Lecture No 11

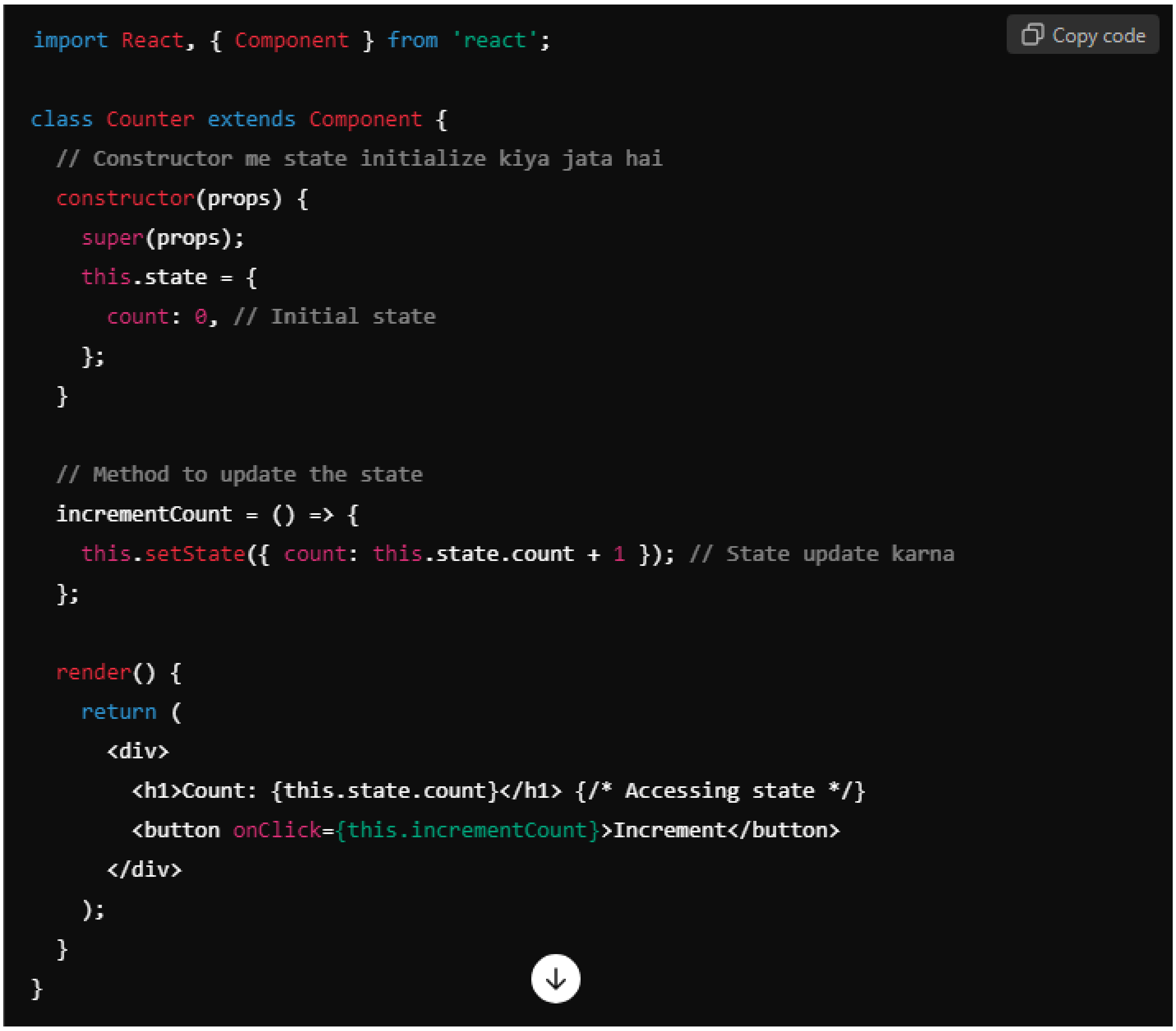
*class compts me ham data state object k ander rakhty hen , state ek built-in object ha Component class me jo React provide krta ha.

React class components me **state** ek special built-in object hoti hai jo Component class ka hissa hoti hai, aur React ne ye feature provide kiya hota hai. State ka main kaam hai component ke internal data ko store karna jo time ke sath change ho sakta hai, aur jab state update hoti hai, to React component ko re-render karta hai taake updated UI dikhayi ja sake.

Key Points:

1. **State Object**:
   - state class component ka internal data hota hai.
   - Iske andar aap dynamic data ko store karte hain, jo kisi user interaction ya kisi event ke base pe change ho sakta hai.
2. **Component Class me Built-in**:
   - Class components me state object React ne automatically Component class ke through provide ki hoti hai. Aap state ko directly constructor me define karte hain ya setState method ka use karke update karte hain.
3. **State Changes Trigger Re-render**:
   - Jab bhi state change hoti hai, React component ko re-render karta hai taake naye state values ke sath updated UI dikhayi ja sake.

Example of Class Component with State:



Copy code

```
import React, { Component } from 'react';

class Counter extends Component {
  // Constructor me state initialize kiya jata hai
  constructor(props) {
    super(props);
    this.state = {
      count: 0, // Initial state
    };
  }

  // Method to update the state
  incrementCount = () => {
    this.setState({ count: this.state.count + 1 }); // State update karna
  };

  render() {
    return (
      <div>
        <h1>Count: {this.state.count}</h1> {/* Accessing state */}
        <button onClick={this.incrementCount}>Increment</button>
      </div>
    );
  }
}
```

****

Lecture No 13

How we can dynamically load component with URL patterns (Routes) ?

URL = Uniform Resource Location

- Address Bar <---- URL
- Address Bar <---- www.google.com

The default URL of ReactJs Application is

This is a URL of react Application **Base Address** ➔ http://localhost:3000

- Base Address/Home ==➔Route
- Base Address/someRoute

Suppose I have to create 2 Pages

1.Login

http://localhost:3000/login

2. Register

http://localhost:3000/register

3. Home

http://locahost:3000

*Base address k bad / laga kr jo bhi likhty hen wo rcat me routes hoty hen . in routes ki bunyad pe ham apne compts ko dynamically render krwa skty hen.

react-router-dom is a 3rd party library

npm install react-rotuer-dom

npm install <someLibrary> -------> https://npmjs.com ----> downloaded in ur system

working from scratch:

- public/index.html (View File)
- src/index.js (Login File)
- package.json

Flow of React App:

npm start --> react-script start----> src/index.js-->> Webpack-->Babel--> public/index.html

React Router Dom Flow

Grandparent (BrowserRouter Compt)

Parent (Routes Compt)

Child (Route compt)

Grandparent (BrowserRouter)

BrowserRouter component React app ke top-level pe use hota hai. Iska kaam hota hai app ke andar routing ko manage karna using the browser's history API. Yeh har routing-related component ko context provide karta hai.

Parent (Routes)

Routes component routes ko define karta hai. Yeh ek wrapper hota hai jo multiple Route components ko manage karta hai. Yeh app ke andar define karta hai kaunse URL pe kaunsa component render hoga.

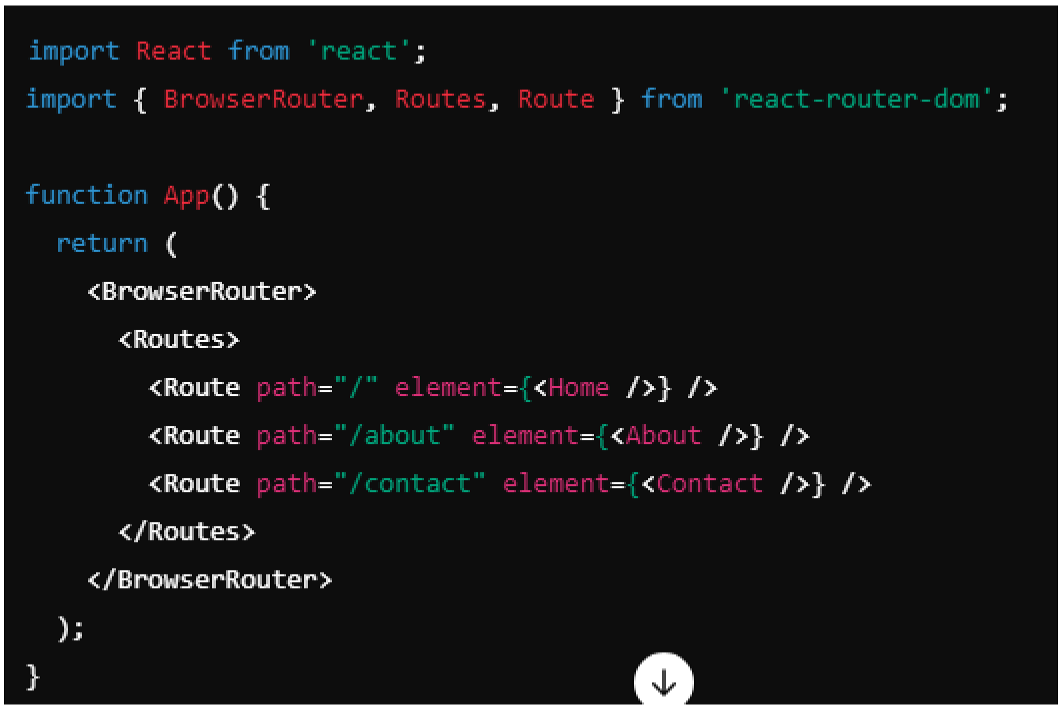
Child (Route)

Route component ka kaam hota hai specific URL path ke liye component ko render karna. Jab user kisi specific route pe jata hai (jaise /about ya /contact), to us route ka corresponding component render hota hai.

Flow Explanation:

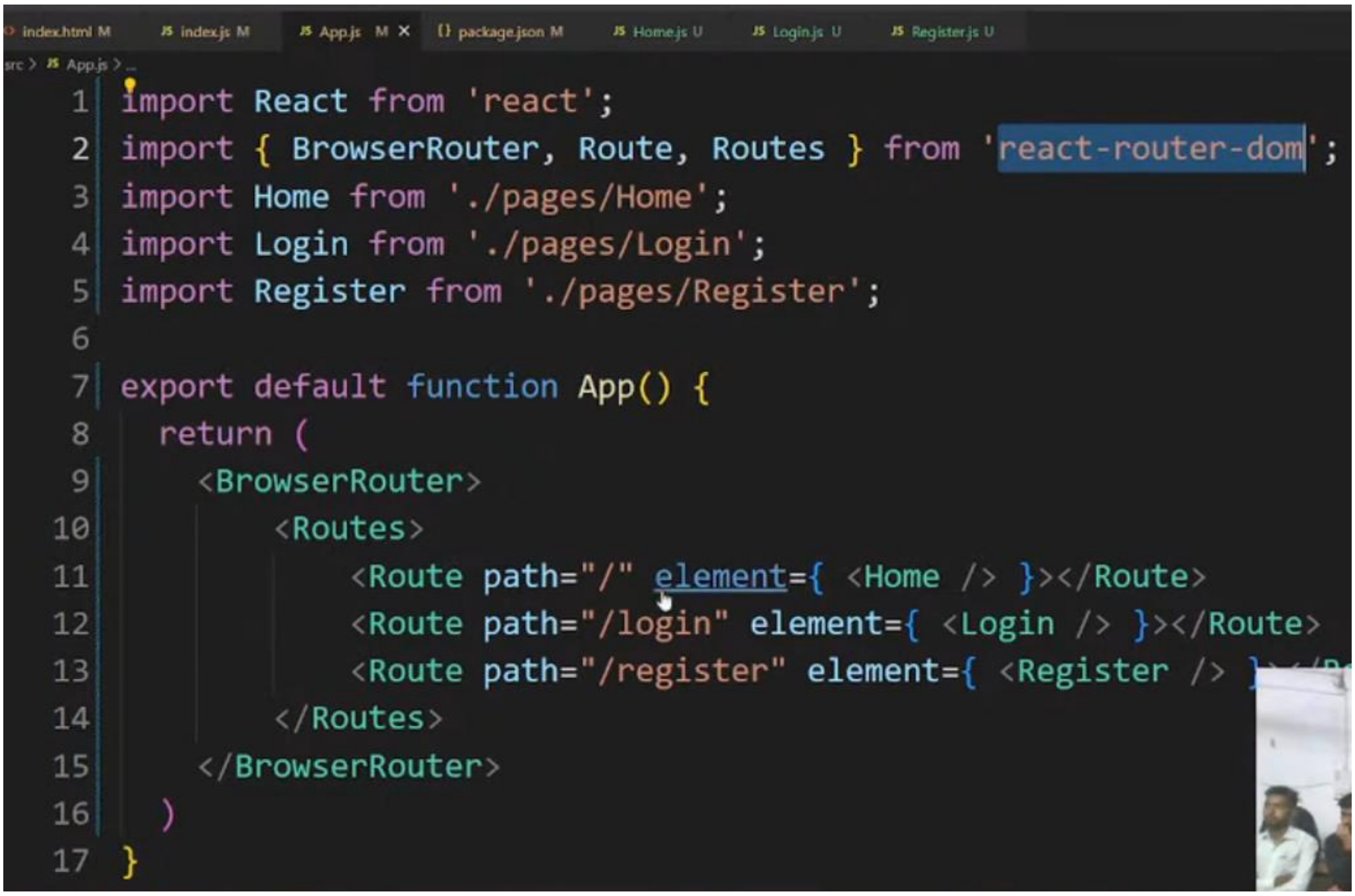
1. **BrowserRouter (Grandparent)**: Yeh app ka wrapper hai jo routing system ko manage karta hai.
2. **Routes (Parent)**: Isme tum define karte ho ki kaunse routes available hain, aur kaunse path pe kaunsa component render hoga.
3. **Route (Child)**: Har Route component specific path ke liye ek component ko render karta hai, jaise /about pe About component aur /contact pe Contact component.

Is tarah ka structure follow karne se React app ke andar routing manage karna easy ho jata ha



```
import React from 'react';
import { BrowserRouter, Routes, Route } from 'react-router-dom';

function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
        <Route path="/contact" element={<Contact />} />
      </Routes>
    </BrowserRouter>
  );
}
```

****

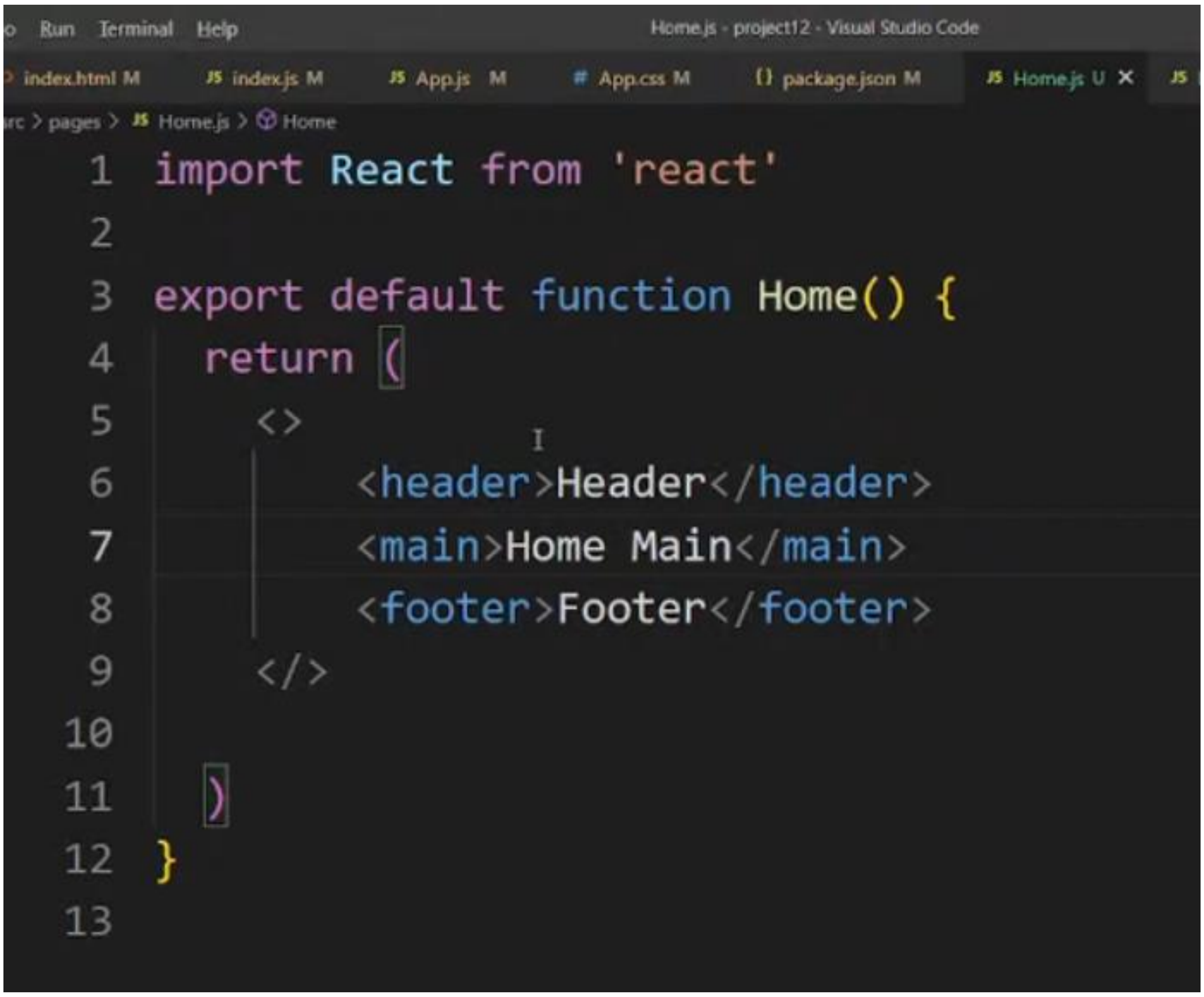

index.html M | index.js M | App.js M | package.json M | Home.js U | Login.js U | Register.js U

src > App.js > ...

```
import React from 'react';
import { BrowserRouter, Route, Routes } from 'react-router-dom';
import Home from './pages/Home';
import Login from './pages/Login';
import Register from './pages/Register';

export default function App() {
  return (
    <BrowserRouter>
        <Routes>
            <Route path="/" element={ <Home /> }></Route>
            <Route path="/login" element={ <Login /> }></Route>
            <Route path="/register" element={ <Register /> }
        </Routes>
    </BrowserRouter>
  )
}
```

Yahan tak to smjh aati ha………………………..

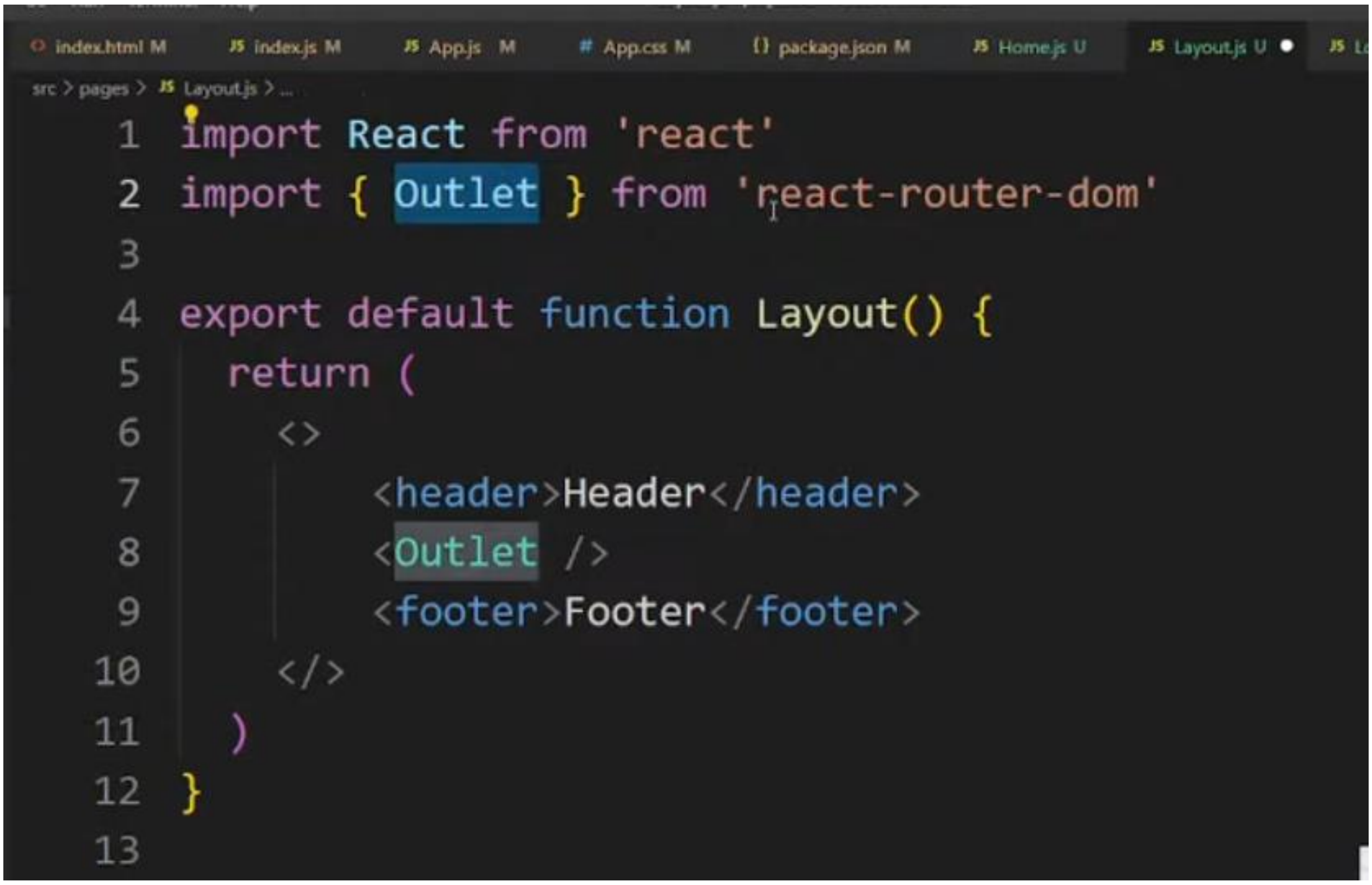
Ab Home Compt me kuch content put krty hen



```
import React from 'react'

export default function Home() {
  return (
    <>
        <header>Header</header>
        <main>Home Main</main>
        <footer>Footer</footer>
    </>

  )
}

```

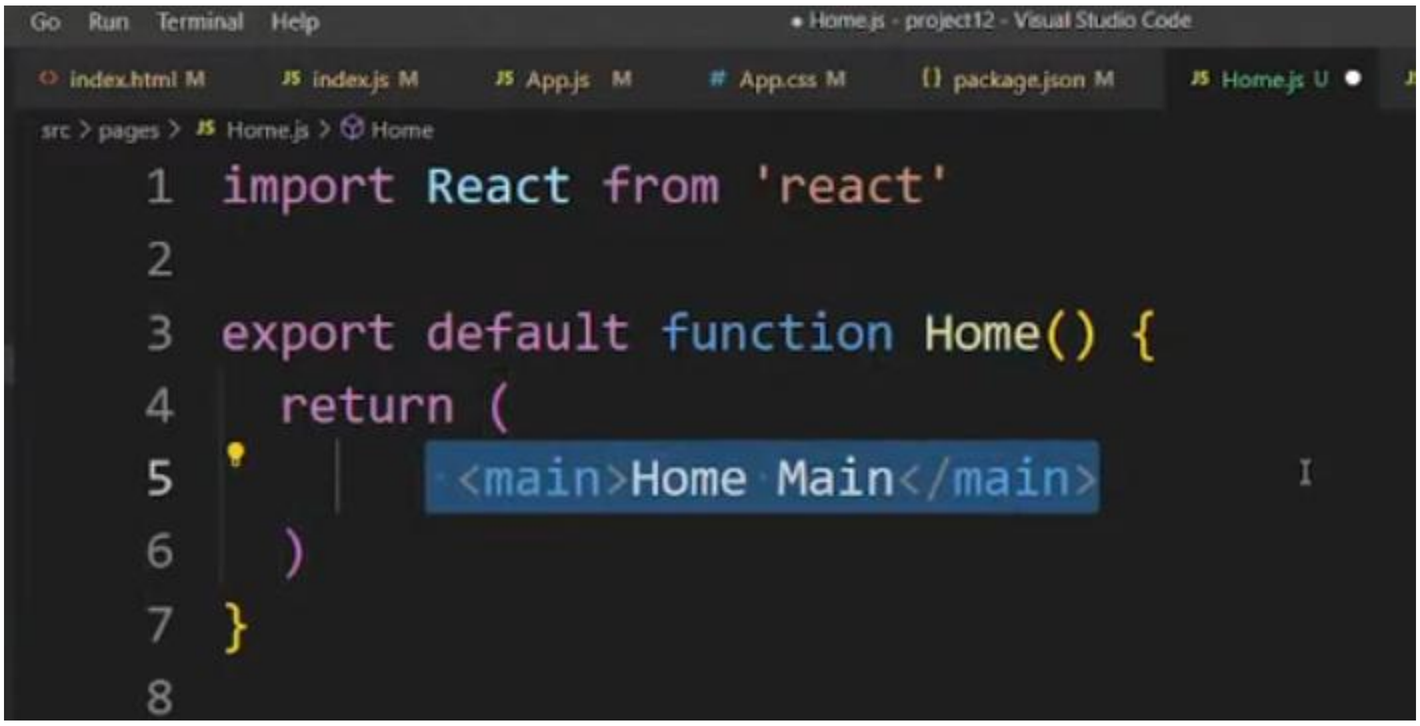
Ab me chata hu k jab /login ho to header & footer to same rahy magar main area me login ka compt render ho, ab us ke liye hamen ek Layout compt banana paray ga, usme hamen pora home page ka structure put krna paray ga . (oper wala)

Or jo compt change hona ha wahan “**outlet**” compt use kren .



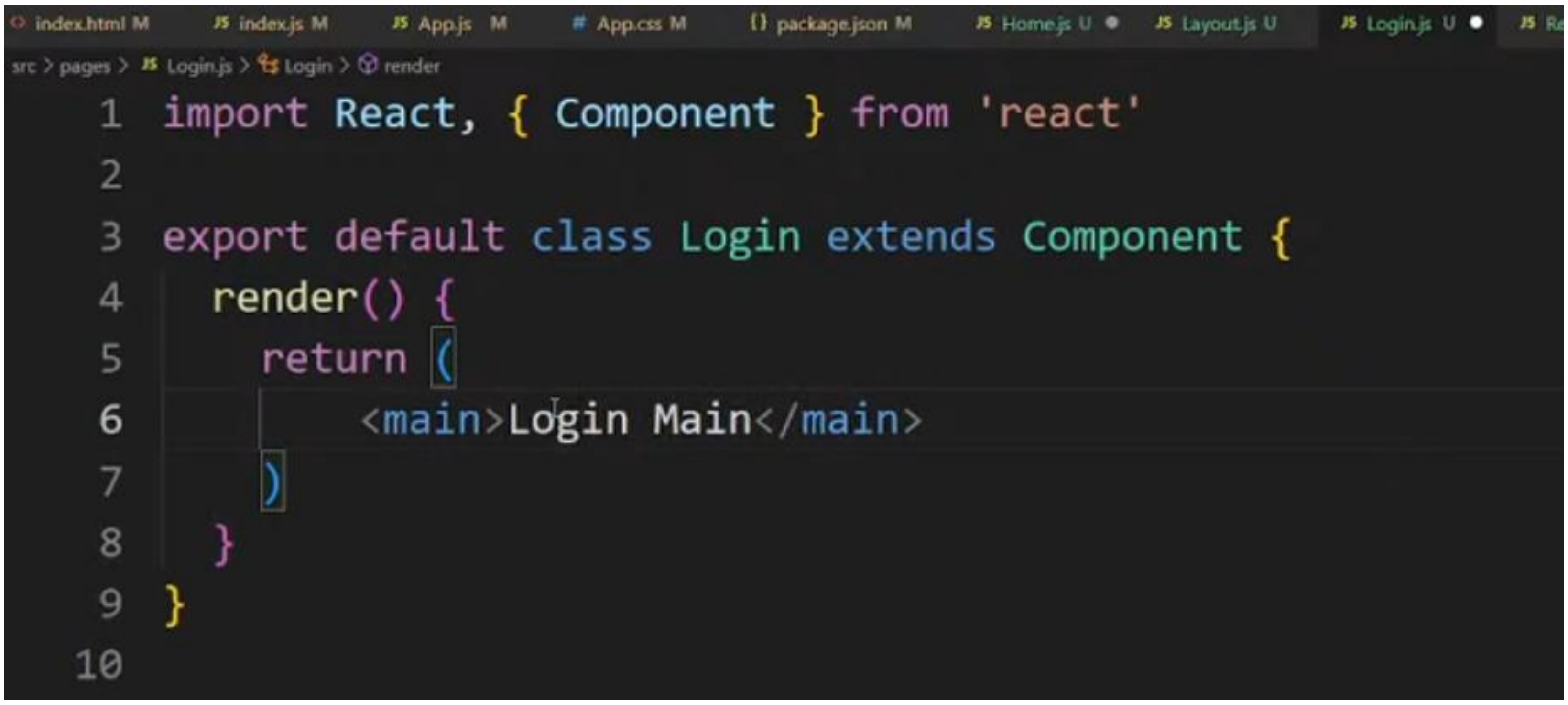
```
import React from 'react'
import { Outlet } from 'react-router-dom'

export default function Layout() {
  return (
    <>
        <header>Header</header>
        <Outlet />
        <footer>Footer</footer>
    </>
  )
}

```

Ab home.js me ja kr wo cheez sirf rakho jo Home pe render krna ha , Qk ham ne Home k content ko change krna ha baki sab to same rahy ga.

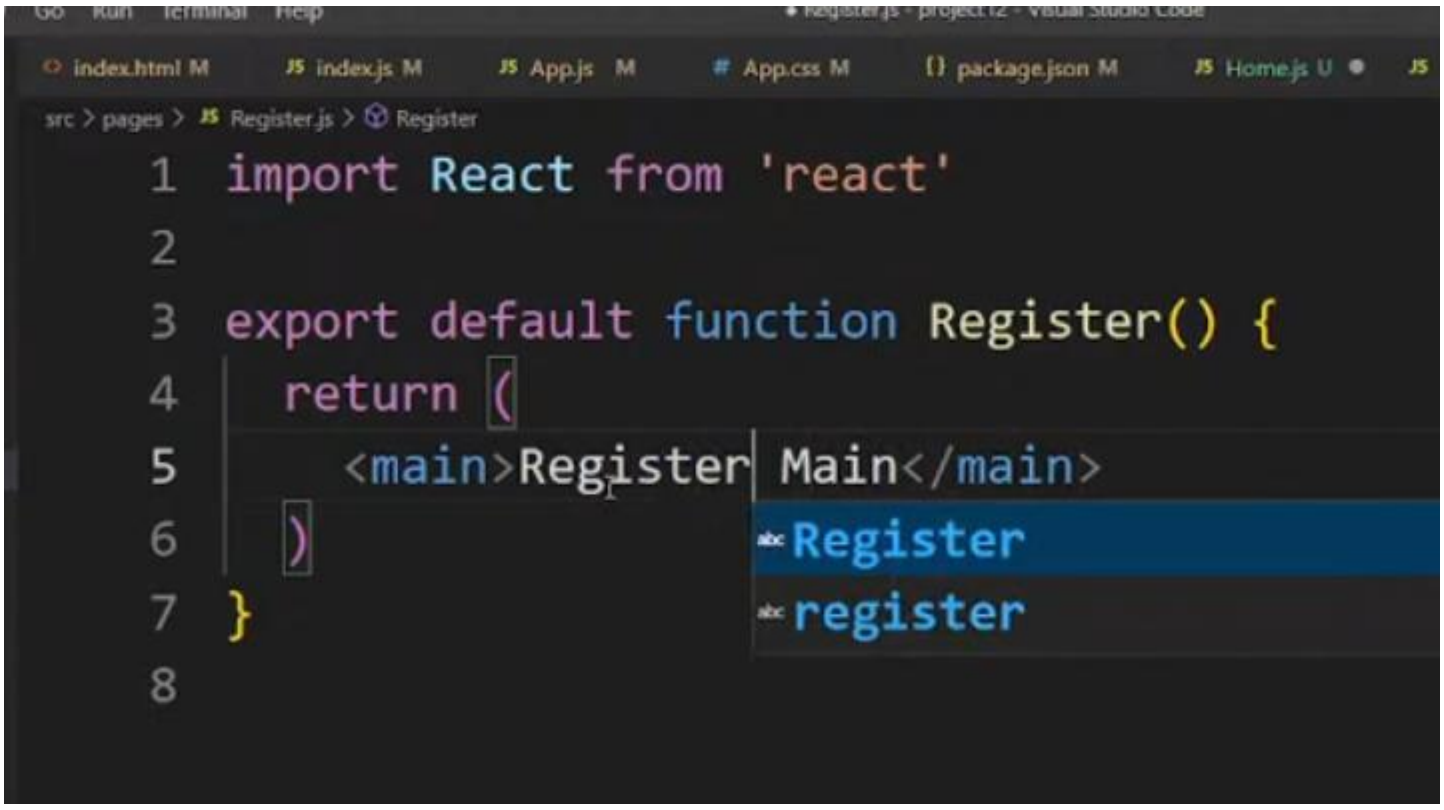


```
import React from 'react'

export default function Home() {
  return (
      <main>Home Main</main>
  )
}

```

Ab isi traha login or register ko bhi change krdo :

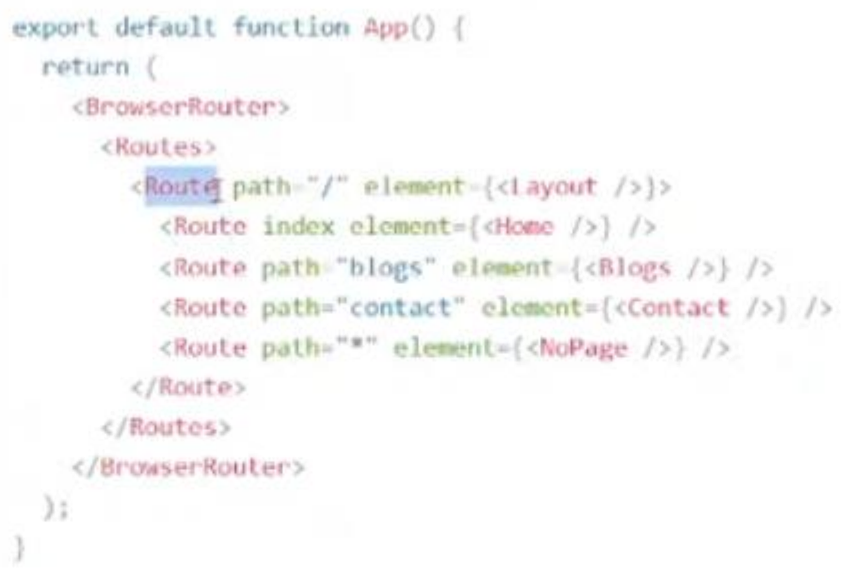


```
import React, { Component } from 'react'

export default class Login extends Component {
  render() {
    return (
        <main>Login Main</main>
    )
  }
}
```

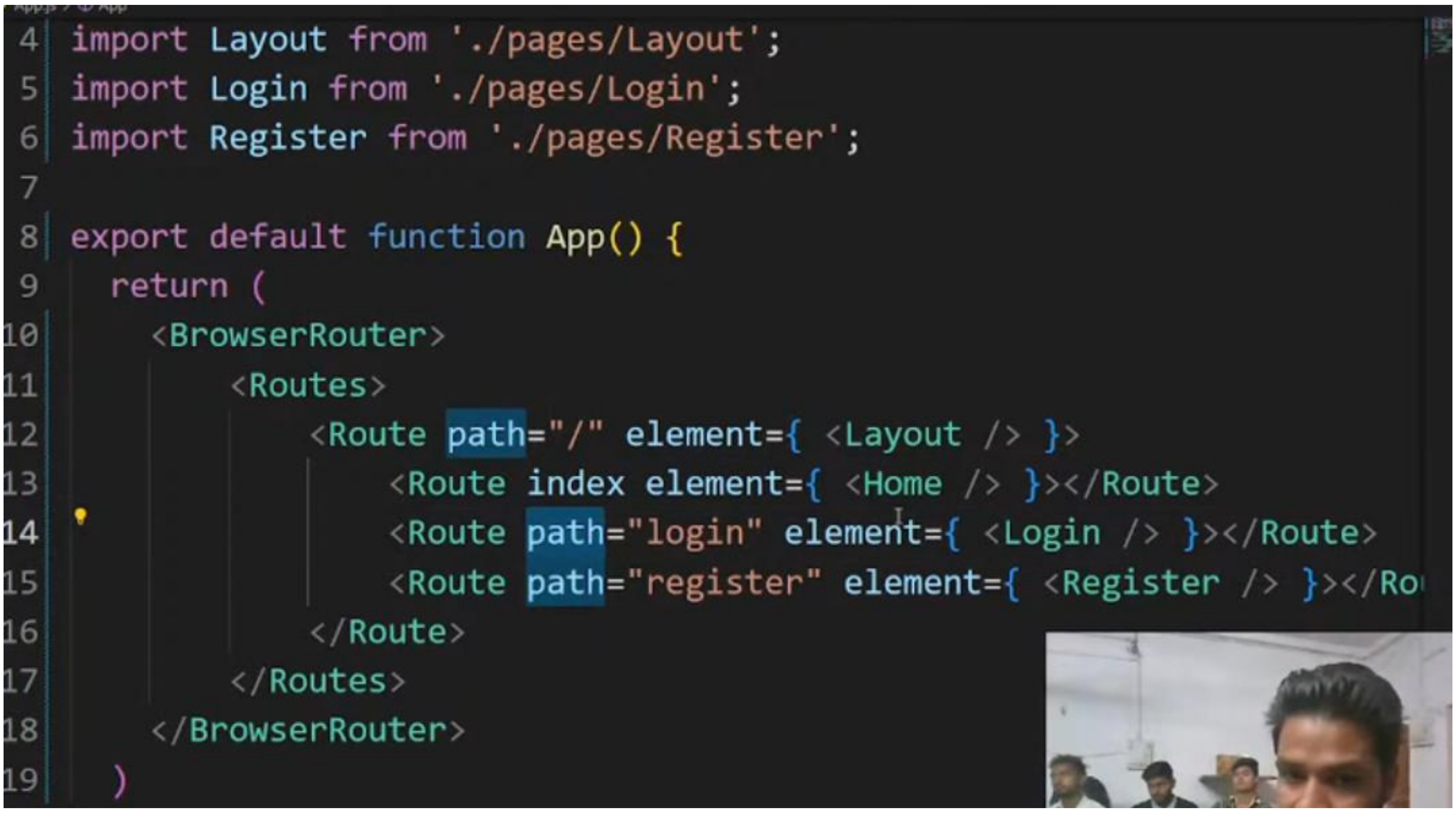


```
import React from 'react'

export default function Register() {
  return (
    <main>Register Main</main>
  )
}
```

Ab thora react router k structure me changes lana hongi route k abder further route dene paren gy:



```
export default function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Layout />}>
          <Route index element={<Home />} />
          <Route path="blogs" element={<Blogs />} />
          <Route path="contact" element={<Contact />} />
          <Route path="*" element={<NoPage />} />
        </Route>
      </Routes>
    </BrowserRouter>
  );
}
```



```
import Layout from './pages/Layout';
import Login from './pages/Login';
import Register from './pages/Register';

export default function App() {
  return (
    <BrowserRouter>
        <Routes>
            <Route path="/" element={ <Layout /> }>
                <Route index element={ <Home /> }></Route>
                <Route path="login" element={ <Login /> }></Route>
                <Route path="register" element={ <Register /> }></Ro
            </Route>
        </Routes>
    </BrowserRouter>
  )
```

To ab ye nested ki tarah kaam kren gy , means

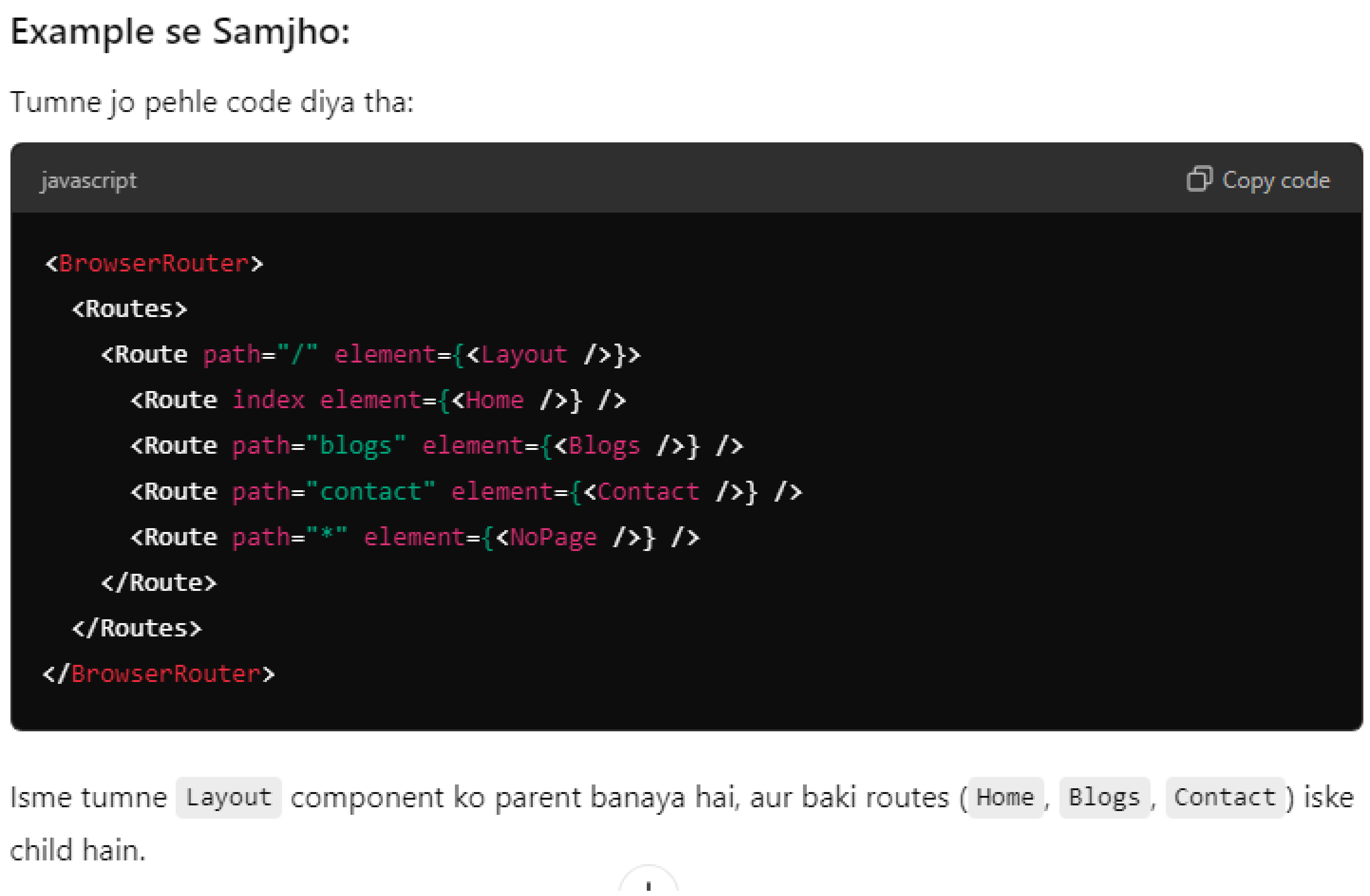
baseURL/ =➔ Us case me Layout component ke saath-saath jo index route define kiya gaya hai, wo bhi render hoga. Yani ke Layout ke andar jo Home component hai, wo bhi render hoga.

baseUrl/login ➔ Agar aapka browser ka address bar http://localhost:3000/blogs hai, to is case mein:

1. **Layout Component Render Hoga**: Jab aapka route / se start hota hai, to pehle <Route path="/" element={<Layout />}> match hota hai, is liye Layout component render hoga.
2. **Blogs Component Render Hoga**: Uske andar, aapka route <Route path="blogs" element={<Blogs />} /> hai. Yeh specific route hai jo /blogs par match hota hai, is liye Blogs component bhi render hoga.

<outlet/> ki kahani:

<Outlet /> React Router me ek placeholder hota hai jahan child routes render hoti hain. Iska use tab hota hai jab tumhare pass nested routes hon, yaani ek parent route ke andar aur bhi routes hoon.

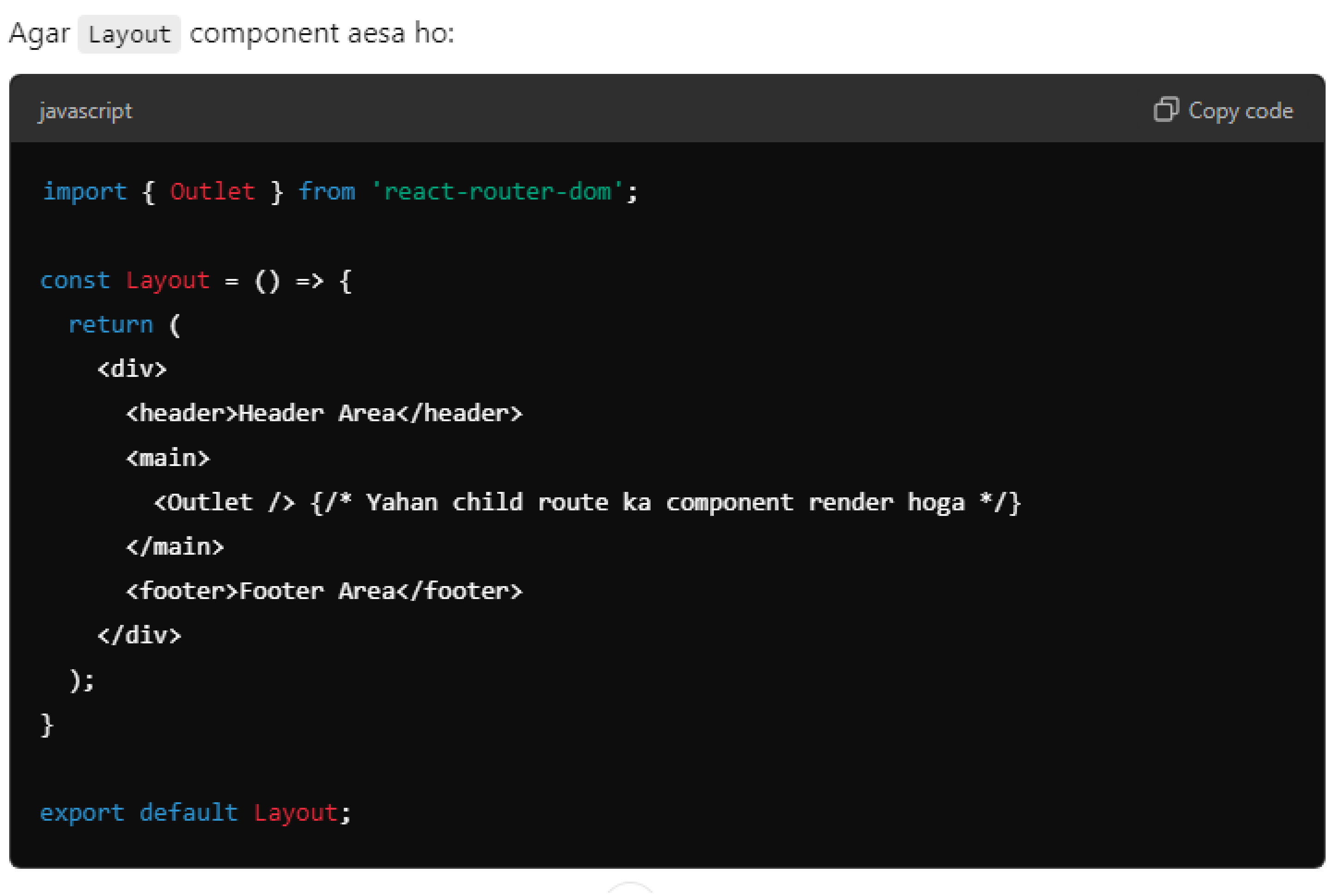


Example se Samjho:

Tumne jo pehle code diya tha:

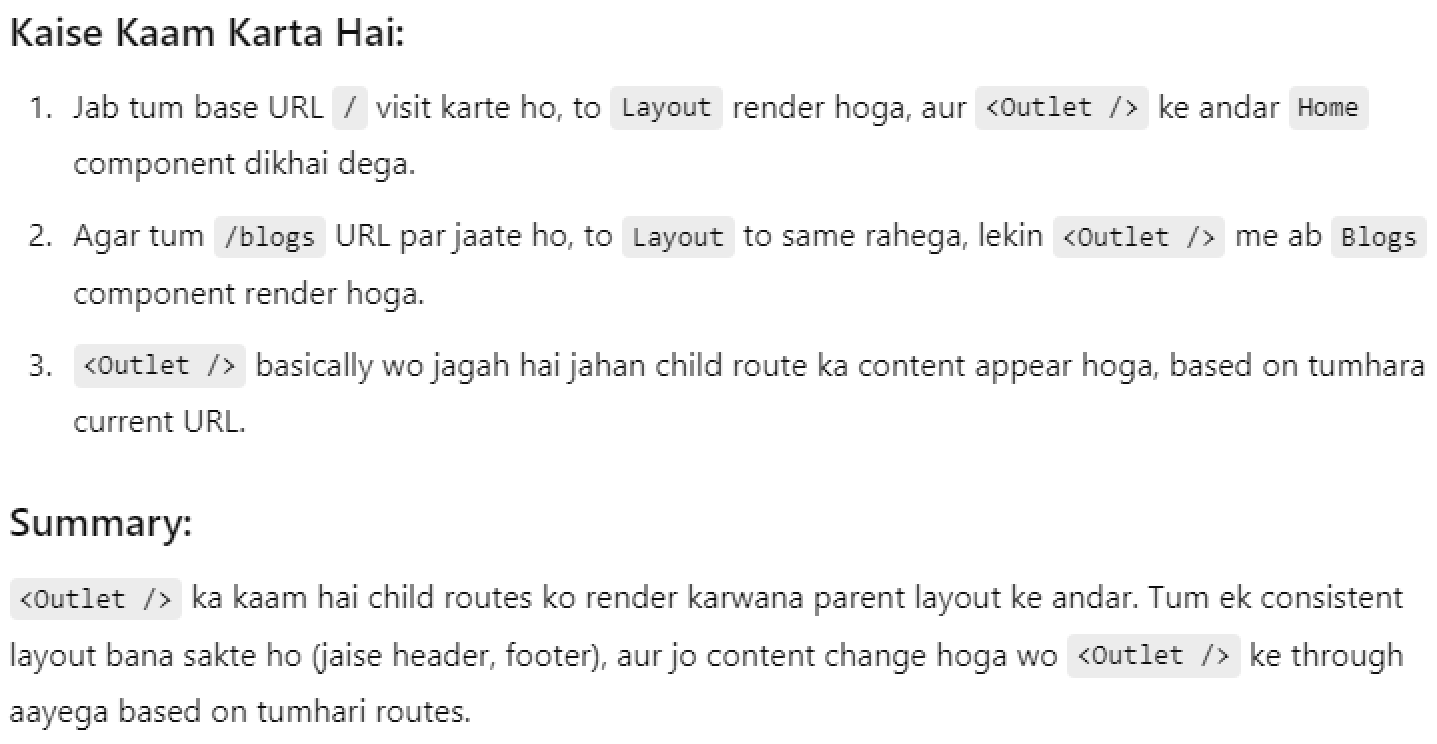
```javascript
<BrowserRouter>
  <Routes>
    <Route path="/" element={<Layout />}>
      <Route index element={<Home />} />
      <Route path="blogs" element={<Blogs />} />
      <Route path="contact" element={<Contact />} />
      <Route path="*" element={<NoPage />} />
    </Route>
  </Routes>
</BrowserRouter>
```

Isme tumne `Layout` component ko parent banaya hai, aur baki routes (`Home`, `Blogs`, `Contact`) iske child hain.



Agar `Layout` component aesa ho:

```javascript
import { Outlet } from 'react-router-dom';

const Layout = () => {
  return (
    <div>
      <header>Header Area</header>
      <main>
        <Outlet /> {/* Yahan child route ka component render hoga */}
      </main>
      <footer>Footer Area</footer>
    </div>
  );
}

export default Layout;
```



Kaise Kaam Karta Hai:

1. Jab tum base URL `/` visit karte ho, to `Layout` render hoga, aur `<Outlet />` ke andar `Home` component dikhai dega.
2. Agar tum `/blogs` URL par jaate ho, to `Layout` to same rahega, lekin `<Outlet />` me ab `Blogs` component render hoga.
3. `<Outlet />` basically wo jagah hai jahan child route ka content appear hoga, based on tumhara current URL.

Summary:

`<Outlet />` ka kaam hai child routes ko render karwana parent layout ke andar. Tum ek consistent layout bana sakte ho (jaise header, footer), aur jo content change hoga wo `<Outlet />` ke through aayega based on tumhari routes.

Outlet wo compt hoga jo us URL me / k bad aye ga. Means ab sirf wo main content ko change kre ga based on URL, baki sab Layout conpt ka content render hoga.

Matlab ap keh skty ho k ==➔ props.children = <Outlet/>

****