

❁ React Interview Questions and Answers (1–50)

1. What is React?

Answer: React is a JavaScript library developed by Facebook for building user interfaces, especially single-page applications.

2. What are the features of React?

Answer: JSX, Components, Virtual DOM, One-way Data Binding, Unidirectional Flow.

3. What is JSX?

Answer: JSX stands for JavaScript XML. It allows writing HTML inside JavaScript.

4. What is a component in React?

Answer: A component is a reusable piece of UI. It can be a function or class.

5. What is the difference between functional and class components?

Answer:

- Functional: Uses functions and Hooks
- Class: Uses ES6 classes and lifecycle methods

6. How do you create a functional component?

Answer:

```
function Greeting() {  
  return <h1>Hello</h1>;  
}
```

7. What is state in React?

Answer: State is an object that holds dynamic data and controls component behavior.

8. How do you create state in a functional component?

Answer: Using `useState` Hook:

```
const [count, setCount] = useState(0);
```

9. What is a prop in React?

Answer: Props (short for properties) are inputs passed from parent to child components.

10. How do you pass props to a component?

Answer:

```
<Greeting name="John" />
```

11. What is the difference between state and props?

Answer:

- Props: Passed from parent, read-only
- State: Managed inside the component

12. What is the Virtual DOM?

Answer: A lightweight copy of the real DOM that React uses to optimize updates.

13. How does React update the DOM?

Answer: It compares the Virtual DOM with the previous one and updates only the changed parts.

14. What is a key in React?

Answer: A unique identifier for list elements to help React optimize rendering.

15. What is a Hook in React?

Answer: Functions that let you use React features (like state, lifecycle) in functional components.

16. What are some common React Hooks?

Answer: `useState`, `useEffect`, `useContext`, `useRef`, `useReducer`.

17. What is `useEffect` used for?

Answer: It runs side effects like data fetching, DOM updates, etc.

18. What is the difference between `useEffect` and `useLayoutEffect`?

Answer:

- `useEffect`: Runs after rendering
- `useLayoutEffect`: Runs before the browser paints

19. What is conditional rendering?

Answer: Displaying content based on conditions using if-else or ternary operators.

20. What are fragments in React?

Answer: A wrapper (`<></>`) that lets you group elements without adding extra nodes to the DOM.

21. What is the use of `useRef`?

Answer: Access or reference DOM elements or persist values across renders.

22. What is controlled vs uncontrolled components?

Answer:

- Controlled: Form data is handled by state
- Uncontrolled: Uses refs to get form values

23. How do you handle forms in React?

Answer: Using `useState` to control inputs and `onChange` to update values.

24. What is context in React?

Answer: Allows sharing values between components without prop drilling.

25. How do you use `useContext`?

Answer:

```
const value = useContext(MyContext);
```

26. What is prop drilling?

Answer: Passing props down multiple levels unnecessarily.

27. What is lifting state up?

Answer: Moving state to a common parent to share it between components.

28. What is memoization in React?

Answer: Optimizing performance by caching component renders using `React.memo` or `useMemo`.

29. What is `React.memo`?

Answer: A higher-order component that prevents unnecessary re-renders.

30. What is the use of `useMemo`?

Answer: Memoizes the result of a computation to avoid recalculating on every render.

31. What is `useCallback`?

Answer: Memoizes functions so they don't change between renders.

32. What are higher-order components (HOC)?

Answer: Functions that take a component and return a new component with additional functionality.

33. What is a Pure Component?

Answer: A component that only re-renders if its props or state change.

34. What are portals in React?

Answer: Renders children into a DOM node outside the parent hierarchy.

35. What is error boundary?

Answer: A class component that catches JavaScript errors in its child component tree.

36. What is React Router?

Answer: A library for routing in React apps (navigating between pages/components).

37. How do you set up routing in React?

Answer: Using `react-router-dom`:

```
<BrowserRouter>
  <Routes>
    <Route path="/" element={<Home />} />
  </Routes>
</BrowserRouter>
```

38. What is the difference between `<Link>` and `<a>`?

Answer: `<Link>` prevents full page reloads and enables SPA navigation.

39. What is a SPA (Single Page Application)?

Answer: An application that loads a single HTML page and updates dynamically without refreshing.

40. How do you fetch data in React?

Answer: Using `fetch`, `axios`, or `useEffect`.

41. What is the difference between `setState` and direct assignment?

Answer: `setState` triggers re-rendering, direct assignment does not.

42. What is reconciliation in React?

Answer: The process of updating the DOM by comparing new and previous Virtual DOM.

43. What is a key advantage of React?

Answer: Reusability, performance via Virtual DOM, and component-based architecture.

44. What is suspense in React?

Answer: A component that lets you “wait” for something like lazy-loaded components.

45. What is lazy loading in React?

Answer: Dynamically loading components using `React.lazy()` to reduce initial load.

46. How do you optimize performance in React apps?

Answer: Code splitting, memoization, avoiding unnecessary renders, using keys, etc.

47. What is hydration in React?

Answer: Process of attaching event handlers to server-rendered HTML in SSR.

48. What are React DevTools?

Answer: A browser extension to inspect the React component hierarchy and props/state.

49. How do you debug a React app?

Answer: Using React DevTools, console logs, breakpoints, and error boundaries.

50. What are the limitations of React?

Answer:

- SEO challenges without SSR
- Steeper learning curve with advanced features
- Requires external libraries for routing and state management