Refs. Portal. React Patterns. Routing

Alen Murtić

sofascore

Sofascore

01

Refs

Refs

- React is declarative and abstracts DOM manipulations

- Refs are used to get reference on the underlying DOM element

    - Access DOM element properties (e.g. height, position, …)

- Official docs

- Returns object with current key set to DOM object or null

- useRef hook - const ref = useRef(null)

- createRef method - this.ref = createRef(null)

- Refs are also used to store data which should not trigger re-renders

Sofascore

Refs example - refs.md

02

React Portal

React Portal

- [Official docs](#)

- Render children outside of DOM hierarchy

    - Used with modals, dialogs, notifications, toasts

- Commonly used with Singleton pattern - e.g. Toast notifications

- Also used for expanding menus of elements whose containers have hidden overflows

- ReactDOM specific

- createPortal(child, container)

- Event propagation and Context work as if it is still in the hierarchy

Sofascore

Portal example - portal.md

03

Controlled components

Controlled components
Controlled components

- Components which are controlled by the parent via props

    - They receive value and onChange handler

- Often used in forms to allow the parent to always know current input values

    - Like you did in Input.tsx in your 5th homework

- Example: Tabs component

    - Controlled: receive current value and onChange/onClick as props

    - Uncontrolled: internally stores state and handles changed, only receives list of Tab

      instances and TabPanel instances as children (handled via context probably)

- Other React patterns: High order components and Render prop

    - Mostly made obsolete by custom hooks 📈

Sofascore

04

Routing

Routing
Routing

- URL addresses specific resource on the Internet (page, response, …)

- Makes user navigation easier (refresh, browser back)

- Process of navigating to the specific resource on the web

SEO (Search Engine Optimization)

- Process of making a website more visible in search results
- Search engines crawl web, index content from pages, points to relevant pages in search results
- Ranking algorithm is secret, known only by search engine companies.

Sofascore

Routing
SEO (Search Engine Optimization)

- More traffic -> More 💰

sofascore

Routing
Server side routing

- Route transition is handled on the server

    - When the URL changes, a new HTML document is retrieved from the server

- Browser has to communicate with the server

- Good:

    - Minimal data for each page

    - Search engine friendly

- Bad:

    - Slower interaction between pages

    - Full refresh of a page -> Context is lost

- Does not necessarily imply server-side rendering

Sofascore

Routing

Client side routing

- Route transition is handled on the client

    - When URL changes, new HTML is not needed, current one is changed

- Good:

    - Faster page transitions

    - Preserves context

    - Page transitions can be animated

- Bad:

    - Larger first load

    - Possible extra data

    - Not so friendly to search engines - they have to render the page like a true browser

- First load can be server-side rendered, but it isn't default in React

Sofascore

Thank you for your attention!

Sofascore