

# Top 50 JavaScript Interview Questions with Answers

1. What is JavaScript? → A scripting language used to add interactivity and logic to web pages.
2. Is JavaScript the same as Java? → No. They are completely different languages.
3. Where is JavaScript used? → In browsers, servers (Node.js), and apps.
4. How do you add JavaScript to an HTML page? → Using <script> tag or external JS file.
5. Data types: string, number, boolean, undefined, null, object, symbol, bigint.
6. var vs let vs const → var: function-scoped; let/const: block-scoped.
7. Hoisting → JS moves declarations to top before executing.
8. Temporal Dead Zone → Time before let/const initialized.
9. Global vs Local Scope → Global usable everywhere, local inside a block.
10. Reserved keywords → Words you cannot use as variable names.
11. == vs === → == compares values, === compares values and types.
12. Logical operators → &&, ||, !
13. Ternary operator → condition ? trueValue : falseValue
- 14.Truthy/Falsy values → Falsy: 0, "", null, undefined, NaN, false.
15. typeof → Checks the data type.
16. Loops → for, while, do...while, for...in, for...of
17. for...in vs for...of → in: object keys, of: array values.
18. break & continue → break stops loop; continue skips iteration.
19. switch → Used for multiple conditions.
20. Example: switch (grade) { case 'A': console.log('Excellent'); break; }
21. Function → Reusable block of code.
22. Arrow function → const add = (a, b) => a + b;
23. Default parameter → function greet(name = 'User') {}
24. Anonymous function → function() {}
25. IIFE → (function(){}());
26. Array → Ordered list of values.
27. Array methods → push, pop, map, filter, reduce.
28. Object → Key-value pairs.
29. Access object → obj.key or obj['key'].
30. Destructuring → const {name, age} = person;
31. Callback function → Function passed to another function.

32. Promise → Handle async success/failure.
33. async/await → Cleaner promise handling.
34. Closure → Function remembers outer variables.
35. Higher-order function → Function using other function as arg or return.
36. DOM → Document Object Model for HTML structure.
37. Select element → getElementById(), querySelector()
38. Event listener → element.addEventListener('click', fn)
39. Event bubbling → Event moves up from child to parent.
40. stopPropagation() → Stops bubbling.
41. JSON → Text data format for communication.
42. this keyword → Refers to current object context.
43. Template literals → `Hello \${name}`
44. NaN → Not a Number.
45. Deep vs Shallow copy → Shallow: only top layer; Deep: all nested objects.
46. localStorage vs sessionStorage → local: permanent, session: temporary.
47. Cookies vs localStorage → Cookies sent to server; localStorage stays in browser.
48. Sync vs Async → Sync runs line by line; Async runs in background.
49. Event loop → JS mechanism handling async tasks.
50. Modules → export/import for code organization.