

React.js Interview Questions & Answers

1. What is React.js?

React.js is an open-source JavaScript library developed by Facebook for building user interfaces, especially single-page applications.

2. What are the main features of React?

Declarative, Component-based, Virtual DOM, Unidirectional data flow, JSX support.

3. What is JSX?

JSX (JavaScript XML) is a syntax extension that allows writing HTML-like code inside JavaScript.

4. What is the difference between Virtual DOM and Real DOM?

Virtual DOM is a lightweight copy of the real DOM. React updates the Virtual DOM first, then efficiently updates the real DOM.

5. What are React components?

Components are independent and reusable UI blocks in React. They can be functional or class-based.

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

Functional components are stateless and use hooks, while class components have state and lifecycle methods.

7. What are props in React?

Props (properties) are read-only inputs passed to components to make them dynamic.

8. What is state in React?

State is a built-in object that stores component data and can change over time.

9. What is the difference between state and props?

Props are read-only and passed from parent to child. State is mutable and managed within the component.

10. What is lifting state up in React?

Lifting state up means moving shared state to the closest common ancestor so multiple components can access it.

11. What is React Hooks?

Hooks are functions that let functional components use state and lifecycle features.

12. What are useState and useEffect hooks?

useState is used to manage state, while useEffect is used to handle side effects like data fetching.

13. What is the difference between useEffect and componentDidMount?

useEffect runs after every render by default, while componentDidMount runs once after initial render in class components.

14. What is useContext hook?

useContext allows access to context values without using props drilling.

15. What is useReducer hook?

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

16. What is useMemo hook?

useMemo is used to memoize expensive computations and avoid recalculating on every render.

17. What is useCallback hook?

useCallback returns a memoized version of a function to prevent unnecessary re-renders.

18. What is React Context API?

Context API allows passing data through the component tree without using props drilling.

19. What is Redux?

Redux is a state management library for JavaScript applications, often used with React.

20. What are the principles of Redux?

Single source of truth, State is read-only, Changes are made using pure functions (reducers).

21. What is the difference between Redux and Context API?

Redux is more powerful and suited for large apps, while Context API is simpler and best for small-scale state sharing.

22. What is a higher-order component (HOC)?

A HOC is a function that takes a component and returns a new component with additional props or behavior.

23. What are controlled and uncontrolled components?

Controlled components rely on state to manage form inputs, while uncontrolled components use refs.

24. What is reconciliation in React?

Reconciliation is the process by which React updates the Virtual DOM and compares it with the real DOM to apply changes efficiently.

25. What is React Fiber?

React Fiber is the new reconciliation algorithm introduced in React 16 to improve rendering performance.

26. What is the difference between React and Angular?

React is a library focused on UI, while Angular is a full-fledged MVC framework.

27. What is the difference between React and Vue?

React is backed by Facebook with unidirectional data flow, Vue is progressive and offers both one-way and two-way binding.

28. What are React fragments?

Fragments let you group multiple elements without adding extra DOM nodes.

29. What are keys in React lists?

Keys help React identify which list items have changed, been added, or removed.

30. What are portals in React?

Portals allow rendering children into a DOM node outside the parent component hierarchy.

31. What are error boundaries?

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

32. What is React Router?

React Router is a library used for navigation and routing in React applications.

33. What is lazy loading in React?

Lazy loading is a technique where components are loaded only when needed to optimize performance.

34. What is suspense in React?

Suspense lets components wait for something (like data fetching) before rendering fallback UI.

35. What is server-side rendering (SSR) in React?

SSR is the process of rendering React components on the server and sending HTML to the client.

36. What is hydration in React?

Hydration is the process where React attaches event handlers to existing server-rendered HTML.

37. What is Next.js?

Next.js is a React framework for SSR, static site generation, and routing.

38. What is the difference between useRef and createRef?

useRef persists the same reference across renders, createRef creates a new ref every time.

39. What is the difference between controlled and uncontrolled inputs?

Controlled inputs are tied to component state, uncontrolled inputs use refs for DOM access.

40. What is prop drilling?

Prop drilling occurs when props are passed through multiple nested components unnecessarily.

41. What are render props?

Render props is a technique where a function is passed as a prop to determine what to render.

42. What is memo in React?

React.memo is a higher-order component that memoizes functional components to prevent unnecessary re-renders.

43. What is PureComponent?

PureComponent is a React class component that implements shouldComponentUpdate with a shallow prop and state comparison.

44. What are synthetic events in React?

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

45. What is the difference between stateful and stateless components?

Stateful components manage state internally, while stateless components do not.

46. What is the difference between shallow rendering and deep rendering?

Shallow rendering tests components without rendering children, deep rendering renders the full component tree.

47. What is the difference between mount and render?

Render describes what to display, mounting is when React actually inserts components into the DOM.

48. What are React dev tools?

React DevTools is a browser extension for inspecting React component hierarchies.

49. What are best practices in React development?

Use functional components with hooks, keep components small, use keys in lists, use memoization for performance.

learnincreation