React Router DOM Project

Basic Setup

1. Install React Router DOM:

Task: Set up React Router DOM in your project.
Guidance: Use npm or yarn to install `react-router-dom`. Explain the steps involved in the installation.

2. Create Routes:

Task: Define basic routes for your application.
Guidance: Use the `BrowserRouter` component to wrap your application and create different routes using the `Route` component.

Page Components

3. Info Page:

Task: Create an Info Page component with student information.
Guidance: Use the `Route` component to render the Info Page when the path matches "/info."

4. Animal Page:

Task: Implement an Animal Page component with a favorite animal display.
Guidance: Utilize dynamic route parameters to display different animal information. Render this page when the path matches "/animal/:animalName."

5. Contact Page:

Task: Develop a Contact Page component with a static form.
Guidance: Render this page when the path matches "/contact." The form doesn't need submission functionality.

Navigation

6. Implement Navigation Links:

Task: Add navigation links between pages.
Guidance: Use the `Link` component to create navigation links. Include a link to the Animal Page from the Info Page and a link to the Contact Page from the Animal Page.

7. Dynamic Navigation:

Task: Implement dynamic navigation using buttons.
Guidance: Add buttons on the Info and Animal pages to navigate to other pages. Use the `useHistory` hook for programmatic navigation.

Advanced Routing

8. Nested Routes:

Task: Create nested routes for additional content.
Guidance: Design a nested route structure where additional content is displayed when navigating to a specific section of a page (e.g., "/animal/:animalName/details").

9. 404 Page:

Task: Implement a custom 404 page for undefined routes.
Guidance: Use the `Switch` component to wrap your routes and include a default route that renders a custom 404 page when no other routes match.