**Javascript Coding Snippets**

**(level 2)**

1. **Array Method Chaining**

What will be the output of the following code?

const numbers = [1, 2, 3, 4, 5];

const result = numbers.filter(num => num % 2 === 0).map(num => num \* 2);

console.log(result);

* A) [1, 2, 3, 4, 5]
* B) [2, 4, 6, 8, 10]
* C) [2, 4]
* D) [4, 8]

**Answer:** D) [4, 8]

1. **Destructuring Assignment**

What will the following code output?

const user = { name: 'Alice', age: 25, country: 'USA' };

const { name, age } = user;

console.log(name, age);

* A) undefined undefined
* B) Alice 25
* C) Alice USA
* D) undefined 25

**Answer:** B) Alice 25

1. **Async/Await**

Which line will be executed first in the following code?

async function fetchData() {

console.log('Start');

const response = await fetch('https://jsonplaceholder.typicode.com/todos/1');

console.log('Data Fetched');

}

fetchData();

console.log('End');

* A) 'Start'
* B) 'Data Fetched'
* C) 'End'
* D) The code will throw an error.

**Answer:** A) 'Start'

1. **Event Delegation**

Given the following HTML structure, which event target will trigger the click event in the code below?

<div id="parent">

<button class="delete">Delete 1</button>

<button class="delete">Delete 2</button>

</div>

document.querySelector('#parent').addEventListener('click', (event) => {

if (event.target && event.target.matches('button.delete')) {

console.log('Delete button clicked:', event.target.textContent);

}

});

* A) #parent
* B) .delete
* C) Both A and B
* D) None of the above

**Answer:** B) .delete

1. **Closures**

What will the following code output?

function createCounter() {

let count = 0;

return function() {

count += 1;

return count;

};

}

const counter = createCounter();

console.log(counter());

console.log(counter());

* A) 0 1
* B) 1 1
* C) 1 2
* D) 2 3

**Answer:** C) 1 2

1. **Debouncing**

What will be the effect of debouncing on the following resize event?

function debounce(func, delay) {

let timeout;

return function() {

clearTimeout(timeout);

timeout = setTimeout(func, delay);

};

}

window.addEventListener('resize', debounce(() => {

console.log('Window resized');

}, 300));

* A) The event will fire immediately on each resize.
* B) The event will fire after 300ms of inactivity.
* C) The event will fire every 300ms during resize.
* D) The event will not fire at all.

**Answer:** B) The event will fire after 300ms of inactivity.

1. **Class Inheritance**

Given the following code, what will be the output?

class Animal {

constructor(name) {

this.name = name;

}

speak() {

console.log(`${this.name} makes a sound.`);

}

}

class Dog extends Animal {

speak() {

console.log(`${this.name} barks.`);

}

}

const dog = new Dog('Rex');

dog.speak();

* A) Rex makes a sound.
* B) Rex barks.
* C) Dog barks.
* D) The code will throw an error.

**Answer:** B) Rex barks.

1. **Optional Chaining**

What will the following code output?

const user = {

name: 'John',

address: {

city: 'New York',

}

};

console.log(user.address?.city);

console.log(user.address?.zipCode);

* A) 'New York' and undefined
* B) 'New York' and null
* C) 'New York' and Error
* D) undefined and undefined

**Answer:** A) 'New York' and undefined

1. **Rest and Spread Operators**

What will be the output of the following code?

const arr1 = [1, 2, 3];

const arr2 = [4, 5];

const arr3 = [...arr1, ...arr2];

console.log(arr3);

* A) [1, 2, 3, 4, 5]
* B) [1, 2, 3, [4, 5]]
* C) [4, 5, 1, 2, 3]
* D) [[1, 2, 3], [4, 5]]

**Answer:** A) [1, 2, 3, 4, 5]

1. **Object Destructuring with Default Values**

What will be the output of the following code?

const person = { name: 'Jane', age: 28 };

const { name, city = 'Unknown' } = person;

console.log(city);

* A) 'Jane'
* B) 'Unknown'
* C) undefined
* D) The code will throw an error.

**Answer:** B) 'Unknown'

1. **Promise Chaining**

What will be the output of the following code?

const promise = new Promise((resolve, reject) => {

resolve(10);

});

promise

.then(value => {

console.log(value);

return value \* 2;

})

.then(value => {

console.log(value);

return value \* 2;

})

.finally(() => {

console.log('Done');

});

* A) 10, 20, Done
* B) 10, 20
* C) 10, 20, 40, Done
* D) Done, 10, 20

**Answer:** A) 10, 20, Done

1. **Set Object**

What will be the output of the following code?

const mySet = new Set([1, 2, 2, 3, 4]);

console.log(mySet.size);

* A) 5
* B) 4
* C) 3
* D) 2

**Answer:** B) 4

1. **Arrow Functions**

What will be the output of the following code?

const add = (a, b) => {

return a + b;

};

console.log(add(3, 4));

* A) 34
* B) undefined
* C) 7
* D) The code will throw an error.

**Answer:** C) 7

1. **Nullish Coalescing Operator (??)**

const user = { name: 'Sam', age: null };

console.log(user.age ?? 30);

* A) null
* B) 30
* C) undefined
* D) The code will throw an error.

**Answer:** B) 30

1. **Array .reduce() Method**

What will be the output of the following code?

const numbers = [1, 2, 3, 4, 5];

const sum = numbers.reduce((acc, curr) => acc + curr, 0);

console.log(sum);

* A) 15
* B) 10
* C) 5
* D) 0

**Answer:** A) 15

1. **Object Methods**

What will be the output of the following code?

const user = {

name: 'Bob',

greet: function() {

return `Hello, ${this.name}`;

}

};

console.log(user.greet());

* A) Hello, undefined
* B) Hello, Bob
* C) Hello, this.name
* D) The code will throw an error.

**Answer:** B) Hello, Bob

1. **Default Parameters**

What will be the output of the following code?

function multiply(a, b = 5) {

return a \* b;

}

console.log(multiply(3));

* A) 3
* B) 15
* C) NaN
* D) undefined

**Answer:** B) 15

1. **Template Literals**

const user = { firstName: 'John', lastName: 'Doe' };

const greeting = `Hello, ${user.firstName} ${user.lastName}!`;

console.log(greeting);

* A) Hello, John Doe!
* B) Hello, ${user.firstName} ${user.lastName}!
* C) Hello, John!
* D) The code will throw an error.

**Answer:** A) Hello, John Doe!

1. **Spread Operator in Functions**

What will be the output of the following code?

function sum(...args) {

return args.reduce((acc, curr) => acc + curr, 0);

}

console.log(sum(1, 2, 3, 4));

* A) 10
* B) 0
* C) [1, 2, 3, 4]
* D) The code will throw an error.

**Answer:** A) 10

1. **Array Destructuring**

What will be the output of the following code?

const colors = ['red', 'green', 'blue'];

const [firstColor, secondColor] = colors;

console.log(firstColor, secondColor);

* A) red green
* B) red blue
* C) blue green
* D) undefined undefined

**Answer:** A) red green

1. **Object Freeze**

What will be the output of the following code?

const user = Object.freeze({ name: 'Alice', age: 30 });

user.age = 35;

console.log(user.age);

* A) 35
* B) 30
* C) undefined
* D) The code will throw an error.

**Answer:** B) 30

1. **Rest Parameters**

What will be the output of the following code?

function logNumbers(first, ...rest) {

console.log(first);

console.log(rest);

}

logNumbers(1, 2, 3, 4);

* A) 1 [2, 3, 4]
* B) 1 2
* C) [1, 2] [3, 4]
* D) 1 2 3 4

**Answer:** A) 1 [2, 3, 4]

1. **Short-Circuit Evaluation**

What will be the output of the following code?

const name = null;

const displayName = name || 'Guest';

console.log(displayName);

* A) null
* B) Guest
* C) undefined
* D) The code will throw an error.

**Answer:** B) Guest

1. **String Methods**

What will be the output of the following code?

const text = 'Hello, World!';

console.log(text.includes('World'));

* A) true
* B) false
* C) undefined
* D) The code will throw an error.

**Answer:** A) true

1. **Async Function Return Value**

What will be the output of the following code?

async function fetchData() {

return 'Data';

}

fetchData().then(data => console.log(data));

* A) Data
* B) undefined
* C) Promise {<resolved>: "Data"}
* D) The code will throw an error.

**Answer:** A) Data

**Javascript Coding Snippets**

**(level 1)**

**1. Variable Declaration**

What will be the output of the following code?

let x = 10;

x = 20;

console.log(x);

* A) 10
* B) 20
* C) undefined
* D) null

**Answer:** B) 20

**2. Data Types**

What will be the output of the following code?

let name = "Alice";

let age = 25;

console.log(typeof name, typeof age);

* A) string number
* B) number string
* C) boolean string
* D) undefined undefined

**Answer:** A) string number

**3. String Concatenation**

What will be the output of the following code?

let firstName = "John";

let lastName = "Doe";

let fullName = firstName + " " + lastName;

console.log(fullName);

* A) JohnDoe
* B) John Doe
* C) John Doe
* D) undefined

**Answer:** B) John Doe

**4. Arithmetic Operations**

What will be the output of the following code?

let a = 5;

let b = 10;

let c = a + b;

console.log(c);

* A) 15
* B) 510
* C) undefined
* D) NaN

**Answer:** A) 15

**5. Equality Comparison**

What will be the output of the following code?

let x = 5;

let y = "5";

console.log(x == y);

* A) true
* B) false
* C) undefined
* D) null

**Answer:** A) true

**6. Strict Equality Comparison**

What will be the output of the following code?

let x = 5;

let y = "5";

console.log(x === y);

* A) true
* B) false
* C) undefined
* D) null

**Answer:** B) false

**7. Conditional Statement**

What will be the output of the following code?

let age = 18;

if (age >= 18) {

console.log("Adult");

} else {

console.log("Minor");

}

* A) Adult
* B) Minor
* C) undefined
* D) null

**Answer:** A) Adult

**8. Logical Operators**

What will be the output of the following code?

let isStudent = true;

let isMember = false;

console.log(isStudent && isMember);

* A) true
* B) false
* C) undefined
* D) null

**Answer:** B) false

**9. Loops**

What will be the output of the following code?

let sum = 0;

for (let i = 1; i <= 3; i++) {

sum += i;

}

console.log(sum);

* A) 6
* B) 3
* C) 0
* D) undefined

**Answer:** A) 6

**10. Array Access**

What will be the output of the following code?

let fruits = ["apple", "banana", "cherry"];

console.log(fruits[1]);

* A) apple
* B) banana
* C) cherry
* D) undefined

**Answer:** B) banana

**11. Function Declaration**

What will be the output of the following code?

function greet(name) {

return "Hello, " + name;

}

console.log(greet("Alice"));

* A) Hello, Alice
* B) Hello, undefined
* C) Hello,
* D) undefined

**Answer:** A) Hello, Alice

**12. Array Length**

What will be the output of the following code?

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

console.log(numbers.length);

* A) 4
* B) 5
* C) undefined
* D) null

**Answer:** B) 5

**13. Object Properties**

What will be the output of the following code?

let car = { make: "Toyota", model: "Corolla" };

console.log(car.make);

* A) Toyota
* B) Corolla
* C) undefined
* D) null

**Answer:** A) Toyota

**14. Undefined Value**

What will be the output of the following code?

let x;

console.log(x);

* A) undefined
* B) null
* C) 0
* D) NaN

**Answer:** A) undefined

**15. Increment Operator**

What will be the output of the following code?

let x = 5;

x++;

console.log(x);

* A) 5
* B) 6
* C) 7
* D) undefined

**Answer:** B) 6

**16. Array Push Method**

What will be the output of the following code?

let colors = ["red", "green", "blue"];

colors.push("yellow");

console.log(colors);

* A) ["red", "green", "blue"]
* B) ["red", "green", "blue", "yellow"]
* C) ["red", "green", "yellow"]
* D) undefined

**Answer:** B) ["red", "green", "blue", "yellow"]

**17. String Length**

What will be the output of the following code?

let text = "Hello, world!";

console.log(text.length);

* A) 13
* B) 12
* C) 14
* D) undefined

**Answer:** A) 13

**18. Logical OR Operator**

What will be the output of the following code?

let x = 0;

let y = x || 10;

console.log(y);

* A) 0
* B) 10
* C) undefined
* D) null

**Answer:** B) 10

**19. Ternary Operator**

What will be the output of the following code?

let age = 20;

let isAdult = (age >= 18) ? "Yes" : "No";

console.log(isAdult);

* A) Yes
* B) No
* C) undefined
* D) null

**Answer:** A) Yes

**20. Boolean Conversion**

What will be the output of the following code?

let value = "Hello";

console.log(Boolean(value));

* A) true
* B) false
* C) undefined
* D) null

**Answer:** A) true

**21. Default Parameters**

What will be the output of the following code?

function greet(name = "Guest") {

return "Hello, " + name;

}

console.log(greet());

* A) Hello,
* B) Hello, undefined
* C) Hello, Guest
* D) Hello, null

**Answer:** C) Hello, Guest

**22. Template Literals**

What will be the output of the following code?

let name = "John";

let message = `Welcome, ${name}!`;

console.log(message);

* A) Welcome, John!
* B) Welcome, ${name}!
* C) Welcome, !
* D) undefined

**Answer:** A) Welcome, John!

**23. Destructuring Assignment**

What will be the output of the following code?

let [a, b] = [10, 20];

console.log(a, b);

* A) 10 20
* B) 20 10
* C) undefined undefined
* D) null null

**Answer:** A) 10 20

**24. Arrow Functions**

What will be the output of the following code?

let sum = (a, b) => a + b;

console.log(sum(5, 10));

* A) 15
* B) 510
* C) undefined
* D) null

**Answer:** A) 15

**25. Array Map Method**

What will be the output of the following code?

let numbers = [1, 2, 3];

let doubled = numbers.map(num => num \* 2);

console.log(doubled);

* A) [2, 4, 6]
* B) [1, 2, 3]
* C) undefined
* D) null

**Answer:** A) [2, 4, 6]

**26. Spread Operator**

What will be the output of the following code?

let arr1 = [1, 2, 3];

let arr2 = [...arr1, 4, 5];

console.log(arr2);

* A) [1, 2, 3, 4, 5]
* B) [4, 5]
* C) undefined
* D) null

**Answer:** A) [1, 2, 3, 4, 5]

**27. Rest Parameters**

What will be the output of the following code?

function add(...numbers) {

return numbers.reduce((sum, num) => sum + num, 0);

}

console.log(add(1, 2, 3, 4));

* A) 10
* B) 0
* C) undefined
* D) null

**Answer:** A) 10

**28. Object Property Access**

What will be the output of the following code?

let person = {

name: "Alice",

age: 30

};

console.log(person["age"]);

* A) 30
* B) Alice
* C) undefined
* D) null

**Answer:** A) 30

**29. Array Join Method**

What will be the output of the following code?

let words = ["Hello", "World"];

let sentence = words.join(" ");

console.log(sentence);

* A) Hello World
* B) Hello,World
* C) HelloWorld
* D) undefined

**Answer:** A) Hello World

**30. Type Coercion**

What will be the output of the following code?

let result = "5" - 2;

console.log(result);

* A) 3
* B) 52
* C) NaN
* D) undefined

**Answer:** A) 3

**31. Nullish Coalescing Operator**

What will be the output of the following code?

let value = null;

let result = value ?? "Default";

console.log(result);

* A) Default
* B) null
* C) undefined
* D) 0

**Answer:** A) Default

**32. Comparison Operators**

What will be the output of the following code?

let x = 10;

console.log(x > 5);

* A) true
* B) false
* C) undefined
* D) null

**Answer:** A) true

**33. Switch Statement**

What will be the output of the following code?

let fruit = "apple";

switch(fruit) {

case "banana":

console.log("Banana");

break;

case "apple":

console.log("Apple");

break;

default:

console.log("Unknown");

}

* A) Apple
* B) Banana
* C) Unknown
* D) undefined

**Answer:** A) Apple

**34. Falsy Values**

Which of the following values is considered falsy in JavaScript?

let value = 0;

* A) 0
* B) 1
* C) "false"
* D) []

**Answer:** A) 0

**35. Array Filter Method**

What will be the output of the following code?

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

let evenNumbers = numbers.filter(num => num % 2 === 0);

console.log(evenNumbers);

* A) [2, 4]
* B) [1, 3, 5]
* C) [1, 2, 3, 4, 5]
* D) undefined

**Answer:** A) [2, 4]

**36. Object Methods**

What will be the output of the following code?

let person = {

name: "Bob",

greet: function() {

return "Hello, " + this.name;

}

};

console.log(person.greet());

* A) Hello, Bob
* B) Hello,
* C) undefined
* D) null

**Answer:** A) Hello, Bob

**37. While Loop**

What will be the output of the following code?

let i = 0;

while (i < 3) {

console.log(i);

i++;

}

* A) 0 1 2
* B) 1 2 3
* C) 0 1 2 3
* D) undefined

**Answer:** A) 0 1 2

**38. For Loop**

What will be the output of the following code?

for (let i = 1; i <= 3; i++) {

console.log(i);

}

* A) 1 2 3
* B) 0 1 2
* C) undefined
* D) null

**Answer:** A) 1 2 3

**39. String Indexing**

What will be the output of the following code?

let text = "Hello";

console.log(text[1]);

* A) e
* B) H
* C) l
* D) undefined

**Answer:** A) e

**40. Break Statement**

What will be the output of the following code?

for (let i = 0; i < 5; i++) {

if (i === 3) {

break;

}

console.log(i);

}

* A) 0 1 2
* B) 0 1 2 3
* C) 3 4 5
* D) undefined

**Answer:** A) 0 1 2