**Name: Amirtha A  
Superset id: 6394120**

**12. ReactJS-HOL**

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.

**Hint:**

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

**Code:**

**App.js:**

import React, { useState } from 'react';

import './App.css';

// Guest Greeting Component

function GuestGreeting() {

  return (

    <div>

      <h2>Welcome Guest!</h2>

      <p>You can browse flights, but please login to book tickets.</p>

      <FlightList />

    </div>

  );

}

// User Greeting Component

function UserGreeting() {

  return (

    <div>

      <h2>Welcome Back, User!</h2>

      <p>You can now book your flights.</p>

      <FlightList canBook={true} />

    </div>

  );

}

// Greeting Component (Decides what to show based on isLoggedIn)

function Greeting(props) {

  const isLoggedIn = props.isLoggedIn;

  if (isLoggedIn) {

    return <UserGreeting />;

  }

  return <GuestGreeting />;

}

// Login Button

function LoginButton(props) {

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

}

// Logout Button

function LogoutButton(props) {

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

}

// FlightList Component

function FlightList({ canBook = false }) {

  const flights = [

    { id: 1, origin: "Chennai", destination: "Delhi", time: "10:00 AM" },

    { id: 2, origin: "Mumbai", destination: "Bangalore", time: "1:30 PM" },

    { id: 3, origin: "Kolkata", destination: "Hyderabad", time: "5:45 PM" },

  ];

  const handleBooking = (flight) => {

    alert(`Ticket booked from ${flight.origin} to ${flight.destination}!`);

  };

  return (

    <div>

      <h3>Available Flights:</h3>

      <table border="1" cellPadding="10" align="center">

        <thead>

          <tr>

            <th>Origin</th>

            <th>Destination</th>

            <th>Time</th>

            {canBook && <th>Action</th>}

          </tr>

        </thead>

        <tbody>

          {flights.map((flight) => (

            <tr key={flight.id}>

              <td>{flight.origin}</td>

              <td>{flight.destination}</td>

              <td>{flight.time}</td>

              {canBook && (

                <td>

                  <button onClick={() => handleBooking(flight)}>Book</button>

                </td>

              )}

            </tr>

          ))}

        </tbody>

      </table>

    </div>

  );

}

// Main App Component

function App() {

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

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

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

  return (

    <div className="App">

      <h1>Flight Ticket Booking App</h1>

      {isLoggedIn ? (

        <LogoutButton onClick={handleLogoutClick} />

      ) : (

        <LoginButton onClick={handleLoginClick} />

      )}

      <Greeting isLoggedIn={isLoggedIn} />

    </div>

  );

}

export default App;

A screenshot of a computer screen

AI-generated content may be incorrect.

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.