

virtual DOM is object representation of actual DOM.

AUGUST

S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
E	01	02	03	04	05	06	07	08	09	10	11	12	13	14
P	15	16	17	18	19	20	21	22	23	24	25	26	27	28
	29	30												

28

THURSDAY

Week 35

Day (240-125)

Appointments

09 Whenever state variable update, react triggers a reconciliation (render the component cycle).

10 useEffect ( ) = {

11 console.log ("hello"); // This runs when // component mounts

12 return () => console.log ("bye"); // This when // component unmounts

13 } , [ ] );

14 Infinite Scrolling & how to do it is in code? with explanation.

15 It uses Intersection Observer class & useRef.

16 Episode-07 Finding the Path

17

When the useEffect called?

The useEffect is called after the render of the component.

→ mean it will run everytime when we update state or anything in component

If no dependency array => useEffect is called on every render.

If empty dependency array => useEffect is called only on initial render (just once when the component is rendered for first time).

→ It will not run everytime when we change state or anything in component.

2025

If  
29

FRIDAY

Week 35

Day (241-124)

AUGUST

A	M	T	W	T	F	S	S	S	M	T	W	T	F	S	S
A						01	02	03	04	05	06	07	08	09	10
U						11	12	13	14	15	16	17	18	19	20
G						25	26	27	28	29	30	31			

#### Appointments

09 useEffect does not cause re-render itself unless there is a state in it.

10

If dependency array is [btnName] then useEffect is called after btnName changes, although it will be called on initial render.

#### 13 Some useState tips

- Never ~~use~~ <sup>create</sup> useState outside the body of component
- Always try to create useState at the top of body of component
- Don't create useState inside any condition (like if-else) cause it can have inconsistency as due to condition, there will be that state variable & sometimes not.
- Don't create inside loop.

#### Making Routes in react application

The library we will be using for this react routes demo

For creating route, we have to create 2025 routing configuration

CreateBrowserRouter → will create configuration  
RouterProvider → will provider/render configuration  
in our app.

AUGUST

S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
E	01	02	03	04	05	06	07	08	09	10	11	12	13	14
P	15	16	17	18	19	20	21	22	23	24	25	26	27	28
P	29	30												

In App.js  
Appointments

09 import { CreateBrowserRouter } from "react-router-dom";  
RouterProvider

10 const AppLayout = () => {  
11 return (  
12 <div>  
13 <Head />  
14 <Body />  
15 </div>  
16 );  
17 }

Creating config

18 const appRouter = CreateBrowserRouter ([  
19 {  
20 path: "/",  
21 element: <AppLayout />  
22 },  
23 {  
24 path: "/about",  
25 element: <About />  
26 },  
27 ]);  
28 ]);

31 SUNDAY

const root = ReactDOM.createRoot (document.getElementById("root"));  
root.render (<RouterProvider router={appRouter} />);  
↳ Rendering config.

2025

30

SATURDAY

Week 35 Day (242-123)

01

MONDAY

Week 36 Day (244-121)

SEPTEMBER

S	M	T	W	T	F	S	S	M	T	W	F	S	S
	01	02	03	04	05	06	07	08	09	10	11	12	13
	P	15	16	17	18	19	20	21	22	23	24	25	26
		29	30										

Appointments

09 How to add error page for your Route?

10

{

path: ''''

element: &lt;AppLayout/&gt;

errorElement: &lt;Error/&gt;

}

13

There is a hook provided by react-router-dom

14 called useRouteError

15 which gives more info about error

16 const err = useRouteError();

17 { err.status } {err.statusText} .  
(404) (Not Found)

What if we want our Header should remain intact in every page.

In App.js

const AppLayout = () => {  
 return (<div className = "app">  
 <Header/>  
 <Body/>  
</div>

);

# Action Plan

September

Schedule your plan .....For Better future.

2025

Wk	36	37	38	39	40
----	----	----	----	----	----

What we want that <Header> remain  
 Note there bid our <Body> change for About page.  
 & other pages.

To do this we will use childrenRoute  
 and Outlet component or from  
 react-router-dom  
 import {Outlet} from 'react-router-dom';  
 const AppLayout = () => {  
 return (

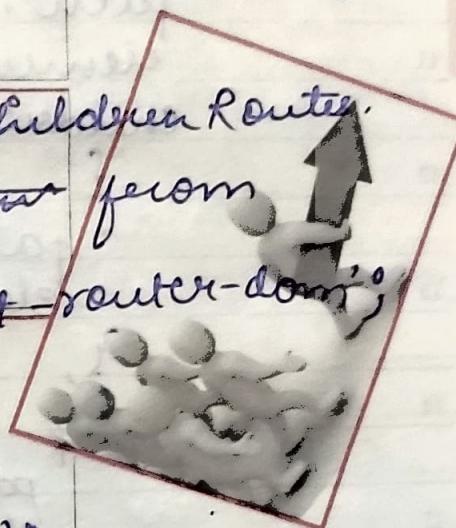
4 <div class="app">

<Header />  
 <Outlet />

5 </div>

19

26



Be  
Leader

THU

FRI

SAT

SUN

);  
};

Now in app outlet

If your actions  
 inspire others to  
 dream more, learn  
 more or do more

You are a  
 Leader.

SEPTEMBER

OCTOBER

NOVEMBER

DECEMBER

02

• TUESDAY

Week 36 • Day (245-120)

SEPTEMBER

S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
E	01	02	03	04	05	06	07	08	09	10	11	12	13	14
P	15	16	17	18	19	20	21	22	23	24	25	26	27	28
	25	29	30											

## Appointments

09 const appRoutes = createBrowserRouter (P)

```

10   {
11     path: "/",
12     element: <AppLayout />
13     children: [
14       {
15         path: "/",
16         element: <Body />
17       },
18       {
19         path: "/about",
20         element: <About />
21       },
22       {
23         path: "/contact",
24         element: <Contact />
25       }
26     ],
27     errorElement: <Error />
28   });
  
```

Now the outlet will be replaced by  
children acc to routes.

For eg:- If we are on "/" then it  
will be replaced by <Body /> & if we  
are on "/about" then replaced by <About />

SEPTEMBER

That's why React is called single-page application cause we are just replacing component in the main app.js up to route.

S	M	T	W	T	F	S	S	M	T	W	T	F	S
01	02	03	04	05	06	07	08	09	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
29	30	31											

WEDNESDAY

Week 36

Day (246-119)

03

Appointments

## Navigation in React

09

If we use `<a>` tag for navigation then we the page will refresh & it will reload the page which will take time

11

So, we will use Link component from `react-router-dom` then the page will not reload it will just replace the things which is different from the page from we are navigating

14

`<Link to = "path to your page"> Home </Link>`

Two types of routing in web apps

① Client-Side Routing

② Server-Side Routing

⇒ Server-Side Routing

If we have an `index.html`, `about.html` if we click on anchor tag ("about") it will reload the whole page, it sends a new call to `about.html`, fetches it and render on our webpage. and reload the whole page.

⇒ Client-Side Routing

Here we doesn't make new all the ~~routes~~ code is in our app so it just loads the component when we click on the route. The only new call we make is for api. what we done here with `links` in routing.

04

THURSDAY

Week 36

Day (247-118)

SEPTEMBER

S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
S	01	02	03	04	05	06	07	08	09	10	11	12	13	14
P	15	16	17	18	19	20	21	22	23	24	25	26	27	28
	25	29	30											

Appointments

## ~~Dynamic Routing~~

10

To make a dynamic route we will use

11

const appRouter = createBrowserRouter({

12

path: "/restaurants/:resId"

13

element: <RestaurantMenu />

14

}

To get the Id in the component we will use 'params' from react-router-dom

16

import { params } from "react-router-dom";

17

const { resId } = params;

Note:- Whenever you are using map function you have to use key to uniquely identify each item and it should on parent jsx.

restaurant.map((res) => {

<Link key={res.id} to={`/restaurants/\${res.id}`}>

2025

<RestaurantCard /></Link>

} );

GraphQL helps you with overfetching to load only the data which is required.

SEPTEMBER

M	T	W	T	F	S	S
01	02	03	04	05	06	07
08	09	10	11	12		
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

05

FRIDAY

Week 36

Day (248-117)

Appointments Here Link is parent json some use key on it.

0

11

12

13

14

15

16

17

OCTOBER

DECEMBER

2025