# Aritmética do JavaScript

[❮ Anterior](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_operators.asp&usg=ALkJrhgyp9kZtWdKPAeyh7ON6X22jIilyg)[Próximo ❯](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_assignment.asp&usg=ALkJrhgDWJbLDQUlWzrFSJF9on8JN8ZhBg)

## Operadores Aritméticos de JavaScript

Operadores aritméticos executam aritmética em números (literais ou variáveis).

|  |  |
| --- | --- |
| **Operator** | **Description** |
| + | Addition |
| - | Subtraction |
| \* | Multiplication |
| / | Division |
| % | Modulus |
| ++ | Increment |
| -- | Decrement |

## Operaçoes aritimeticas

Uma operação aritmética típica opera em dois números.

Os dois números podem ser literais:

### Exemplo

var x = 100 + 50;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_operation&usg=ALkJrhj_TjT3Yp7vhAjgedZtq84LWXEiAg)

ou variáveis:

### Exemplo

var x = a + b;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_variables&usg=ALkJrhhis6PPC8GgokD22LZKsJ-BukRQtw)

ou expressões:

### Exemplo

var x = (100 + 50) \* a;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_expressions&usg=ALkJrhjN8t3HxK8CLh6r16Y2u_jZ8VsErA)

## Operadores e Operandos

Os números (em uma operação aritmética) são chamados de **operandos** .

A operação (a ser realizada entre os dois operandos) é definida por um **operador** .

|  |  |  |
| --- | --- | --- |
| **Operando** | **Operador** | **Operando** |
| 100 | + | 50 |

O operador de **adição** (+) adiciona números:

### Adicionando

var x = 5;  
var y = 2;  
var z = x + y;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_add&usg=ALkJrhjWlj5Aw8jDLa2UxJaBPKgn6tiJ-A)

O operador de **subtração** (-) subtrai números.

### Subtraindo

var x = 5;  
var y = 2;  
var z = x - y;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_sub&usg=ALkJrhhALO6f5bpvq_dhfYcKEGGZ5GMbzQ)

O operador de **multiplicação** (\*) multiplica números.

### Multiplicando

var x = 5;  
var y = 2;  
var z = x \* y;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_mult&usg=ALkJrhgRXqG0OT3gbHbC4V-TPUSrYfRMng)

O operador da **divisão** (/) divide os números.

### Dividindo

var x = 5;  
var y = 2;  
var z = x / y;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_div&usg=ALkJrhh-aHkf3lz4QVm5DxgcAE6CJmShGA)

O operador **modular** (%) retorna o restante da divisão.

### Módulo

var x = 5;  
var y = 2;  
var z = x % y;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_mod&usg=ALkJrhi1CGN07sK2IVN_tIGW_9vC2Q16Fw)

O operador de **incremento** (++) incrementa os números.

### Incrementando

var x = 5;  
x++;  
var z = x;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_increment&usg=ALkJrhh79X2OXa307PJGpZlnFEw7rxj7Ig)

O operador de **decremento** (-) diminui os números.

### Decricionamento

var x = 5;  
x--;  
var z = x;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_oper_decrement&usg=ALkJrhgUPXgePwXWCPQqeTwRwjFX8X_Yow)

## operador precedente

O precedente do operador descreve a ordem em que as operações são realizadas em uma expressão aritmética.

### Exemplo

var x = 100 + 50 \* 3;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_precedence1&usg=ALkJrhjkTysb-PD1ppu9hw2DDL8hc-gbkw)

O resultado é de um exemplo acima do mesmo que 150 \* 3, ou é o mesmo que 100 + 150?

A adição ou a multiplicação são feitas primeiro?

Como na matemática escolar tradicional, a multiplicação é feita primeiro.

A multiplicação (\*) e a divisão (/) têm maior **precedência do** que a adição (+) e a subtração (-).

E (como na matemática escolar), a precedência pode ser alterada usando parênteses:

### Exemplo

var x = (100 + 50) \* 3;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_precedence2&usg=ALkJrhhqilSBu9QU1YwpFj5TKISI1nm-gg)

Ao usar parênteses, as operações dentro dos parênteses são calculadas primeiro.

Quando muitas operações têm a mesma precedência (como adição e subtração), elas são computadas da esquerda para a direita:

### Exemplo

var x = 100 + 50 - 3;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_arithmetic_precedence3&usg=ALkJrhiTkETzXwfhRIHi43c9ctYDEDRz6Q)

## Valores de precedência do operador JavaScript

|  |  |  |  |
| --- | --- | --- | --- |
| **Valor** | **Operador** | **Descrição** | **Exemplo** |
| 19 | () | Agrupamento de expressões | (3 + 4) |
|  |  |  |  |
| 18 | . | Membro | person.name |
| 18 | [] | Membro | pessoa ["nome"] |
|  |  |  |  |
| 17 | () | Chamada de função | myFunction () |
| 17 | Novo | Crio | Nova data() |
|  |  |  |  |
| 16 | ++ | Incremento de Postfix | i ++ |
| 16 | - | Postfix Decrement | Eu-- |
|  |  |  |  |
| 15 | ++ | Prefixo Incremento | ++ i |
| 15 | - | Prefix Decrement | --Eu |
| 15 | ! | Lógico não | ! (x == y) |
| 15 | tipo de | Tipo | tipo de x |
|  |  |  |  |
| 14 | \* | Multiplicação | 10 \* 5 |
| 14 | / | Divisão | 10/5 |
| 14 | % | Divisão de Modulo | 10% 5 |
| 14 | \*\* | Exponenciação | 10 \*\* 2 |
|  |  |  |  |
| 13 | + | Adição | 10 + 5 |
| 13 | - | Subtração | 10 - 5 |
|  |  |  |  |
| 12 | << | Shift para a esquerda | x << 2 |
| 12 | >> | Deslocar para a direita | x >> 2 |
| 12 | >>> | Deslocar para a direita (sem assinatura) | x >>> 2 |
|  |  |  |  |
| 11 | < | Menos que | x <y |
| 11 | <= | Menor ou igual | x <= y |
| 11 | > | Melhor que | x> y |
| 11 | > = | Maior ou igual | x> = y |
|  |  |  |  |
| 10 | == | Igual | x == y |
| 10 | === | Estrita igual | x === y |
| 10 | ! = | Desigual | x! = y |
| 10 | ! == | Estrita desigual | x! == y |
|  |  |  |  |
| 6 | && | Lógico e | x && y |
| 5 | || | Lógico ou | x || y |
|  |  |  |  |
| 3 | = | Tarefa | x = y |
| 3 | + = | Tarefa | x + = y |
| 3 | - = | Tarefa | x - = y |
| 3 | \* = | Tarefa | x \* = y |
| 3 | % = | Tarefa | x% = y |
| 3 | << = | Tarefa | x << = y |
| 3 | >> = | Tarefa | x >> = y |
| 3 | >>> = | Tarefa | x >>> = y |
| 3 | & = | Tarefa | x & = y |
| 3 | ^ = | Tarefa | x ^ = y |
| 3 | | = | Tarefa | x | = y |

As entradas vermelhas claras indicam tecnologia experimental ou proposta (ECMAScript 2016 ou ES7)

As expressões entre parênteses são totalmente calculadas antes que o valor seja usado no restante da expressão.