

Console API

There's More To
console
Than
`.log()`

```
.log()  
.table()  
.group()  
.count()  
.trace()  
.debug()  
.error()  
.warn()  
.info()  
.time()
```

Source: <https://www.freecodecamp.org/news/javascript-console-log-example-how-to-print-to-the-console-in-js/>

AGENDA

1. Log variations
 2. Multiple values
 3. Assert
 4. Counting
 5. Track time
 6. Groups
 7. Stack trace
 8. Tables
 9. Styles
 10. Clear
-

1. Log variations

```
console.log('LOG: You properly just need to use me and no need to care others >.<');
console.info('INFO: Javascript is easy');
console.debug('DEBUG: Nice');
console.warn('WARN: Beautiful');
console.error('ERROR: Lovely');
```

```
> console.log('LOG: You properly just need to use me and no need to care others >.<');
  console.info('INFO: Javascript is easy');
  console.debug('DEBUG: Nice');
  console.warn('WARN: Beautiful');
  console.error('ERROR: Lovely');
```

LOG: You properly just need to use me and no need to care others >.<

[react_devtools_backend.js:4049](#)

INFO: Javascript is easy

[VM542:2](#)

DEBUG: Nice

[VM542:3](#)

⚠️ WARN: Beautiful

[react_devtools_backend.js:4049](#)

❌ ERROR: Lovely

[react_devtools_backend.js:4049](#)

2. Multiple values

Group multiple values into object for better readable

```
const student = {
  id: 1,
  name: 'Alice',
  age: 18,
};

const city = {
  id: 1,
  name: 'HCMC'
};

// NOT GOOD
console.log(student);
console.log(city);

// BETTER
console.log({ student, city });
```

▶ {id: 1, name: 'Alice', age: 18}

▶ {id: 1, name: 'HCMC'}

```
▼ {student: {...}, city: {...}} ⓘ  
  ▼ city:  
    id: 1  
    name: "HCMC"  
    ► [[Prototype]]: Object  
  ▼ student:  
    age: 18  
    id: 1  
    name: "Alice"  
    ► [[Prototype]]: Object  
  ► [[Prototype]]: Object
```

Log with variables

```
const student = {  
  id: 1,  
  name: 'Alice',  
  age: 18,  
};  
  
console.log(`My student: ${student} is lazy`);  
// My student: [object Object] is lazy
```

```
const student = {  
  id: 1,  
  name: 'Alice',  
  age: 18,  
};  
  
console.log('My student: %o is lazy', student);
```

```
My student: ► {id: 1, name: 'Alice', age: 18} is lazy
```

```
// OK  
console.log('Name: %s and Age: %i', student.name, student.age);  
// Name: Alice and Age: 18  
  
// GOOD  
console.log(`Name: ${student.name} and Age: ${student.age}`);  
// Name: Alice and Age: 18
```

3. Assert

```
console.assert(condition, message);
```

- Only show message if condition is false.

```
console.assert(1 === 1, 'equal but not show in console');  
console.assert(1 === 2, 'not equal and show in console');
```

```
> console.assert(1 === 1, 'equal but not show in console');  
console.assert(1 === 2, 'not equal and show in console');
```

✖ ▶ Assertion failed: not equal and show in console

4. Counting

#	Name	Desc
1	console.count(label = 'default')	count how many times that have been called
2	console.countReset(label = 'default')	reset the counter

```
console.count();           // default: 1  
console.count('default'); // default: 2  
console.count();           // default: 3
```

```
console.count('easy'); // easy: 1  
console.count('easy'); // easy: 2  
  
console.count('intermediate'); // intermediate: 1  
console.count('intermediate'); // intermediate: 2  
console.countReset('intermediate');  
console.count('intermediate'); // intermediate: 1  
  
console.count('hard'); // hard: 1  
console.count('hard'); // hard: 2  
console.count('hard'); // hard: 3
```

5. Track time

#	Name	Desc
1	<code>console.time(name = 'default')</code>	start timer
2	<code>console.timeLog(name = 'default')</code>	log time ellapsed from start
3	<code>console.timeEnd(name = 'default')</code>	end timer

```
console.time('loop'); // start timer

let count = 0;
for (let i = 0; i < 10000; i++) {
  count++;

  if (i % 1000 === 0) console.timeLog('loop')
}

console.timeEnd('loop'); // end timer
```

6. Groups

#	Name	Desc
1	<code>console.group(name)</code>	start a group with status = expanded
2	<code>console.groupCollapsed(name)</code>	start a group with status = collapsed
3	<code>console.groupEnd()</code>	end a group

```
console.group('TODO ADD');
console.log('Before', { title: 'easy' });
console.log('After', { id: 1, title: 'easy' });
console.groupEnd();
```

▼ TODO ADD	
Before ▶	{title: 'easy'}
After ▶	{id: 1, title: 'easy'}

```
console.groupCollapsed('TODO ADD');
console.log('Before', { title: 'easy' });
console.log('After', { id: 1, title: 'easy' });
console.groupEnd();
```

[▶ TODO ADD](#)

7. Stack trace

```
function foo() {  
  function bar() {  
    console.trace();  
  }  
  bar();  
}  
  
foo();
```

▼ console.trace
bar @ VM734:3
foo @ VM734:5
(anonymous) @ VM734:8

8. Tables

```
console.table({  
  id: 1,  
  name: 'Alice',  
  age: 18,  
  isHero: true,  
});
```

VM1876:1

(index)	Value
id	1
name	'Alice'
age	18
isHero	true

► Object

```
console.table([
  {
    id: 1,
    name: 'Alice',
    age: 18,
    isHero: true,
  },
  {
    id: 2,
    name: 'Bob',
    age: 20,
    isHero: false,
  },
], ['id', 'name', 'age']);
```

VM1871:1

(index)	id	name	age
0	1	'Alice'	18
1	2	'Bob'	20

► Array(2)

9. Styles

- use `%c` to start a styled text.

```
> console.log('Welcome to %cEasy Frontend', 'font-size: 64px; color: goldenrod; font-weight: bold; background: black;')
```

[react_devtools_backend.js:4049](#)

Welcome to

Easy Frontend

10. Clear

- `console.clear()` to clear current console.
- or using **Ctrl + L** to quickly clear the console


Tham khảo

- <https://developer.mozilla.org/en-US/docs/Web/API/console>
 - <https://www.freecodecamp.org/news/javascript-console-log-example-how-to-print-to-the-console-in-js/>
-

Khoá học Javascript cho người mới bắt đầu 2021 🎉

- Tác giả: **Hậu Nguyễn** - Founder Easy Frontend
- Khoá học chỉ được published trên Udemy, không thông qua trung gian.
- Khoá học không bán dạng videos upload trên Google Drive hay bất cứ hình thức nào tương tự.
- Khoá học có nhóm discord để hỗ trợ trong quá trình học tập.

 Liên hệ tác giả để được hỗ trợ:

-  Facebook: <https://www.facebook.com/nvhauesmn/>
-  Fanpage: <https://www.facebook.com/learn.easyfrontend>
-  Youtube Channel: <https://www.youtube.com/easyfrontend>