the className and the properties of that element with the className will be defined inside an external css file.

Ex import ". /styles / header.css"
<NavLink

activeClassName = "header-active" to = "/courses"
Courses

</NavLink>

header.css

• header - active {

font-weight : 700 ;
font-size : 2 rem ;
color : red ;

y