**REACT**

Create a React Application named “ticketbookingapp” where the guest user can browse the page where the flight details are displayed whereas the logged in user only can book tickets.

The Login and Logout buttons should accordingly display different pages. Once the user is logged in the User page should be displayed. When the user clicks on Logout, the Guest page should be displayed.

App.js

import React, { useState } from 'react';

import './App.css';

function LoginButton(props) {

  return (

    <button onClick={props.onClick}>

      Login

    </button>

  );

}

function LogoutButton(props) {

  return (

    <button onClick={props.onClick}>

      Logout

    </button>

  );

}

function GuestGreeting() {

  return (

    <div>

      <h2>Welcome Guest!</h2>

      <p>Please login to book tickets.</p>

      <FlightList />

    </div>

  );

}

function UserGreeting() {

  return (

    <div>

      <h2>Welcome User!</h2>

      <FlightList />

      <button onClick={() => alert('Ticket Booked!')}>Book Ticket</button>

    </div>

  );

}

function Greeting(props) {

  const isLoggedIn = props.isLoggedIn;

  if (isLoggedIn) {

    return <UserGreeting />;

  }

  return <GuestGreeting />;

}

function FlightList() {

  const flights = [

    { id: 1, from: 'Delhi', to: 'Mumbai', time: '10:00 AM' },

    { id: 2, from: 'Hyderabad', to: 'Bangalore', time: '1:00 PM' },

    { id: 3, from: 'Chennai', to: 'Kolkata', time: '6:00 PM' }

  ];

  return (

    <div>

      <h3>Flights:</h3>

      <ul>

        {flights.map(flight => (

          <li key={flight.id}>

            {flight.from} → {flight.to} at {flight.time}

          </li>

        ))}

      </ul>

    </div>

  );

}

function App() {

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

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

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

  return (

    <div className="App">

      <h1>Ticket Booking App</h1>

      {isLoggedIn ? (

        <LogoutButton onClick={handleLogoutClick} />

      ) : (

        <LoginButton onClick={handleLoginClick} />

      )}

      <Greeting isLoggedIn={isLoggedIn} />

    </div>

  );

}

export default App;

App.css

.App {

  text-align: center;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  min-height: 100vh;

  font-family: Arial, sans-serif;

  background-color: #f5f5f5;

}

button {

  padding: 10px 20px;

  margin: 15px;

  background-color: #007bff;

  color: white;

  border: none;

  border-radius: 5px;

  cursor: pointer;

}

button:hover {

  background-color: #0056b3;

}

ul {

  list-style: none;

  padding: 0;

}

OUTPUT



