

Commonly Asked JavaScript Interview Questions

◆ Core JavaScript

1. What is the difference between var, let, and const in JavaScript?
2. What are closures in JavaScript, and how do they work?
3. What is the this keyword in JavaScript, and how does it behave in different contexts?
4. What is a JavaScript Promise, and how does it handle asynchronous code?
5. What is the event loop, and how does JavaScript handle asynchronous operations?
6. What is hoisting in JavaScript, and how does it work?
7. What are JavaScript data types, and how do you check the type of a variable?
8. What is the difference between null and undefined in JavaScript?
9. What is a callback function, and how is it used?
10. How do you manage errors in JavaScript?

◆ Async JavaScript

11. What is the difference between setTimeout() and setInterval()?
12. How do JavaScript Promises work, and what is the then() method?
13. What is async/await, and how does it simplify asynchronous code in JavaScript?
14. What are the advantages of using async functions over callbacks?
15. How do you handle multiple Promises simultaneously?

◆ Advanced JavaScript Features

16. What are higher-order functions in JavaScript, and can you provide an example?
17. What is destructuring in JavaScript, and how is it useful?
18. What are template literals in JavaScript, and how do they work?
19. How does the spread operator work in JavaScript?
20. What is rest parameter in JavaScript, how does it differ from the arguments object?

◆ Objects and Arrays

21. What is the difference between an object and an array in JavaScript?
22. How do you clone an object or array in JavaScript?
23. What are object methods like Object.keys(), Object.values(), and Object.entries()?
24. How does the map() method work in JavaScript, and when would you use it?
25. What is the difference between map() and forEach() in JavaScript?

◆ Special JavaScript Concepts

26. What is event delegation in JavaScript, and why is it useful?
27. What are JavaScript modules, and how do you import/export them?
28. What is the prototype chain in JavaScript, and how does inheritance work?
29. What are bind(), call(), and apply() in JavaScript, and when do you use them?
30. How does JavaScript handle equality comparisons with == and ===?

◆ Frontend JavaScript Concepts

31. What is the Document Object Model (DOM), how does JavaScript interact with it?
32. How do you prevent default actions and stop event propagation in JavaScript?
33. What is the difference between synchronous and asynchronous code in JavaScript?
34. What is the difference between an event object and a custom event in JavaScript?
35. How do you optimize performance in JavaScript applications?