JavaScript Questions and Answers

1. What will this print?

```
console.log(1 + "2" + "2");
```

A) "122"

B) "5"

C) 122

D) Na

a

2. Which of the following is not a valid way to export a module in ES6?

A) export default function() {}

B) export const foo = 1;

C) module.exports = function() {}

D) export class MyClass {}

c

3. What does the finally block do in a try-catch-finally structure?

A) Executes only if there is an error

B) Executes only if no error occurs

C) Executes regardless of an error

D) Ignores any thrown errors

c

4. What will be the output of this code?

```
(async () => {
  console.log('1');
  await Promise.resolve();
  console.log('2');
})();
console.log('3');
```

A) 1 2 3

B) 1 3 2

C) 3 1 2

D) 2 1 3

b

5. Which method creates a new array with only the elements that pass a condition?

A) map()

B) reduce()

C) filter()

D) forEach()

c

6. Which value is considered falsy in JavaScript?

A) ""

B) 0

C) null

D) All of the above

d

7. What is the output of this code?

```
let a = 5;
(function () {
  console.log(a);
  let a = 10;
})();
```

A) 5

B) 10

C) undefined

D) ReferenceError

d

8. Which of the following is NOT a valid state of a JavaScript Promise?

A) pending

B) resolved

C) fulfilled

D) rejected

b

9. Which of these is a correct way to handle errors in an async function?

A) try/catch

B) .catch()

C) try/finally

D) Both A and B

d

10. What does the ?. operator do in JavaScript?

A) Throws an error if the property is undefined

B) Skips property lookup if the object is null/undefined

C) Automatically logs errors

D) Creates a proxy

b

11. What will this output?

```
Promise.resolve(1)
  .then(x => x + 1)
  .then(x => { throw new Error("Oops") })
  .catch(err => console.log(err.message));
```

A) 2

B) Oops

C) Error

D) undefined

b

12. What is the main difference between map() and forEach()?

A) map() mutates the array

B) forEach() returns a new array

C) map() returns a new array, forEach() does not

D) There is no difference

c

13. Which of the following is not a JavaScript data type?

A) Number

B) String

C) Float

D) Boolean

c

14. What is the result of this expression?

[...'hello']

A) ["hello"]

B) ["h", "e", "l", "l", "o"]

C) TypeError

D) undefined

b

15. Which is a correct use of optional chaining with method calls?

A) obj?.method()

B) obj.?method()

C) obj:?method()

D) obj.method?()

a

16. What will this code output?

```
(function() {
  var a = b = 5;
})();
console.log(typeof b);
```

A) "undefined"

B) "number"

C) "object"

D) "ReferenceError"

b

17. Which of the following is not a way to create an empty object in JS?

A) var obj = {};

B) var obj = Object();

C) var obj = new Object();

D) var obj = Object.create(null);

d

18. What is closure in JavaScript?

A) A function with no return

B) A function inside a loop

C) A function having access to variables in its lexical scope

D) A hidden class

c

19. What is the result of this expression?

false == '0'

A) true

B) false

C) NaN

D) undefined

a

20. What does Object.keys({a:1, b:2}).length return?

A) 0

B) 1

C) 2

D) undefined

c

21. What will this print?

```
const arr = [1, 2, 3];
arr.length = 0;
console.log(arr[0]);
```

A) 1

B) undefined

C) 0

D) Error

b

22. What is the result of the following?

```
const result = '5' - 3;
console.log(result);
```

A) 2

B) 8

C) "53"

D) NaN

a

23. What is the value of x after this runs?

let x = 0;

x ||= 5;

A) 0

B) 5

C) undefined

D) true

b

24. Which of the following are not hoisted?

A) var declarations

B) function declarations

C) let and const declarations

D) All are hoisted

c

25. Which one is NOT a primitive data type in JavaScript?

A) Symbol

B) String

C) Object

D) Undefined

c

26. Which of the following is the correct syntax to print a message in the console in JavaScript?

A) print("Hello World")

B) console.log("Hello World")

C) echo("Hello World")

D) printf("Hello World")

b

27. What will be the output of: typeof null?

A) "null"

B) "object"

C) "undefined"

D) "boolean"

b

28. Which keyword is used to declare a constant in JavaScript?

A) let

B) const

C) var

D) constant

b

29. What is the result of 2 + '2' in JavaScript?

A) 4

B) 22

C) NaN

D) undefined

b

30. Which of the following is the correct way to write a function in JavaScript?

A) function myFunc() {}

B) def myFunc():

C) fun myFunc() {}

D) function:myFunc() {

a

31. Which of the following loop will execute at least once even if the condition is false?

A) for loop

B) while loop

C) do...while loop

D) foreach loop

c

32. How do you write an if statement in JavaScript?

A) if i = 5 then

B) if (i == 5)

C) if i == 5 then

D) if i = 5

b

33. Which method is used to select an element by ID in the DOM?

A) document.querySelectorAll()

B) document.getElementsByName()

C) document.getElementById()

D) document.getElementByClass()

c

34. Which of the following methods is used to combine two arrays in JavaScript?

A) append()

B) concat()

C) combine()

D) attach()

b

35. What will the following code output?

console.log(0.1 + 0.2 === 0.3);

A) true

B) false

C) NaN

D) undefined

b

36. What is a closure in JavaScript?

A) A function that returns another function

B) A function that has access to variables from another function's scope

C) A way to execute functions asynchronously

D) An error handling mechanism

b

37. Which of the following is a feature introduced in ES6?

A) let and const

B) Arrow functions

C) Template literals

D) All of the above

d

38. Which keyword is used to handle exceptions in JavaScript?

A) catch

B) error

C) throw

D) All of the above

d

39. What will this code output?

let x = [1, 2];

let y = [1, 2];

console.log(x == y);

A) true

B) false

C) undefined

D) 1,2

b

40. Which of the following will return true?

[] == false

A) true

B) false

C) throws error

D) undefined

a

41. What is the difference between == and === in JavaScript?

A) == checks value, === checks value and type

B) == checks type only

C) === converts type before comparing

D) No difference

a

42. Which method converts a JSON string to a JavaScript object?

A) JSON.stringify()

B) JSON.parse()

C) JSON.objectify()

D) JSON.convert()

b

43. Which of the following removes the last element of an array?

A) shift()

B) pop()

C) splice()

D) slice()

b

44. What will this code output?

console.log(a);

var a = 5;

A) 5

B) undefined

C) ReferenceError

D) null

b

45. What does async function always return?

A) A function

B) A string

C) A promise

D) An object

c

46. Which of the following is the correct syntax for a Promise?

A) new Promise(success, failure)

B) new Promise(function(resolve, reject) {})

C) Promise.create(function(resolve, reject) {})

D) new Promise(resolve, reject)

b

47. What will this code output?

console.log(typeof NaN);

A) "undefined"

B) "number"

C) "NaN"

D) "object"

b

48. What are the differences between map(), filter(), reduce(), and forEach()?

map() - Transforming array values

filter() - Filtering elements

reduce() - Aggregating values

forEach() - Iterating with side effects

49. What are callback functions?

A function passed as an argument to another function to be called later.

```
function greet(name, callback) {
  callback(`Hello, ${name}`);
}
greet('Sam', msg => console.log(msg)); // Hello, Sam
```

50. Explain async/await with examples.

Syntactic sugar over Promises, making async code look synchronous.

```
async function fetchData() {
  const res = await fetch('/api');
  const data = await res.json();
  console.log(data);
}
```

51. What are Promises and how do you use them?

A Promise represents a value that may be available now, later, or never.

```
const p = new Promise((resolve, reject) => {
```

```
  resolve('done');
});
p.then(console.log); // done
```

52. What is the difference between synchronous and asynchronous code?

Synchronous: executes line by line

Asynchronous: allows non-blocking operations using setTimeout, Promises, async/await

53. What is event delegation?

A technique where a parent element handles events for its child elements using event bubbling.

Used to reduce memory usage and attach fewer event listeners.

```
<body>
  <ul id="menu">
    <li>Home</li>
    <ul>
      <li>Page 1</li>
      <li>Page 2</li>
    </ul>
    <li>About</li>
    <li>Contact</li>
  </ul>
  <script>
    document.getElementById("menu").addEventListener("click", function (e) {
      console.log("You clicked:", e.target.textContent);
    });
  </script>
</body>
```

54. What is a closure in JavaScript?

A closure is a function that "remembers" the variables from its lexical scope, even after that scope has exited.

```
function outer() {
  let count = 0;
  return function inner() {
```

```
  count++;
  console.log(count);
 };
}
const counter = outer();
counter(); // 1
counter(); // 2
```

55. What is the scope of a variable?

Scope defines where a variable is accessible:

var → function scope

let, const → block scope

56. How does JavaScript handle type coercion?

JavaScript automatically converts types when needed, e.g.:

```
'5' + 1 // "51" (string concatenation)
'5' - 1 // 4 (numeric coercion)
```

57. What is the difference between primitive and reference types?

Primitives are copied by value.

Reference types (objects, arrays) are copied by reference.

```
let a=10
let b=a //value is copied to b
a=20
console.log(b) //10 not 20
let arr=[12,34]
let arr1 = arr //reference is pointing to arr
arr.push(34)
console.log(arr1) // [ 12, 34, 34 ] because arr changed
```

58. What are template literals?

String literals that support embedded expressions using backticks:

```
const name = 'Sam';
console.log(`Hello, ${name}!`);
```

59. What is the difference between null and undefined?

undefined: a variable that has been declared but not assigned a value

null: explicitly set to represent no value

60. How does the == vs === operator work?

== compares values after type coercion

=== compares values and types strictly

Example: 5 == '5' is true, 5 === '5' is false.

61. What is the difference between var, let, and const?

var is function-scoped and hoisted.

let and const are block-scoped and also hoisted but not initialized (temporal dead zone).

const can’t be reassigned (but its contents can be mutated if it's an object/array).

62. What are the different data types in JavaScript?

Primitive types: string, number, bigint, boolean, undefined, null, symbol

Reference types: object, array, function