**React**

1. What is the difference between state and props in React?
2. What are the advantages of using functional components over class components in React?
3. How does React handle event handling?
4. What is the virtual DOM in React?
5. How does React handle component lifecycle methods?
6. How does React handle component re-rendering and performance optimization?
7. How does React Router work for client-side routing in a React application?
8. How does React handle error boundaries and error handling in components?
9. What is the purpose of Redux in a React application, and when would you use it?
10. How does React handle forms and form validation?
11. Can you explain the concept of higher-order components (HOCs) in React?
12. How does React handle key concepts like reconciliation and the virtual DOM to optimize rendering performance?
13. How does React handle context in component communication and sharing data between components?
14. What are the advantages and disadvantages of using Next.js for server-side rendering (SSR) in React applications?
15. How does React perform reconciliation when rendering components?
16. What are React hooks, and how do they differ from class components?
17. What are the different lifecycle methods in React and when are they invoked?
18. How does React Router work for server-side rendering (SSR) in a React application?
19. Explain the concept of server-side rendering (SSR) and its advantages
20. How does React handle code splitting and lazy loading for optimizing bundle size?
21. What is the purpose of Webpack in a React application, and how does it work?
22. How does React handle handling form inputs and managing their state?
23. What are the different ways to style components in React?
24. How does React handle code reusability and component composition?