Question 1

What will be the output of the following code?

```javascript
let a = 5;
let b = 10;
console.log(`The sum is ${a + b}`);
```

- A) `The sum is 510`
- B) `The sum is 15`
- C) `The sum is 5 + 10`
- D) `The sum is a + b`

Answer: B) `The sum is 15`

Question 2

What will be the result of the following code?

```javascript
const person = { name: "John" };
person.age = 30;
```

```
console.log(person);
```

- A) `{ name: "John" }`

- B) `{ name: "John", age: 30 }`

- C) `Error`

- D) `undefined`

Answer: B) `{ name: "John", age: 30 }`

Question 3

What is the output of the following code?

```javascript
console.log(typeof NaN);
```

- A) `number`

- B) `string`

- C) `undefined`

- D) `object`

Answer: A) `number`

Question 4

What does the following code snippet do?

```javascript
const array = [1, 2, 3, 4, 5];
const newArray = array.map(num => num * 2);
console.log(newArray);
```

- A) `[1, 2, 3, 4, 5]`
- B) `[2, 4, 6, 8, 10]`
- C) `[1, 4, 9, 16, 25]`
- D) `Error`

Answer: B) `[2, 4, 6, 8, 10]`

Question 5

What will be the output of the following code?

```javascript
function sayHello(name) {
```

```javascript
  return `Hello, ${name}!`;
}
console.log(sayHello());
```

- A) `Hello, !`
- B) `Hello, undefined!`
- C) `Hello, null!`
- D) `Error`

Answer: B) `Hello, undefined!`

Question 6

What is the output of the following code?

```javascript
let x = 0;
while (x < 5) {
  console.log(x);
  x++;
}
```

- A) `0 1 2 3 4`
- B) `1 2 3 4 5`
- C) `0 1 2 3 4 5`
- D) `0 1 2 3 4 6`

Answer: A) `0 1 2 3 4`

Question 7

What will be the output of the following code?

```javascript
console.log(Boolean("false"));
```

- A) `false`
- B) `true`
- C) `undefined`
- D) `Error`

Answer: B) `true`

Question 8

What will be the result of the following code?

```javascript
const numbers = [10, 20, 30];
numbers[0] = 100;
console.log(numbers);
```

- A) `[10, 20, 30]`
- B) `[100, 20, 30]`
- C) `[100, 10, 20, 30]`
- D) `Error`

**Answer: B) `[100, 20, 30]`

Question 9
What is the output of the following code?

```javascript
const sum = (a, b) => a + b;
console.log(sum(5));
```

- A) `NaN`

- B) `5`

- C) `undefined`

- D) `Error`

Answer: A) `NaN`

Question 10

What will be the output of the following code?

```javascript
console.log("5" + 5);
```

- A) `55`

- B) `10`

- C) `TypeError`

- D) `undefined`

Answer: A) `55`

Question 11

What is the output of the following code?

```javascript
let a;
console.log(a);
```

- A) `null`
- B) `undefined`
- C) `0`
- D) `Error`

Answer: B) `undefined`

Question 12

What will be the output of the following code?

```javascript
const numbers = [1, 2, 3];
numbers.push(4);
console.log(numbers.length);
```

- A) `3`

- B) `4`

- C) `5`

- D) `Error`

Answer: B) `4`

Question 13

What is the output of the following code?

```javascript
let x = 10;
if (x == "10") {
  console.log(true);
} else {
  console.log(false);
}
```

- A) `true`

- B) `false`

- C) `Error`

- D) `undefined`

Answer: A) `true`

Question 14

What will be the output of the following code?

```javascript
let x = 10;
let y = "10";
console.log(x === y);
```

- A) `true`
- B) `false`
- C) `Error`
- D) `undefined`

Answer: B) `false`

Question 15

What will the following code output?

```javascript
const obj = { a: 1, b: 2 };
const { a, b, c } = obj;
console.log(c);
```

- A) `1`
- B) `2`
- C) `undefined`
- D) `Error`

Answer: C) `undefined`

Question 16
What is the output of the following code?

```javascript
console.log(typeof null);
```

- A) `object`
- B) `null`

- C) `undefined`
- D) `boolean`

Answer: A) `object`

Question 17
What does the following code output?

```javascript
let a = 3;
let b = new Number(3);
console.log(a == b);
console.log(a === b);
```

- A) `true true`
- B) `true false`
- C) `false true`
- D) `false false`

Answer: B) `true false`

Question 18

What is the output of the following code?

```javascript
function test() {
  console.log(this);
}
test();
```

- A) `undefined`
- B) `null`
- C) `window` or `global`
- D) `Error`

Answer: C) `window` or `global`

Question 19

What will the following code log?

```javascript
let x = [1, 2, 3];
```

```
let y = x;
y.push(4);
console.log(x);
```

- A) `[1, 2, 3]`
- B) `[1, 2, 3, 4]`
- C) `undefined`
- D) `Error`

**Answer: B) `[1, 2, 3, 4]`

Question 20
What is the result of the following code?

```javascript
const x = () => {
  return { message: "Hello" };
};
console.log(x().message);
```

- A) `undefined`

- B) `Hello`

- C) `Error`

- D) `null`

Answer: B) `Hello`

Question 21

What is the output of the following code?

```javascript
let x = [1, 2, 3];
let y = [...x];
y.push(4);
console.log(x);
```

- A) `[1, 2, 3]`

- B) `[1, 2, 3, 4]`

- C) `undefined`

- D) `Error`

Answer: A) `[1, 2, 3]`

Question 22

What will be the result of the following code?

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

- A) `true`
- B) `false`
- C) `undefined`
- D) `Error`

Answer: B) `false`

Question 23

What is the output of the following code?

```javascript
const obj1 = { a: 1 };
const obj2 = { ...obj1, b: 2 };
console.log(obj2);
```

```
```

- A) `{ a: 1 }`

- B) `{ b: 2 }`

- C) `{ a: 1, b: 2 }`

- D) `Error`

Answer: C) `{ a: 1, b: 2 }`

Question 24

What will the following code output?

```javascript
function sayHello(name = "World") {
  return `Hello, ${name}!`;
}
console.log(sayHello("Alice"));
console.log(sayHello());
```

- A) `Hello, Alice! Hello, World!`

- B) `Hello, Alice! Hello, !`

- C) `Hello, ! Hello, World!`

- D) `Error`

Answer: A) `Hello, Alice! Hello, World!`

Question 25
What is the output of the following code?

```javascript
console.log([1, 2, 3] + [4, 5, 6]);
```

- A) `[1, 2, 3, 4, 5, 6]`
- B) `[1, 2, 3][4, 5, 6]`
- C) `1,2,34,5,6`
- D) `Error`

Answer: C) `1,2,34,5,6`

Question 26
What does the following code output?

```javascript
```

```javascript
const arr = [1, 2, 3];
const newArr = arr.filter(num => num > 1);
console.log(newArr);
```

- A) `[1, 2, 3]`
- B) `[2, 3]`
- C) `[1]`
- D) `Error`

Answer: B) `[2, 3]`

Question 27

What will be the output of the following code?

```javascript
console.log(typeof []);
```

- A) `object`
- B) `array`
- C) `undefined`
- D) `Error`

Answer: A) `object`

Question 28

What is the result of the following code?

```javascript
const add = (a, b) => a + b;
console.log(add(5, "5"));
```

- A) `10`
- B) `55`
- C) `NaN`
- D) `Error`

Answer: B) `55`

Question 29

What will the following code log?

```javascript
```

```javascript
let x = 10;
x = x++;
console.log(x);
```

- A) `9`
- B) `10`
- C) `11`
- D) `Error`

Answer: B) `10`

Question 30
What is the output of the following code?

```javascript
const array = [1, 2, 3];
array.forEach(num => {
  if (num > 1) console.log(num);
});
```

- A) `1 2 3`

- B) `2 3`
- C) `1 2`
- D) `3`

Answer: B) `2 3`

Question 31

What will be the result of the following code?

```javascript
console.log("Hello" || "World");
```

- A) `Hello`
- B) `World`
- C) `undefined`
- D) `Error`

Answer: A) `Hello`

Question 32

What is the output of the following code?

```javascript
const person = {
  name: "John",
  greet() {
    console.log(this.name);
  }
};
const greet = person.greet;
greet();
```

- A) `John`
- B) `undefined`
- C) `Error`
- D) `null`

Answer: B) `undefined`

Question 33
What does the following code log?

```javascript
console.log(typeof function() {});
```

- A) `function`
- B) `object`
- C) `undefined`
- D) `Error`

Answer: A) `function`

Question 34

What will be the output of the following code?

```javascript
const fruits = ["Apple", "Banana", "Mango"];
console.log(fruits.includes("Banana"));
```

- A) `true`
- B) `false`
- C) `undefined`
- D) `Error`

Answer: A) `true`

Question 35

What is the result of the following code?

```javascript
console.log("5" - 3);
```

- A) `2`
- B) `8`
- C) `NaN`
- D) `Error`

Answer: A) `2`

Question 36

What will be the output of the following code?

```javascript
let x = 5;
```

```
console.log(x++);
console.log(x);
```

- A) `5 6`
- B) `6 6`
- C) `5 5`
- D) `6 7`

Answer: A) `5 6`

Question 37
What is the output of the following code?

```javascript
const add = (a, b) => a + b;
console.log(add(2));
```

- A) `2`
- B) `NaN`
- C) `undefined`
- D) `Error`

Answer: B) `NaN`

Question 38

What does the following code output?

```javascript
const obj = { a: 1, b: 2, a: 3 };
console.log(obj);
```

- A) `{ a: 1, b: 2 }`
- B) `{ a: 3, b: 2 }`
- C) `{ a: 1, b: 2, a: 3 }`
- D) `Error`

Answer: B) `{ a: 3, b: 2 }`

Question 39

What will the following code log?

```javascript
```

```
console.log([..."hello"]);
```

- A) `[ 'hello' ]`

- B) `[ 'h', 'e', 'l', 'l', 'o' ]`

- C) `[ 'h', 'hello' ]`

- D) `Error`

**Answer: B) `[ 'h', 'e', 'l', 'l', 'o' ]`

Question 40

What is the result of the following code?

```javascript
const arr = [1, 2, 3, 4, 5];
console.log(arr.slice(1, 3));
```

- A) `[1, 2, 3]`

- B) `[2, 3]`

- C) `[2, 3, 4]`

- D) `[1, 2]`

Answer: B) `[2, 3]`

Question 41

What will be the output of the following code?

```javascript
console.log("5" * 2);
```

- A) `10`
- B) `52`
- C) `NaN`
- D) `Error`

Answer: A) `10`

Question 42

What does the following code output?

```javascript
let a = [1, 2, 3];
let b = a;
a.push(4);
```

```
console.log(b);
```

- A) `[1, 2, 3]`

- B) `[1, 2, 3, 4]`

- C) `undefined`

- D) `Error`

Answer: B) `[1, 2, 3, 4]`

Question 43

What is the output of the following code?

```javascript
const set = new Set([1, 2, 3, 4, 4]);
console.log(set.size);
```

- A) `3`

- B) `4`

- C) `5`

- D) `6`

Answer: B) `4`

Question 44

What will the following code log?

```javascript
let x = 1;
let y = 2;
[x, y] = [y, x];
console.log(x, y);
```

- A) `1 2`
- B) `2 1`
- C) `undefined undefined`
- D) `Error`

Answer: B) `2 1`

Question 45

What is the result of the following code?

```javascript
const numbers = [1, 2, 3, 4, 5];
console.log(numbers.reduce((acc, curr) => acc +
curr));
```

- A) `15`

- B) `10`

- C) `5`

- D) `Error`

Answer: A) `15`

Question 46

What will be the output of the following code?

```javascript
let x = 10;
console.log(x.toString(2));
```

- A) `10`

- B) `1010`

- C) `2`
- D) `Error`

Answer: B) `1010`

Question 47

What is the output of the following code?

```javascript
const promise = new Promise((resolve, reject) => {
  resolve("Success");
});
promise.then(data => console.log(data));
```

- A) `Promise`
- B) `undefined`
- C) `Success`
- D) `Error`

Answer: C) `Success`

Question 48

What does the following code log?

```javascript
const obj = { a: 1, b: 2 };
console.log(Object.keys(obj));
```

- A) `[1, 2]`
- B) `["a", "b"]`
- C) `["1", "2"]`
- D) `Error`

Answer: B) `["a", "b"]`

Question 49

What will the following code log?

```javascript
const arr = [1, 2, 3];
const result = arr.map(num => {
  if (num === 2) return;
```

```
  return num * 2;
});
console.log(result);
```

- A) `[2, undefined, 6]`
- B) `[2, 6]`
- C) `[2, , 6]`
- D) `[2, null, 6]`

**Answer: C) `[2, , 6]`

Question 50
What is the result of the following code?

```javascript
console.log("Hello".charAt(1));
```

- A) `H`
- B) `e`
- C) `l`
- D) `o`

Answer: B) `e`