

# Top 40 React Interview Questions & Answers

## 1. What is React?

React is a JavaScript library for building user interfaces, mainly for single-page applications using components.

## 2. What is useMemo?

useMemo is a hook that memoizes a computation result and re-calculates only when its dependencies change.

## 3. What are the features of React?

Features include component-based architecture, virtual DOM, JSX, unidirectional data flow, and more.

## 4. What is JSX?

JSX is a syntax extension that allows writing HTML-like syntax directly within JavaScript code.

## 5. What is DOM?

DOM is the Document Object Model, an API representing the structure of a document.

## 6. What is Virtual DOM?

Virtual DOM is a lightweight copy of the real DOM used by React for efficient re-rendering.

## 7. What is the component lifecycle of a React class component?

It has phases: mounting, updating, and unmounting with methods like `componentDidMount` and `componentWillUnmount`.

## 8. What are fragments in React?

Fragments let you group elements without adding extra nodes to the DOM.

9. What are props in React?

Props are inputs to components that are passed from parent to child components.

10. What are synthetic events in React?

Synthetic events are React's cross-browser wrapper for native events.

11. Difference between package.json and package-lock.json?

package.json records project dependencies, while package-lock.json locks the versions of installed dependencies.

12. Differences between client-side and server-side rendering?

Client-side renders in the browser, while server-side renders HTML on the server.

13. What is state in React?

State is an object that holds data that may change over time and is used to re-render components.

14. What are props?

Props are read-only inputs passed from parent to child components in React.

15. Differences between State and Props in React?

State is mutable, Props are immutable.

16. What is props drilling?

Passing props through multiple components just to reach a deeply nested component.

17. Disadvantages of props drilling and how to avoid it?

Props drilling is difficult to manage; can be avoided using Context API or Redux.

18. What are Pure Components in React?

Pure components re-render only when their props or state changes.

19. What are refs in React?

Refs provide access to DOM elements or React elements in the component.

20. What is meant by forward ref?

Forward ref allows refs to be passed from parent to child components.

21. What are Error Boundaries?

Error boundaries catch JavaScript errors in child components and display fallback UI.

22. What are Higher-Order Components?

A function that takes a component and returns a new enhanced component.

23. Difference between controlled and uncontrolled components?

Controlled components are controlled by React state, uncontrolled components manage their own state with DOM.

24. What is useCallback?

useCallback memoizes a function and only re-creates it if its dependencies change.

25. Differences between useMemo and useCallback?

useMemo memoizes a value, useCallback memoizes a function.

26. What are keys in React?

Keys are unique identifiers for list items to optimize rendering.

27. What is lazy loading in React?

Lazy loading allows components to be loaded only when needed.

28. What is Suspense in React?

Suspense allows you to show a fallback component while waiting for lazy-loaded components.

29. What are custom hooks?

Custom hooks are reusable functions that encapsulate React hook logic.

30. What is useReducer hook?

useReducer is an alternative to useState for managing more complex state logic.

31. What are Portals in React?

Portals allow rendering children into a different part of the DOM.

32. What is context in React?

Context allows you to share state globally without passing props.

33. Example of context API usage?

Use React.createContext to create a context and provide it to nested components.

34. Purpose of callback function in setState?

The callback is called after the state update is applied, allowing further actions.

35. Custom hook for increment/decrement counter?

Example: useCounter hook that returns count, increment, and decrement functions.

36. Lifecycle hooks replaced by useEffect?

componentDidMount, componentDidUpdate, and componentWillUnmount are replaced by useEffect.

37. What is Strict mode in React?

Strict mode runs additional checks in development mode.

38. Ways to pass data from child to parent?

Using callback functions, context API, or state management libraries.

39. Example: Sending data from child to parent using callback?

Pass a function to the child, which it can call with data as an argument.

40. Example: Sending data from child to parent using useRef?

Pass a ref to the parent to access data in the child's state or DOM element.