

0

@CODE.CLASH

JavaScript Console



NEXT →

Hey Everyone 

Javascript is everywhere. Millions of webpages are built on JS.

In this Post, you will learn about the JavaScript console, types of console methods with examples of how to use them.

Do Like, save and Share This Post If You Found This Helpful.

What is Console?

The **console** is an object in javascript which is used for **debugging** and **logging** results.

- All modern browsers have a **web console** for debugging.
- Console is a **global object** so it is available in every scope.
- It is also available in the **global scope** of the browser window. You can use it as `window.console` or direct **console**.
- Here we'll discuss some **most used Console methods**.

Console Methods

Console in javascript has **several different methods**. Among all the **log** method is most commonly used.

- `console.log()`
- `console.error()`
- `console.clear()`
- `console.time()`
- `console.table()`
- `console.count()`
- `console.assert()`
- ...

console.log()

Mainly used to log the output to the console. We can put any type inside the log(), be it a string, array, object etc.

```
console.log(123); //number  
console.log('Hello World!'); //string  
console.log(10 + 20); //expression  
console.log(new Date()); //object
```

```
123  
Hello World!  
30  
Sat Aug 06 2022 13:16:46 GMT+0530 (India Standard Time)
```

console.error()

It's used to log error message to the console.

```
> console.error('This is a simple error');
✖  ▶ This is a simple error
```

console.clear()

It's used to clear the console.

```
console.clear();
```

Console was cleared

undefined

console.time()

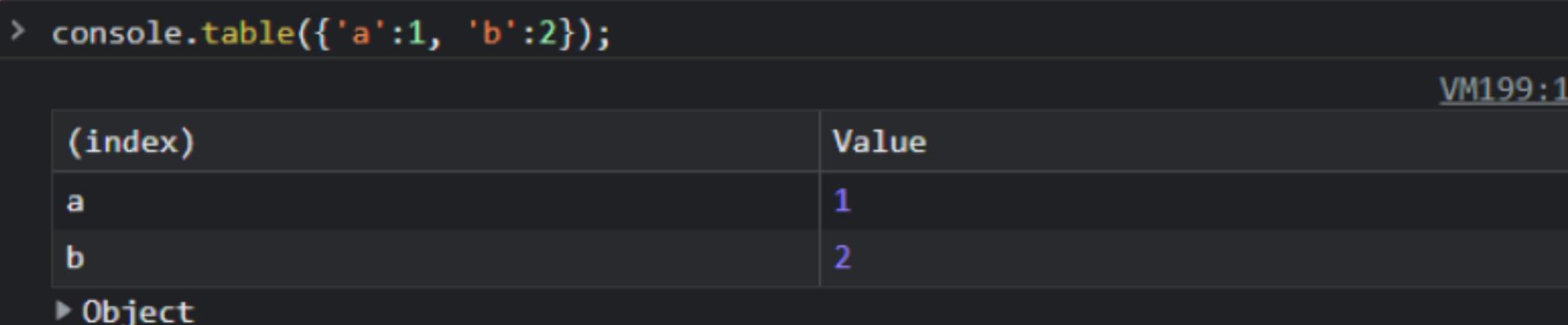
It's used to **track** how long does an operation **takes to execute**.

```
console.time('trackLoop');
for (let i = 0; i < 10000; i++) {}
console.timeEnd('trackLoop');

// trackLoop: 0.546142578125 ms
```

console.table()

It's allows us to **generate a table** inside a console.



A screenshot of a browser developer tools console window. The command `console.table({ 'a': 1, 'b': 2 })` is entered. The output is a table with two rows. The first row has a header 'index' and a value 'Object'. The second row has two columns: 'a' under 'index' and '1' under 'Value'. The third row has two columns: 'b' under 'index' and '2' under 'Value'. The table is displayed with horizontal and vertical grid lines. The status bar at the bottom right shows 'VM199:1'.

(index)	Value
a	1
b	2

console.count()

It's used to **count the number** that the function hit by this **counting method**.

```
for (let i = 0; i < 5; i++) {  
    console.count('Count number');  
}  
  
// Output  
Count number: 1  
Count number: 2  
Count number: 3  
Count number: 4  
Count number: 5
```

Custom Console Logs

User can **add Styling** to the console logs
in order to make **logs Custom** .

```
const spacing = '10px';
const styles =
  `padding: ${spacing}; background-color: white;
  color: blue; font-weight: bold;
  border: 1px solid black; font-size: 2em;`;

console.log('%cCode.Clash', styles);
```

Code.Clash

@CODE.CLASH

THANKS FOR YOUR ATTENTION



CODE.CLASH



IMTIYAZ NANDASANIYA

**LIKE AND SAVE IT FOR
LATER**