1. **Explain the CSS Box Model.**
2. **What is a closure in JavaScript, and why is it important?**
3. **Describe the differences between inline, inline-block, and block-level elements in HTML and their use cases.**
4. **How does React handle state management?**
5. **What are React Hooks, and when would you use them?**
6. **Explain the concept of Virtual DOM and its benefits in React.**
7. **How would you optimize the performance of a React application?**
8. **What are code splitting and lazy loading in the context of React?**
9. **Describe the different types of testing in React. How would you write unit tests for a React component?**
10. **What is Jest, and how is it used in React testing?**
11. **What are some common security vulnerabilities in web applications, and how can you mitigate them in a frontend application?**
12. **Explain the role of Webpack in a modern frontend development workflow.**
13. **How would you configure and optimize a Webpack build for a production application?**
14. **How would you design a scalable frontend architecture for a banking application like J.P. Morgan's?**
15. **What considerations would you keep in mind when designing for high availability?**
16. **Describe your experience integrating with RESTful APIs and GraphQL endpoints.**
17. **How would you handle rate limiting and caching in a frontend application that relies heavily on API calls?**
18. **What coding standards and best practices do you follow to ensure the quality of your frontend code?**
19. **Explain event delegation and why it's useful.**
20. **What is the difference between let, const, and var in JavaScript?**
21. **What is a closure, and how might you use one in JavaScript?**
22. **How do you handle asynchronous operations in JavaScript, and what are Promises?**
23. **What are React components, and how do they differ from HTML elements?**
24. **Explain the concept of props and state in React. When would you use one over the other?**
25. **What is React Router, and why is it useful in single-page applications?**