MCQ TEST FOR JS

1.

```
function foo() {
  return new Promise((resolve, reject) => {
    setTimeout(() => {
      resolve("Hello");
    }, 200);
  });
}

foo().then((result) => console.log(result));
```

What will be printed to the console?

a. "Hello"

b. undefined

c. Promise {}

d. Error

2.

```
class Car {
  constructor(make, model) {
    this.make = make;
    this.model = model;
  }

  getDetails() {
    return `${this.make} ${this.model}`;
  }
}

let myCar = new Car("Toyota", "Camry");
console.log(myCar.getDetails());
```

What will be printed to the console?

a. "Toyota"

b. "Camry"

c. "Toyota Camry"

d. undefined

3.

```
let numbers = [1, 2, 3, 4, 5];

let sum = numbers.reduce((acc, val) => {
  return acc * val;
}, 1);

console.log(sum);
```

What will be the output?

a. 0

b. 120

c. 15

d. Error

4.

```
async function getData() {
  return "Async Data";
}

getData().then((result) => console.log(result));
```

What will be printed to the console?

a. "Async Data"

b. undefined

c. Promise {: "Async Data"}

d. Error

5.

```
let x = 5;

function increaseX() {
  return new Promise((resolve) => {
    setTimeout(() => {
      x += 2;
      resolve();
    }, 100);
  });
}

increaseX().then(() => console.log(x));
```

What will be printed to the console?

a. 5

b. 7
c. 2
d. Error

6.

```
let greeting = "Hello";

function changeGreeting() {
  greeting = "Hi";
}

changeGreeting();
console.log(greeting);
```

What will be printed to the console?
a. "Hello"
b. "Hi"
c. undefined
d. Error

7.

```
function* generateNumbers() {
  for (let i = 1; i <= 3; i++) {
    yield i;
  }
}

let iterator = generateNumbers();
console.log(iterator.next().value);
console.log(iterator.next().value);
```

What will be printed to the console?
a. 1 2
b. 1 1
c. 2 3
d. Error

8.

```
let person = {
  name: "John",
  age: 25,
  job: "Developer"
};
```

```
let { name, ...info } = person;
console.log(info);
```

What will be printed to the console?

a. { age: 25, job: "Developer" }

b. { name: "John" }

c. { name: "John", age: 25 }

d. undefined

9.

```
function factorial(n) {
  return n === 0 ? 1 : n * factorial(0);
}

let result = factorial(4);
console.log(result);
```

What will be the output?

a. 24

b. 120

c. 6

d. 4

10.

```
let num1 = 10;
let num2 = "20";

let sum = num1 + num2;
console.log(sum);
```

What will be the output?

a. 30

b. "1020"

c. "1020"

d. Error

11.

```
let promise = new Promise((resolve, reject) => {
  setTimeout(() => {
    reject("Error");
  }, 200);
});

promise.catch((error) => console.log(error));
```

What will be printed to the console?

a. "Error"

b. undefined

c. Promise {: undefined}

d. Error

12.

```
let numbers = [2, 4, 6, 8];
let result = numbers.find((num) => num % 3 === 0);
console.log(result);
```

What will be the output?

a. 2

b. 6

c. 8

d. undefined

13.

```
let text = "Hello, world!";
let reversedText = text.split("").reverse().join("");
console.log(reversedText);
```

What will be printed to the console?

a. "Hello, world!"

b. "!dlrow ,olleH"

c. "dlrow olleH"

d. Error

14.

```
let numbers = [1, 2, 3];
let squaredNumbers = numbers.map((num) => num ** 2);
console.log(squaredNumbers);
```

What will be the value of `squaredNumbers`?

a. [1, 2, 3]

b. [2, 4, 6]

c. [1, 4, 9]
d. [1, 8, 27]

15.

```javascript
let fruits = ["apple", "banana", "orange"];
let [first, ...rest] = fruits;
console.log(first, rest);
```

What will be printed to the console?
a. "apple" ["banana", "orange"]
b. "apple" "banana"
c. "apple" "banana", "orange"
d. undefined

16.

```javascript
let name = "John";
let reversedName = [...name].reverse().join("");
console.log(reversedName);
```

What will be printed to the console?
a. "John"
b. "nhoJ"
c. ["J", "o", "h", "n"]
d. Error

17.

```javascript
let numbers = [1, 2, 3];
let sum = numbers.reduce((acc, num) => acc + num, 1);
console.log(sum);
```

What will be the output?
a. 1
b. 6
c. 10
d. 0

18.

```javascript
let x = 10;
```

```
function changeX() {
  let x = 20;
  return function() {
    x++;
    console.log(x);
  };
}

let closure = changeX();
closure();
closure();
```

What will be printed to the console?

a. 20 21

b. 21 22

c. 20 20

d. 22 23

19.

```
let person = {
  name: "Alice",
  age: 30,
  job: "Engineer"
};

let { name, job } = person;
console.log(name, job);
```

What will be printed to the console?

a. "Alice" "Engineer"

b. "Alice" undefined

c. undefined undefined

d. Error

20.

```
let x = 5;

function multiplyByX(y) {
  return function(z) {
    return x * y * z;
  };
}

let result = multiplyByX(2)(3);
console.log(result);
```

What will be printed to the console?

a. 30

b. 15

c. 6

d. 10