

# Operators in Javascript

Một số thuật ngữ cần nắm:

#	Name	Desc	Eg
1	Arithmetic operators	Toán tử số học	<code>a + b; a % b;</code>
2	Unary operators	Operator có 1 operand	<code>count++ --number !error</code>
3	Binary operators	Operator có 2 operand	<code>a = b; a + b;</code>
4	Ternary operators	Operator có 3 operand	<code>isShow ? 'Show' : 'Hide'</code>
5	Logical operators	Operator luận lý	<code>a    b; a &amp;&amp; b;</code>
6	Bitwise operators	Operator xử lý BIT	<code>a &gt;&gt; b; a   b</code>

Tham khảo thêm: <https://javascript.info/operators>

## Assignment - Phép gán

```
const count = 1;

const result = 1 + 2 * 3;

const name = 'Easy Frontend';

const isValid = true;
```

## Unary

```
const count = 1;
const negativeCount = -count; // -1

let idx = 1;
const incrementPrefix = ++idx; // 2

let idx = 1;
const incrementSuffix = idx++; // 1

const quantity = +'2'; // number 2

const isValid = !true; // false
```

## Ternary

```
condition ? exprIfTrue : exprIfFalse
```

```
const grade = mark > 8 ? 'GOOD' : 'BAD';

// Avoid nested ternary
const grade = mark > 8 ? 'GOOD' : mark > 5 ? 'OK' : 'BAD';
```

## Operator Precedence - Thứ tự ưu tiên của các operators

```
const result = 1 + (2 + -2) * 3 - +"3"; // KQ là bao nhiêu?
```

- Điểm càng cao thì độ ưu tiên càng lớn.
- Dựa vào bảng này để biết là Javascript sẽ tính toán cái nào trước, cái nào sau.

### Table

The following table lists operators in order from highest precedence (21) to lowest precedence (1).

Note that [spread syntax](#) is intentionally not included in the table — because, to quote an [answer at Stack Overflow](#), “[Spread syntax is not an operator](#)” and therefore does not have a precedence. It is part of the array literal and function call (and object literal) syntax.”

Precedence	Operator type	Associativity	Individual operators
21	<a href="#">Grouping</a>	n/a	( ... )
20	<a href="#">Member Access</a>	left-to-right	... . ...
	<a href="#">Computed Member Access</a>	left-to-right	... [ ... ]
	<a href="#">new</a> (with argument list)	n/a	new ... ( ... )
	<a href="#">Function Call</a>	left-to-right	... ( ... )
	<a href="#">Optional chaining</a>	left-to-right	? .
19	<a href="#">new</a> (without argument list)	right-to-left	new ...
18	<a href="#">Postfix Increment</a>	n/a	... ++
	<a href="#">Postfix Decrement</a>		... --

Tham khảo: [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator\\_Precedence](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator_Precedence)

---

## 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>