**React Router Dom**

* Client side routing
* Manage advance features in routing
* Also manage authentication routing

**Install & Setup**

* To install reac router dom   
  npm install react-router-dom
* After successfully install router, now config it. Wrap full app in <BrowserRouter><App /> </BrowserRouter>

**Basic Routing**

* To create a basic routing in react   
  <Routes>  
   <Route path=”” element={} />  
  </Routes>

**404 Routing**

* To create a basic routing in react   
  <Routes>  
   <Route path=”\*” element={} />  
  </Routes>

**Link vs NavLink**

* Link is used to navigate to a route   
  <Link to=”/blog”> Blog<Link>
* But navlink helps to know the active route   
  <NavLink to=”/blog”> Blog</NavLink>

**Dynamic Route**

* To manage route with dynamic parameters

<Route path=”/blog/:slug” element={} />

* Now get this slug as params   
  const { slug } = useParams();
* Get Search params   
  const [ searchParams, setSearchParams ] = useSearchParams();
* Get Search params   
  searchParams.get(‘key\_name’)

**Nested Route**

* Nested route is used for nested routing
* It helps routing in a route
* Structure of nested   
  <Route path=”” element={}>  
   <Route path=”” element={} />  
  </Route>
* To load nested component   
  <Outlet />

**Nested index Route**

* It define index element in nested routing   
  <Route path=”” element={}>  
   <Route index element={} />  
   <Route path=”” element={} />  
  </Route>

**Navigate hook**

* To navigate anywhere use navigate hook

const navigate = useNavigate();

* Navigate to route   
  navigate(‘/route\_name’, { replace : true } );
* Navigate back history route   
  navigate(-1);

**Lazy Loading**

* To reduce loading time for large content
* To load lazy loading first import component through react lazy

**Create Custom Browser Router**

* We can manage our router by creating custom browser
* To manage custom way first create a browser router   
  const router = createBrowserrouter([  
   {  
   path : “”,  
   element : <Conponent>  
   }  
  ]);
* Now use this router into our application   
  <RouterProvider router={router}>