

JavaScript Interview Questions & Answers (Zero → Hero)

Basics

1. What is JavaScript? → A high-level, interpreted language used for browsers & servers (Node.js).
2. Difference between var, let, and const? → var=function scoped, let/const=block scoped.
3. Data types? → Primitive (string, number, boolean, null, undefined, bigint, symbol), Reference (objects, arrays, functions).
4. == vs ===? → == does type coercion, === strict equality.
5. Template literals → Strings with backticks allowing interpolation.

Intermediate

6. Hoisting? → Declarations moved to top of scope before execution.
7. Closures? → Functions remember outer scope variables even after scope ends.
8. Event Loop? → Manages call stack & async tasks.
9. null vs undefined? → undefined=declared not assigned, null=intentional empty.
10. Promises? → Objects for async completion/failure.

Advanced

11. this keyword? → Depends on context (global, function, arrow function).
12. Prototype Inheritance → Objects inherit via prototype chain.
13. call, apply, bind → Methods to control function context.
14. Currying → Converting multi-arg function into chained single-arg functions.
15. Debouncing vs Throttling → Debounce=delay exec, Throttle=limit rate of exec.

Expert

16. Event Delegation → Attach listener to parent, handle via event.target.
17. Generators → Functions that pause/resume using yield.
18. Async/Await → Cleaner syntax for promises.
19. WeakMap/WeakSet → Collections with weakly referenced keys for memory efficiency.
20. Memory Management → Garbage collection removes unreferenced objects.

Security & Performance

21. XSS → Injected malicious scripts. Prevent with sanitization, CSP.

- 22. CSRF → Unauthorized requests. Prevent with CSRF tokens, SameSite cookies.
- 23. JS Performance → Minify, lazy load, avoid memory leaks, async handling.

Latest ECMAScript Features

- 24. Optional Chaining → Safe property access with ?.
- 25. Nullish Coalescing → ?? returns RHS if LHS is null/undefined.
- 26. ES Modules → import/export standard syntax.
- 27. BigInt → For large integers beyond MAX_SAFE_INTEGER.
- 28. Dynamic Import → Import modules on demand.