AeroAspire -SDE Intern Training

John Nikhil G

Week2-Day4-Oct3

Questions:

1. Describe how client-side routing works (history API or hash routing).

a. History API

- React Router relies on the browser's History API to manage the URL and navigation history.
- This allows the URL to change without reloading the entire page, so users can navigate back and forth smoothly.

b. Hash routing

- In hash routing, navigation happens entirely on the client side.
- It uses window.location.hash to determine the current view, so the page never reloads.

---

2. What happens when you navigate: how React Router matches routes and renders components.

- When the URL changes, React Router detects the new location.
- It looks for a route that matches the current URL path.
- Once a match is found, the corresponding component is rendered.
- Nested routes allow both parent and child components to appear simultaneously on the page.

---

3. How to pass params or query params; nested routes.

- **Path parameters:** Defined like /task/:id, and can be accessed in components using hooks such as useParams.
- **Query parameters:** Defined like /task?id=999, and can be read using useSearchParams.

- Nested routes help display the parent and child components together without replacing the parent.

---

4. What is the flow: writing to localStorage → reading on app startup?

- **Writing:** After updating a state, we can save it to localStorage using localStorage.setItem(key, value).
- **Reading:** When the app starts, we read the saved data using localStorage.getItem(key).
- **Initializing state:** On startup, the state is set from localStorage if available; otherwise, default values are used.

---

5. How do you sync state with localStorage safely (e.g., updates, JSON parse/stringify)?

- Convert objects or arrays to strings before saving using JSON.stringify.
- When reading, parse the data back using JSON.parse and handle any potential errors.
- Keep the state in sync with localStorage by updating it whenever the state changes, typically using hooks or effects.

---

6. What performance / size concerns with storing too much in localStorage?

- Most browsers limit localStorage to about 5MB.
- Storing large amounts of data can slow down reading and writing operations.
- Exceeding the limit may cause errors.
- Sensitive information or big datasets should be avoided in localStorage.