

## React JS Assignment – Quiz

### Multiple Choice Questions (MCQs) React JS Assignment – Quiz

1. Which library is commonly used for navigation in React applications?

- a) React Core
- b) React Router DOM
- c) Axios
- d) Redux

**Ans: b) React Router DOM**

2. What is the purpose of ?

- a) Handles API requests
- b) Enables routing and keeps UI in sync with URL
- c) Stores component state
- d) Loads external scripts

**Ans: b) Enables routing and keeps UI in sync with URL**

3. Which hook is used to navigate programmatically in React Router v6? a) useParams

- b) useLocation
- c) useNavigate
- d) useEffect

**Ans: c) useNavigate**

4. Which component is used to define individual routes?

**Ans: c) <Route>**

5. What does do differently from ?

- a) Allows nested routes
- b) Applies an “active” class to the current link
- c) Prevents navigation
- d) Adds dynamic parameters

**Ans: b) Applies an “active” class to the current link**

6. How do you handle undefined routes in React Router DOM?

- a) Using route
- b) Using <Route path="\*">
- c) Using component
- d) Using block

**Ans: b) Using <Route path="\*">**

7. What hook helps you access URL parameters like /user/:id?

- a) useRoute
- b) useParam
- c) useParams
- d) useLocation

**Ans: c) useParams**

8. What is the correct way to nest routes inside a parent route?

- a) Using multiple components
- b) By placing elements inside another
- c) Using multiple App() components
- d) Declaring routes inside a .json file

**Ans: b) By placing elements inside another**

9. What does the useLocation() hook return?

- a) Current route parameters
- b) The route object of the previous page
- c) Information about the current URL (pathname, search, hash)
- d) Navigation history

**Ans: c) Information about the current URL (pathname, search, hash)**

10. What is one advantage of client-side routing in React apps?

- a) Pages reload faster
- b) No need for any HTML
- c) Navigation occurs without full page reload
- d) Better server-side SEO by default

**Ans: c) Navigation occurs without full page reload**

## Short Answer Questions

1. **What is the difference between <Link> and <NavLink> in React Router?**

**Ans:** <Link> is used to navigate between pages without reloading. <NavLink> does the same but also adds a special class (like active) when the link matches the current URL. That active class is useful to style the currently selected link (for example highlight the active menu item).

2. **Explain the purpose of the useNavigate() hook.**

**Ans:** useNavigate() gives you a function to change the route in code. You can use it to go to another page when something happens (like after a form submit).

**Example:**

```
const navigate = useNavigate();
navigate('/home'); // go to /home
navigate(-1); // go back one page
```

**3. What are dynamic routes in React Router? Give an example.**

**Ans:** Dynamic routes are routes that include parameters which change at runtime. They let one route handle many similar pages.

**Example:** /user/:id — here :id is dynamic, so /user/1 and /user/42 both match the same route and you can read the id with useParams().

**4. How can you handle undefined or invalid routes in React?**

**Ans:** Add a catch-all route like <Route path="\*"> at the end of your route list and render a NotFound or 404 component. This shows a friendly message when the user types a URL that doesn't match any defined routes.

**5. What is the use of the useParams() hook?**

**Ans:** useParams() returns an object with the route parameters from the URL. If the route is /post/:postId, useParams() will return { postId: '123' } when the URL is /post/123. You use it to fetch data or show the right content for that parameter.