JavaScript Interview Questions and Answers (50 Q&A)

1. What is JavaScript?
Answer: JavaScript is a lightweight, interpreted, or just-in-time compiled programming language primarily used for creating interactive web pages.

---

2. What are the main features of JavaScript?
Answer: Lightweight, interpreted, object-oriented, asynchronous, event-driven, cross-platform, and supports functional and prototype-based programming.

---

3. What is the difference between Java and JavaScript?
Answer: Java is a compiled, strongly typed language, while JavaScript is an interpreted, loosely typed scripting language mainly used for web development.

---

4. What are the data types in JavaScript?
Answer: Primitive: string, number, boolean, null, undefined, symbol, bigint.
Non-primitive: objects and arrays.

---

5. What is the difference between null and undefined?
Answer: null is an assigned empty value, while undefined means a variable has been declared but not assigned a value.

---

6. What are objects in JavaScript?
Answer: Objects are collections of key-value pairs, where keys are strings (or symbols) and values can be any type.

---

7. What is an array in JavaScript?
Answer: An array is an ordered list of values stored under a single variable name.

---

8. What is the difference between var, let, and const?
Answer: var has function scope, let and const have block scope. const cannot be reassigned.

---

9. What is hoisting in JavaScript?
Answer: Hoisting moves variable and function declarations to the top of their scope before execution.

---

10. What is a closure?
Answer: A closure is a function that remembers variables from its outer scope even after the outer function has finished executing.

---

11. Difference between == and ===?
Answer: == checks equality with type coercion, while === checks strict equality without type conversion.

---

12. What is NaN?
Answer: NaN stands for "Not-a-Number," representing an invalid number result (e.g., parseInt("abc")).

---

13. What is synchronous vs asynchronous programming?
Answer: Synchronous code runs line by line; asynchronous code does not block execution and can run later (e.g., callbacks, promises).

---

14. What is the difference between setTimeout and setInterval?
Answer: setTimeout runs a function once after a delay, setInterval runs repeatedly at specified intervals.

---

15. What are Promises?
Answer: Promises represent the eventual result (success or failure) of an asynchronous operation.

---

16. What are async/await?
Answer: Async/await is syntactic sugar over promises that allows writing asynchronous code in a synchronous style.

---

17. What is the event loop?
Answer: The event loop manages asynchronous operations by pushing callbacks into the call stack when it is free.

---

18. Difference between call, apply, and bind?
Answer:

- call() invokes a function with given arguments.
- apply() invokes with arguments as an array.
- bind() returns a new function with this set.

---

19. What is a prototype?
Answer: A prototype is an object from which other objects inherit properties.

---

20. What is prototypal inheritance?
Answer: Objects inherit properties and methods from other objects through the prototype chain.

---

21. What are ES6 features?
Answer: let/const, arrow functions, template literals, destructuring, classes, promises, modules, default/rest/spread operators.

---

22. What is destructuring?
Answer: Destructuring allows extracting values from arrays or objects into distinct variables.

---

23. What are template literals?
Answer: Strings that allow embedded expressions using backticks (``) and ${} placeholders.

---

24. What are arrow functions?
Answer: A shorter syntax for functions that also lexically bind this.

---

25. Function declaration vs function expression?
Answer: Declarations are hoisted, expressions are not.

---

26. What is an IIFE?
Answer: Immediately Invoked Function Expression runs as soon as it is defined.

---

27. What are modules in JavaScript?
Answer: Modules encapsulate code for reusability, using import and export.

---

28. What are JavaScript events?
Answer: Events are browser actions like clicks, keypress, load, etc.

---

29. What is event bubbling vs capturing?
Answer: Bubbling propagates from child to parent; capturing goes from parent to child.

---

30. What is event delegation?
Answer: Assigning a single event handler to a parent element to manage child events.

---

31. Difference between innerHTML and innerText?
Answer: innerHTML includes HTML tags, innerText only returns text.

---

32. What is localStorage, sessionStorage, and cookies?
Answer:

- localStorage: persistent storage across sessions.
- sessionStorage: cleared after session ends.
- cookies: small data sent with requests.

---

33. What is JSON?
Answer: JavaScript Object Notation, a lightweight format for data exchange.

---

34. Difference between for...in and for...of?
Answer: for...in iterates keys of an object; for...of iterates values of iterable objects.

---

35. Difference between map, filter, and reduce?
Answer:

- map: transforms elements.
- filter: selects elements.
- reduce: accumulates values.

---

36. Difference between slice and splice?
Answer: slice returns a shallow copy without modifying the array; splice modifies the array.

---

37. What is Object.is()?
Answer: A strict equality method similar to ===, but correctly compares NaN and distinguishes +0 and -0.

---

38. What is a higher-order function?
Answer: A function that takes other functions as arguments or returns a function.

---

39. What is currying?
Answer: Converting a function with multiple arguments into nested single-argument functions.

---

40. What are JavaScript classes?
Answer: Syntactic sugar over prototypes to create objects and inheritance.

---

41. What are JavaScript symbols?
Answer: Unique, immutable primitive values often used as object keys.

---

42. Shallow copy vs deep copy?
Answer: Shallow copy only copies references, deep copy duplicates all nested data.

---

43. DOM vs BOM?
Answer: DOM = document structure; BOM = browser objects like window, navigator.

---

44. What is strict mode?
Answer: A restricted mode that enforces better coding practices and fewer errors.

---

45. What are generators?
Answer: Functions that can pause execution using yield and resume later.

---

46. Undefined vs undeclared variables?
Answer: Undefined = declared but not assigned. Undeclared = not defined at all.

---

47. Synchronous vs asynchronous callbacks?
Answer: Synchronous callbacks block execution; asynchronous callbacks run later.

---

48. What are WeakMap and WeakSet?
Answer: Collections where references are weak and subject to garbage collection.

---

49. Difference between const object and frozen object?
Answer: const prevents reassignment of the variable, Object.freeze prevents modification of properties.

---

50. Difference between eval() and Function constructor?
Answer: eval() executes code in the current scope, Function creates a new scope.