

AeroAspire -SDE Intern Training

John Nikhil G

Week2/Day2(Sept 30)

Questions:

1. Explain how props are passed from parent to child; what happens if props change?

- a. Props (short for "properties") are the mechanism used to transfer data from a parent component to its child components in React.
 - b. The child component accesses the props as arguments, allowing it to use the passed-in data.
 - c. When a prop value is modified in the parent, React **automatically triggers a re-render** of the child component to reflect the updated values on the UI.
-

2. What is the Virtual DOM in React, and how does re-rendering work when props change?

- a. The Virtual DOM is an in-memory representation of the actual browser DOM — it's much faster to work with.
 - b. React leverages this virtual DOM to optimize rendering performance by minimizing direct manipulation of the real DOM.
 - c. When props are updated, React builds a new version of the virtual DOM that represents the latest UI state.
 - d. React then performs a **diffing algorithm** to compare the old virtual DOM with the new one and detect what has changed.
 - e. Only the differences (or "patches") are then applied to the real DOM — this makes rendering efficient and smooth.
-

3. How can unnecessary re-renders be prevented in React?

- a. The `React.memo` function (or the `useMemo` hook) allows functional components to skip re-rendering unless their props change in a meaningful way.
- b. In class-based components, `PureComponent` performs a **shallow comparison** of props and state to determine whether a re-render is necessary.
- c. Avoiding inline object and array declarations in JSX prevents unnecessary re-renders, as new references are created on each render.
- d. Providing consistent and unique `key` props when rendering lists helps React efficiently update only the items that have changed, avoiding redundant renders.

