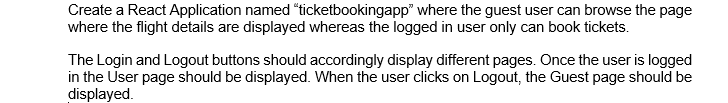
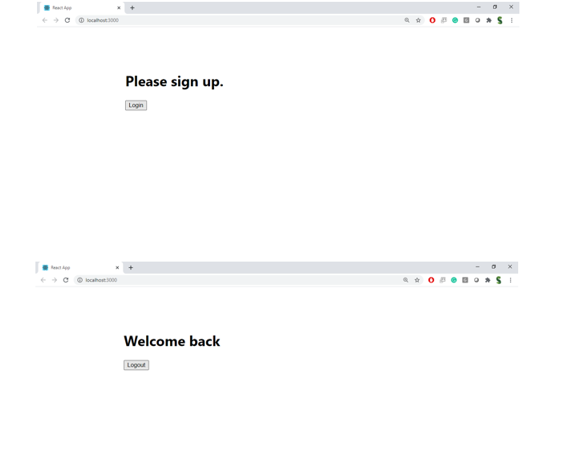
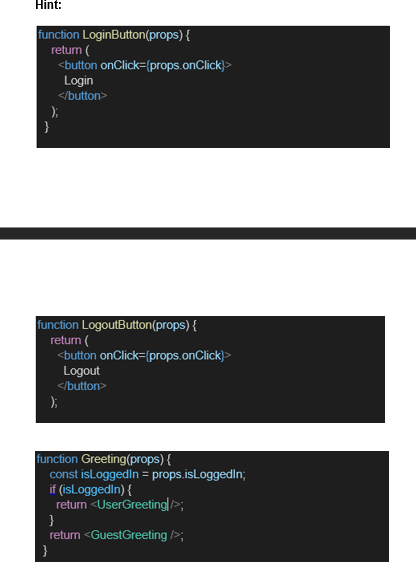


* Conditional rendering in React is a technique that allows us to render different elements or components based on certain conditions or states. It provides a way to control what is displayed on the user interface, making it dynamic and responsive to data changes. For example, we can use conditional rendering to show or hide a loading spinner, display a different component based on a user's login status, or render a list of items only when the list is not empty.
* An element variable is a JavaScript variable that holds a React element. This allows us to store and reuse an element throughout our component's render method, which can be useful for conditionally rendering different parts of the UI. Instead of creating the JSX directly inside the return statement, we can assign it to a variable and then use the variable within the JSX, making our code cleaner and more readable.
* To prevent a component from rendering in React, we can return null from its render method. When a component's render method returns null, React skips the rendering process for that component and does not update the DOM. This can be useful when we want to conditionally render a component based on a specific state or prop, effectively hiding it from the user interface without unmounting it completely. The component's lifecycle methods will still be triggered, but its output will not be displayed.







**Solution**

**FlightList.js**

**GuestPage.js**

import React from 'react';

import FlightList from './FlightList';

const GuestPage = () => (

<div>

<h1>Welcome Guest</h1>

<FlightList />

</div>

);

export default GuestPage;

**UserPage.js**

import React from 'react';

import FlightList from './FlightList';

const UserPage = () => (

<div>

<h1>Welcome User</h1>

<FlightList />

<button>Book Ticket</button>

</div>

);

export default UserPage;

**LoginButton.js**

import React from 'react';

const LoginButton = (props) => (

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

);

export default LoginButton;

**LogoutButton.js**

import React from 'react';

const LogoutButton = (props) => (

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

);

export default LogoutButton;

**Greeting.js**

import React from 'react';

import GuestPage from './GuestPage';

import UserPage from './UserPage';

const Greeting = ({ isLoggedIn }) => {

return isLoggedIn ? <UserPage /> : <GuestPage />;

};

export default Greeting;

**App.js**

import React, { useState } from 'react';

import './App.css';

import Greeting from './Greeting';

import LoginButton from './LoginButton';

import LogoutButton from './LogoutButton';

function App() {

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

const handleLoginClick = () => setIsLoggedIn(true);

const handleLogoutClick = () => setIsLoggedIn(false);

const button = isLoggedIn ? (

<LogoutButton onClick={handleLogoutClick} />

) : (

<LoginButton onClick={handleLoginClick} />

);

return (

<div className="App">

<Greeting isLoggedIn={isLoggedIn} />

{button}

</div>

);

}

export default App;

**Output**

