53 React interview questions to ask developers (with answers)

Questions

1. What is React and why is it used?

2. Can you explain the concept of 'state' in React?

3. What are props in React?

4. What is the difference between a class component and a functional component?

5. How do you handle forms in React?

6. What is JSX and why is it used in React?

7. What are React hooks?

8. What is the purpose of the 'key' prop in a list of React elements?

9. Can you explain how the context API works in React?

10. What are controlled components, and how do they differ from uncontrolled components?

11. How do you optimize performance in a React application?

12. What is the significance of the useEffect hook, and when would you use it?

13. Can you explain what a higher-order component is?

14. What is React Router, and how does it help in building single-page applications?

15. How do you manage global state in a React application?

16. What are the benefits of using TypeScript with React?

17. How would you handle error boundaries in a React application?

18. Can you explain the lifecycle methods in React and their significance?

19. Describe the difference between mounting and updating in React.

20. What is the purpose of the `render` method in React components?

21. How would you explain the concept of 'lifting state up' in React?

22. What is the purpose of default props in React?

23. Can you explain the difference between a controlled and uncontrolled component in React?

24. How would you handle component cleanup in React?

25. What is the significance of keys in React lists?

26. Why is it important to use `setState` instead of modifying the state directly in React?

27. How would you explain the concept of 'one-way data flow' in React?

28. What's the difference between state and props in terms of mutability?

29. Can you describe a scenario where you'd use the useState hook instead of class state?

30. How do you pass data from a child component to its parent?

31. What's the purpose of the useReducer hook and when would you use it over useState?

32. How does prop drilling occur and what are some ways to avoid it?

33. Can you explain the concept of derived state in React?

34. What are the potential pitfalls of using the setState method in class components?

35. How do you handle asynchronous state updates in React?

36. What's the difference between state colocation and lifting state up?

37. How do you decide whether a piece of data should be state or props?

38. Can you explain how to use the useCallback hook to optimize prop passing?

39. What are the benefits and drawbacks of using global state management libraries like Redux?

40. You're working on a React application that's becoming slow. How would you identify and resolve performance bottlenecks?

41. A junior developer on your team is struggling with state management. How would you explain and implement a solution using React's Context API?

42. Your team is debating whether to use Redux or React's built-in state management. How would you approach this decision?

43. You need to integrate a third-party library that doesn't have React bindings. How would you approach this task?

44. A component in your application is re-rendering too often. How would you diagnose and fix this issue?

45. Your team is transitioning from class components to functional components with hooks. How would you plan and execute this refactoring?

46. You're tasked with improving the accessibility of a React application. What steps would you take?

47. How would you implement a custom hook for handling form validation in a React application?

48. You're working on a large-scale React application. How would you structure your components and organize your code?

49. A new team member is unfamiliar with React's synthetic events. How would you explain them and their importance?

50. You need to implement a feature that requires sharing state between unrelated components. How would you approach this?

51. How would you handle authentication in a React application, considering both the UI and API interactions?