**ReactJS-HOL 12**

**Questions & Answers –**

1. **Explain about conditional rendering in React**

**Ans -** Conditional rendering in React means displaying different UI elements or components based on certain conditions. It works the same way conditions work in JavaScript. React uses if-else, ternary operators, or logical && to decide what to render.

**Example -** {isLoggedIn ? <p>Welcome back!</p> : <p>Please log in.</p>}

**Common use cases:**

* Showing a loading spinner before content loads.
* Displaying different content for logged-in vs. guest users.
* Toggling components on/off.

1. **Define element variables**

**Ans -** Element variables in React are variables used to store JSX elements. They allow us to conditionally assign UI elements and use them in the render() method or return block of a component.

**Example –**

let message;

if (isLoggedIn) {

message = <h1>Welcome!</h1>;

} else {

message = <h1>Please sign in</h1>;

}

return <div>{message}</div>;

1. **Explain how to prevent components from rendering**

**Ans -** In React, you can prevent a component from rendering in multiple ways:

**a. Using Conditional Rendering:**

Don’t render the component if the condition is false.

**Example -** {showComponent && <MyComponent />}

**b. Returning null from a Component:**

This tells React not to render anything.

**Example –**

function MyComponent(props) {

if (!props.visible) {

return null;

}

return <div>This is visible</div>;

}

**c. Using shouldComponentUpdate() in Class Components:**

It can return false to prevent unnecessary re-renders.

**Hands-On – (Code)**

*LoginButton.js –*

const LoginButton = (props) => {

    return(

        <div>

            <button onClick={props.onClick}>Login</button>

        </div>

    )

}

export default LoginButton;

*LogoutButton.js –*

const LogoutButton = (props) => {

    return(

        <div>

            <button onClick={props.onClick}>Logout</button>

        </div>

    )

}

export default LogoutButton;

*UserGreeting.js –*

import LogoutButton from "./LogoutButton"

const UserGreeting = ({onLogoutClick}) => {

    return(

            <div

                style={{

                display: "flex",

                flexDirection: "column",

                alignItems: "center",

                justifyContent: "center",

            }}

            >

            <h1>Welcome back</h1>

            <LogoutButton onClick={onLogoutClick}/>

        </div>

    )

}

export default UserGreeting;

*Greetings.js –*

import GuestGreeting from "./GuestGreeting";

import UserGreeting from "./UserGreeting";

const Greetings = ({ isLoggedIn, onLoginClick, onLogoutClick }) => {

        if(isLoggedIn){

            return <UserGreeting onLogoutClick={onLogoutClick}/>;

        }

        return <GuestGreeting onLoginClick={onLoginClick}/>;

}

export default Greetings;

*GuestGreeting.js –*

import LoginButton from "./LoginButton"

const GuestGreeting = ({onLoginClick}) => {

    return(

        <div

            style={{

            display: "flex",

            flexDirection: "column",

            alignItems: "center",

           justifyContent: "center",

            }}

            >

            <h1>Please sign up</h1>

            <LoginButton onClick={onLoginClick}/>

        </div>

    )

}

export default GuestGreeting;

*App.js –*

import './App.css';

import Greetings from './components/Greetings';

import {useState} from 'react';

function App() {

  const [isLoggedIn, setIsLoggedIn] = useState(false);

  const handleLoginClick = () => {

    setIsLoggedIn(true);

  };

  const handleLogoutClick = () => {

    setIsLoggedIn(false);

  };

  return (

    <Greetings

    isLoggedIn={isLoggedIn}

    onLoginClick={handleLoginClick}

    onLogoutClick={handleLogoutClick}

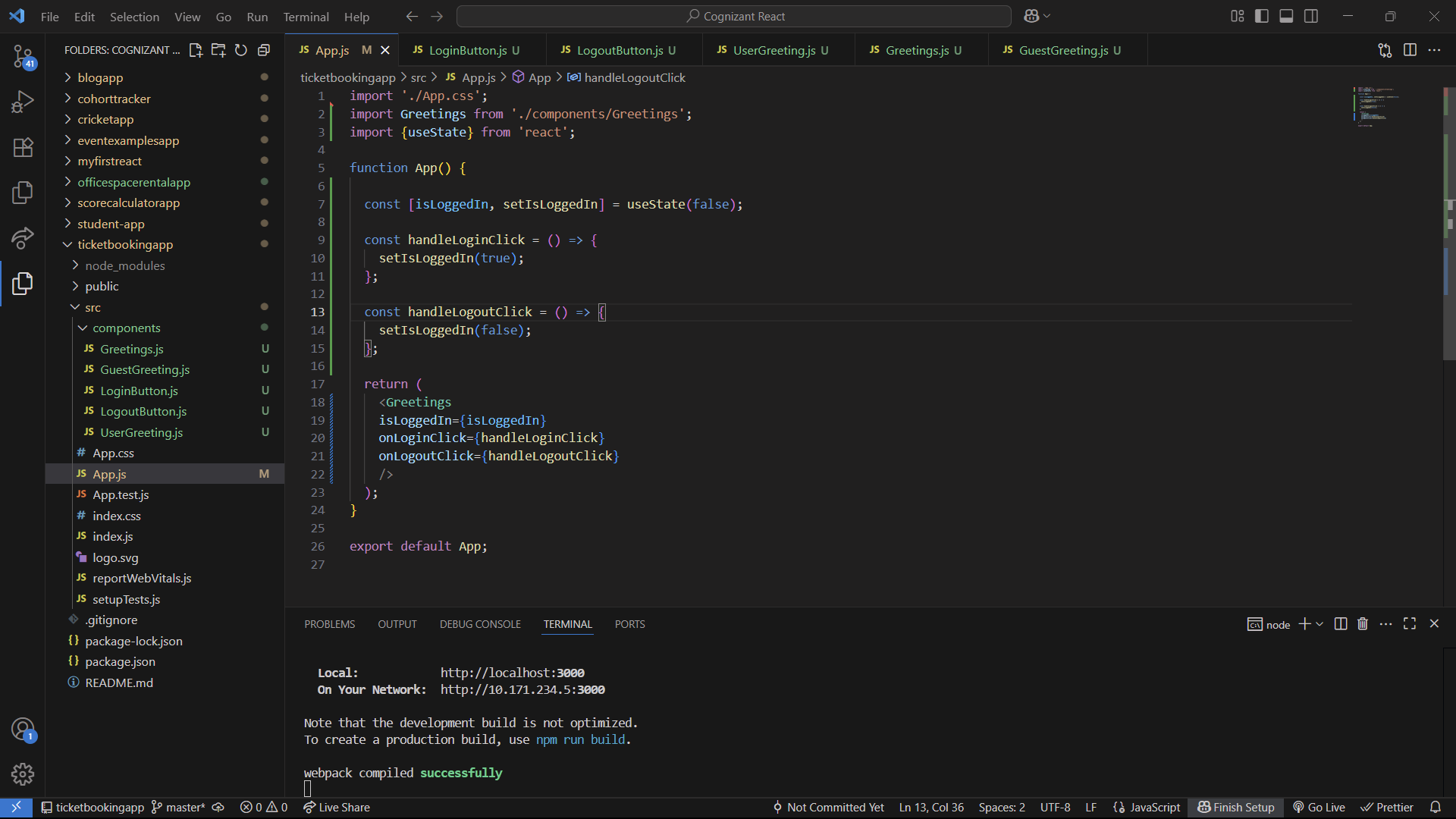
    />

  );

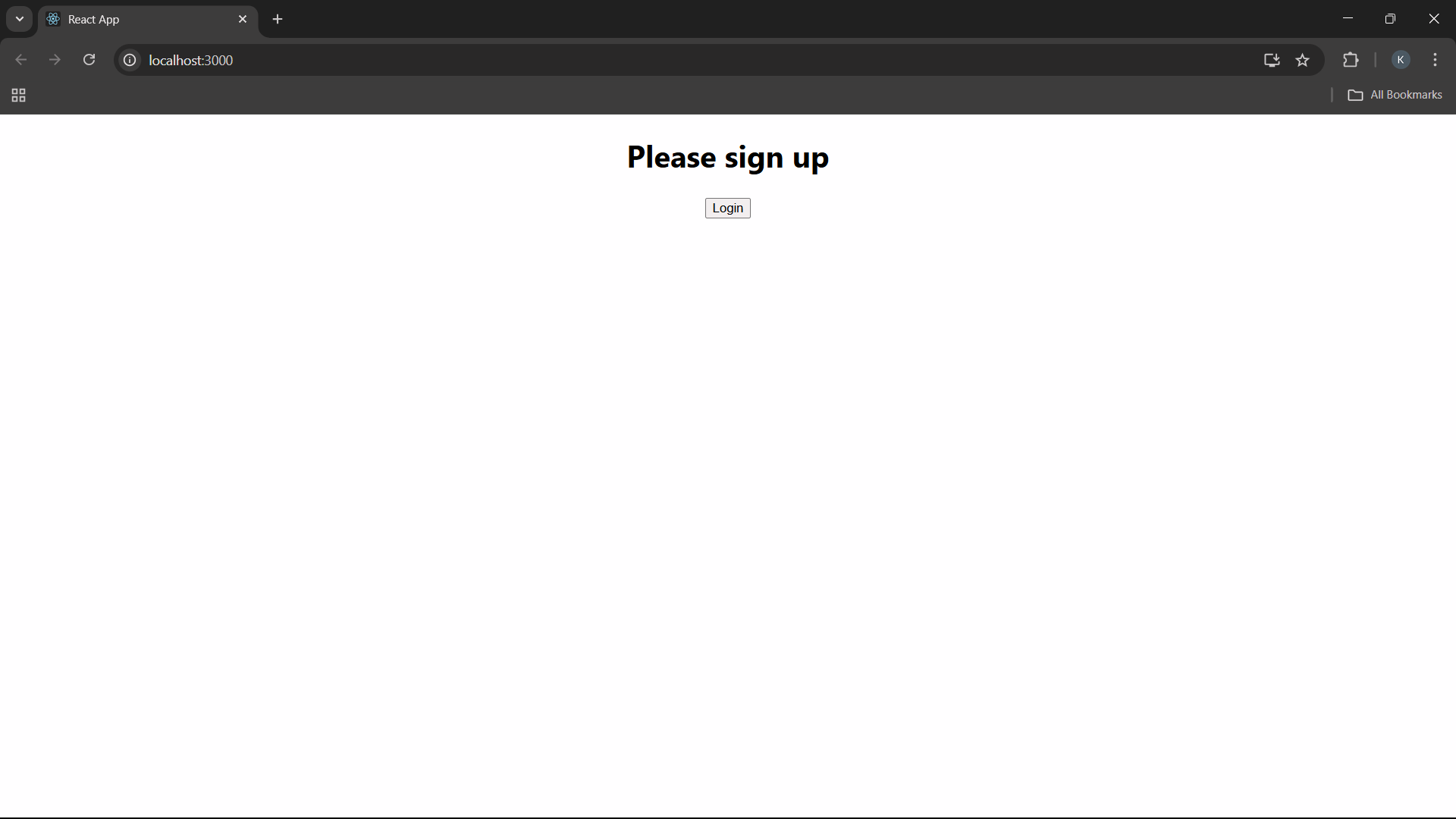
}

export default App;

**Output –**



Guest Greetings Page –



User Greetings Page –

