React Basics

1. What is ReactJS?
2. Explain the key features of React.
3. What are the differences between React and Angular?
4. Explain the virtual DOM in React.
5. How does React achieve better performance with the virtual DOM?
6. What is JSX? Why is it used in React?
7. How do you create components in React?
8. What are props in React?
9. Explain the concept of state in React.
10. What is the difference between state and props?

Components and Props

11. How do you create functional components in React?
12. How do you create class components in React?
13. What are pure components in React?
14. What are higher-order components (HOC) in React?
15. How do you pass props to components in React?
16. How do you handle props and state changes in React components?
17. Explain the lifecycle methods of React components.
18. What is the significance of `shouldComponentUpdate` method?
19. How do you implement default props in a React component?
20. What are controlled and uncontrolled components in React?

State Management

21. What are the different ways to manage state in React?
22. Explain the `useState` hook in React.
23. What is the `useEffect` hook used for in React?
24. How do you manage global state in React applications?
25. What are the differences between Redux and Context API for state management?
26. Explain the Redux architecture and its core principles.
27. How do you connect Redux with a React application?
28. What is an action creator in Redux?
29. Explain reducers in Redux.
30. How does Redux store work?

Hooks

31. What are hooks in React? Why were they introduced?
32. Explain the `useState` hook in detail.
33. Explain the `useEffect` hook in detail.
34. How do you create custom hooks in React?
35. What are the rules of hooks in React?
36. How do you optimize performance with hooks?

Routing and Navigation

37. How do you implement routing in a React application?
38. Explain the role of `react-router` in React applications.
39. How do you handle dynamic routing in React?
40. What are nested routes in React-router?
41. How do you pass parameters in the URL using React-router?

Forms and Data Handling

42. How do you handle forms in React?
43. Explain controlled and uncontrolled components in form handling.
44. How do you validate form input in React?
45. How do you handle form submission in React?
46. How do you perform asynchronous data fetching in React?
47. How do you manage AJAX requests in React?

Component Lifecycle

48. Explain the lifecycle of a React component.
49. What are the deprecated lifecycle methods in React 16?
50. How do you handle side effects in React components?

Performance Optimization

51. How do you optimize performance in React applications?
52. Explain memoization in React.
53. What are the performance best practices for React applications?
54. How do you profile React applications for performance improvements?

Context API

55. What is Context API in React?
56. How do you create and consume context in React?
57. When would you use Context API over props drilling?

Testing

58. How do you test React components?
59. What are the popular testing libraries for React?
60. How do you perform unit testing on React components?

Styling in React

61. What are the different ways to style components in React?
62. Explain CSS Modules in React.
63. How do you use styled-components in React?

Error Handling

64. How do you handle errors in React applications?
65. What are Error Boundaries in React?

Integrations

66. How do you integrate React with other libraries or frameworks?
67. How do you use React with TypeScript?
68. How do you integrate React with GraphQL?

Best Practices

69. What are the best practices for writing clean and maintainable code in React?
70. How do you structure a React application?

Security

71. What are some security best practices in React applications?
72. How do you prevent XSS attacks in React?

Deployment

73. How do you deploy a React application?
74. What are the considerations for deploying a React application?

Advanced Concepts

75. What are portals in React?
76. How do you handle state management without Redux?
77. What are lazy loading and code splitting in React?
78. How do you handle animations in React?

Real-world Scenarios

79. How do you handle authentication in a React application?
80. How do you handle routing transitions in React?

React Native

81. What is React Native?
82. How does React Native differ from React?
83. How do you create a mobile application using React Native?

Performance Monitoring and Optimization

84. How do you monitor performance in a React application?
85. What tools would you use to identify performance bottlenecks in React?

Accessibility

86. How do you ensure accessibility in React applications?

87. What are ARIA roles in React?

Redux Middleware

88. What is Redux middleware? Give examples.
89. How do you write custom middleware for Redux?

Redux Thunk

90. What is Redux Thunk used for?
91. How do you handle asynchronous actions in Redux with Redux Thunk?

Server-side Rendering (SSR)

92. What is Server-side Rendering (SSR) in React?
93. How do you implement SSR in a React application?

Context and useContext Hook

94. What is the useContext hook in React?
95. How do you use context to pass data through the component tree in React?

useMemo and useCallback Hooks

96. What is the useMemo hook in React and when do you use it?
97. What is the useCallback hook in React and when do you use it?

React Concurrent Mode

98. What is React Concurrent Mode and what problem does it solve?
99. How does React Concurrent Mode affect the rendering pipeline?

React Suspense

100. What is React Suspense and how does it improve the user experience?