

# Operadores e operações

**N**esta aula, vamos explorar os operadores e operações em JavaScript, ferramentas fundamentais para manipular e processar dados de forma eficaz. Operadores são símbolos que permitem realizar ações sobre variáveis e valores, como cálculos matemáticos, comparações e atribuições. Entender como esses operadores funcionam é essencial para desenvolver uma lógica de programação clara e eficiente. Nesta jornada, aprenderemos sobre operadores aritméticos, lógicos e de atribuição, compreendendo como utilizá-los corretamente e aplicá-los em diferentes situações para resolver problemas e otimizar nossos códigos.

## Definição de Operadores em Programação

Os operadores em programação são símbolos ou palavras-chave especiais que servem para realizar operações específicas em variáveis, expressões ou valores. Em JavaScript, os operadores desempenham um papel crucial, permitindo que os desenvolvedores manipulem dados de maneira eficaz e direta. A linguagem possui diversos tipos de operadores, que podem ser divididos em diferentes categorias, como aritméticos, de comparação, lógicos, de atribuição, entre outros.

Operadores aritméticos, por exemplo, realizam operações matemáticas como adição ('+'), subtração ('-'), multiplicação ('\*') e divisão ('/'). Já os operadores de comparação são usados para verificar relações entre valores, como igualdade ('=='), desigualdade ('!='), maior que ('>') e menor que ('<'). Esses operadores ajudam a controlar a lógica dos programas, criando condições para diferentes fluxos de execução.

Além disso, temos os operadores de atribuição, que permitem atribuir valores a variáveis. O mais comum é o operador igual ('='), que simplesmente atribui um valor a uma variável, mas existem outros operadores que combinam atribuição com uma operação matemática, como '+ =' (soma e atribui), '- =' (subtrai e atribui), entre outros.

Por fim, os operadores lógicos são utilizados em expressões booleanas, retornando true ou false com base em condições. Eles incluem o E lógico ('&&'), o OU lógico ('||') e a negação lógica ('!'). Esses operadores são amplamente utilizados em estruturas de controle, como as condições if.

Veja um exemplo de operadores de comparação em ação:

```
javascript
```

```
let a = 5;
```

```
let b = 10;
```

```
console.log(a > b); // false
```

```
console.log(a < b); // true
```

```
console.log(a == 5); // true
```

Nesse caso, o operador de comparação verifica se a é maior, menor ou igual a b ou ao valor 5, retornando um resultado booleano.

Entender o papel dos operadores é fundamental para quem está aprendendo programação. Eles são a base para praticamente todas as interações que fazemos com dados em nossos códigos.

## **Operadores Aritméticos**

Os operadores aritméticos são essenciais em qualquer linguagem de programação, pois permitem que realizemos cálculos matemáticos. No JavaScript, esses operadores seguem a lógica da matemática tradicional, com a adição de algumas operações específicas da programação. Vamos detalhar as principais operações e apresentar exemplos para cada uma delas.

### Adição ('+')

Esse operador soma dois valores. Ele pode ser usado tanto com números quanto com strings. No caso de strings, o '+' realiza a concatenação (junção de textos).

Exemplo:

```
javascript
```

```
let num1 = 10;
```

```
let num2 = 5;
```

```
let resultado = num1 + num2;
```

```
console.log(resultado); // 15
```

Quando usado em strings:

```
javascript
```

```
let nome = "Ana";
```

```
let sobrenome = "Silva";
```

```
let nomeCompleto = nome + " " + sobrenome;
```

```
console.log(nomeCompleto); // Ana Silva
```

## Subtração (-)

Esse operador subtrai um valor do outro, usado apenas com números.

Exemplo:

```
javascript
```

```
let num1 = 20;
```

```
let num2 = 5;
```

```
let resultado = num1 - num2;
```

```
console.log(resultado); // 15
```

## Multiplicação (\*)

Multiplica dois valores numéricos.

Exemplo:

```
javascript
```

```
let num1 = 5;
```

```
let num2 = 4;
```

```
let resultado = num1 * num2;
```

```
console.log(resultado); // 20
```

## Divisão (/)

Divide um valor pelo outro. A divisão por zero não é permitida em matemática comum, mas no JavaScript, o resultado será Infinity.

Exemplo:

javascript

```
let num1 = 10;
```

```
let num2 = 2;
```

```
let resultado = num1 / num2;
```

```
console.log(resultado); // 5
```

Se tentar dividir por zero:

javascript

```
let resultadoInvalido = num1 / 0;
```

```
console.log(resultadoInvalido); // Infinity
```

## Módulo (%)

Retorna o resto de uma divisão. Muito útil para verificar se um número é par ou ímpar, por exemplo.

Exemplo:

javascript

```
let num = 11;
```

```
let resultado = num % 2;
```

```
console.log(resultado); // 1, que significa que 11 é ímpar
```

Exponenciação (\*\*)

Esse operador eleva um número a uma potência.

Exemplo:

javascript

```
let base = 2;
```

```
let expoente = 3;
```

```
let resultado = base ** expoente;
```

```
console.log(resultado); // 8
```

Incremento (++) e Decremento (--)

Esses operadores aumentam ou diminuem uma variável em 1 unidade. Eles podem ser usados antes ou depois da variável, resultando em comportamentos diferentes.

Exemplo de incremento:

javascript

```
let contador = 5;
```

```
contador++; // ou ++contador;
```

```
console.log(contador); // 6
```

O entendimento desses operadores é crucial para manipular dados de forma eficiente em JavaScript.

## **Operadores Lógicos e Tabela Verdade**

Os operadores lógicos são ferramentas fundamentais para controlar o fluxo de um programa. Em JavaScript, os operadores lógicos trabalham com valores booleanos (true e false) e são essenciais para criar condições que orientam as ações do código.

### **E Lógico ('&&')**

Esse operador retorna true apenas se ambos os operandos forem verdadeiros. Se um dos operandos for false, o resultado será false.

Exemplo:

```
javascript
```

```
let a = true;
```

```
let b = false;
```

```
console.log(a && b); // false, porque b é falso
```

### **OU Lógico ('||')**

O operador '||' retorna true se pelo menos um dos operandos for verdadeiro. Se ambos forem falsos, ele retornará false.

Exemplo:

javascript

```
let a = true;
```

```
let b = false;
```

```
console.log(a || b); // true, pois a é verdadeiro
```

Não Lógico (!)

Esse operador inverte o valor lógico de um operando. Se ele for true, a negação (!) o torna false, e vice-versa.

Exemplo:

javascript

```
let a = true;
```

```
console.log(!a); // false, pois ! inverte o valor de a
```

## **Tabela Verdade**

A tabela verdade é uma ferramenta que organiza todas as possíveis combinações de entradas e saídas para os operadores lógicos. É muito útil para visualizar como os operadores funcionam.



Veja uma tabela verdade para o operador '&&':

A	B	A && B
-----	-----	-----
true	true	true
true	false	false
false	true	false
false	false	false

Esses operadores são amplamente usados em estruturas condicionais, como if, while e for. Entender como eles operam com base em valores booleanos ajuda a criar lógicas mais robustas em nossos códigos.

## Operadores de Atribuição

Os operadores de atribuição são responsáveis por atribuir valores a variáveis. O operador mais comum é o '=' (igual), que simplesmente atribui o valor à direita à variável à esquerda.

Exemplo simples:

```
javascript
```

```
let x = 10;
```

```
console.log(x); // 10
```

Além do simples operador de atribuição, existem os operadores de atribuição combinados, que permitem realizar operações aritméticas e

atribuir o resultado à mesma variável de forma mais eficiente. Veja alguns exemplos:

### Operador de Atribuição com Soma ('+ = ')

Esse operador soma um valor à variável e atribui o resultado a ela mesma.

Exemplo:

javascript

```
let x = 10;
```

```
x += 5; // equivalente a x = x + 5;
```

```
console.log(x); // 15
```

### Operador de Atribuição com Subtração ('- = ')

Subtrai o valor da variável e atribui o novo valor a ela.

Exemplo:

javascript

```
let x = 10;
```

```
x -= 3; // equivalente a x = x - 3;
```

```
console.log(x); // 7
```

Outros operadores

`* =` `:` Multiplicação com atribuição.

`/ =` `:` Divisão com atribuição.

`% =` `:` Módulo com atribuição.

Esses operadores são especialmente úteis para simplificar o código quando há operações repetitivas, tornando a lógica mais eficiente e legível.

Exemplo de código que usa múltiplos operadores de atribuição:

```
javascript
```

```
let y = 20;
```

```
y * = 2; // equivalente a y = y * 2;
```

```
console.log(y); // 40
```

```
y /= 4; // equivalente a y = y / 4;
```

```
console.log(y); // 10
```

Compreender e praticar o uso dos operadores de atribuição facilita a construção de códigos mais eficientes e robustos, além de promover um estilo de programação mais limpo e fácil de manter.

## **Conteúdo Bônus**

Embora muitos cursos do Coursera sejam pagos, algumas opções podem ser auditadas gratuitamente. O curso de JavaScript oferecido pela Universidade de Michigan cobre conceitos fundamentais, incluindo operadores e o desenvolvimento dinâmico.

Título: Interatividade com JavaScript

Plataforma: Coursera

### **Referência Bibliográfica**

DEITEL, P. J.; DEITEL, H. M. Ajax, rich internet applications e desenvolvimento web para programadores. Pearson: 2008

FELIX, R. (Org.). Programação orientada a objetos. Pearson: 2016

FERREIRA, R. D. Linguagem de programação. Contentus: 2020.

FLATSCHART, F.; BACHINI, C.; CUSIN, C. Open Web Platform.

Brasport: 2013.

NEVES, M. C. B. de A. Sites de Alta Performance. Contentus: 2020

PAGE-JONES, M. Fundamentos do desenho orientado a objeto com UML. Pearson: 2001.

PUGA, S.; RISSETTI, G. Lógica de programação e estruturas de dados, com aplicações em Java. Pearson: 2016.

SEGURADO, V. S. (Org.). Projeto de interface com o usuário. Pearson: 2016.

Atividade Prática 3 – Operadores e operações

Título da Prática: Aprendendo sobre operadores e operações em JavaScript

Objetivos: Fixar conceitos básicos sobre operadores e operações em JavaScript

Materiais, Métodos e Ferramentas: Computador com uma IDE instalada (recomendado: VS Code) e Javascript instalado via node/npm e o entendimento do contexto abaixo

### Atividade Prática

Os operadores são elementos essenciais em qualquer linguagem de programação, pois permitem a execução de operações em valores, variáveis e expressões. Eles são usados para realizar cálculos matemáticos, comparações lógicas, atribuir valores a variáveis e muito mais. Entender como esses operadores funcionam é a base para a construção de códigos eficientes e funcionais além da lógica de programação.

#### Tipos de Operadores Comuns:

- Operadores Aritméticos: Permitem realizar operações matemáticas, como soma (+), subtração (-), multiplicação (\*), divisão (/) e módulo (%), resto da divisão).
- Operadores Lógicos: Usados em operações lógicas, como && (E lógico), || (OU lógico) e ! (NÃO lógico), para verificar condições verdadeiras ou falsas.
  - && → E lógico: retorna verdadeiro se ambas as condições forem verdadeiras.

- $\|\rightarrow$  OU lógico: retorna verdadeiro se pelo menos uma das condições for verdadeira.
- $!$   $\rightarrow$  NÃO lógico: inverte o valor lógico de uma condição.
- Operadores de Atribuição: Utilizados para atribuir valores a variáveis. O operador básico é o  $=$ , mas também existem operadores como  $+=$ ,  $-=$ ,  $=$  e  $/=$ , *que combinam atribuição com operações aritméticas*.
  - $= \rightarrow$  *Atribuição simples (atribui o valor à variável).*
  - $+= \rightarrow$  *Soma e atribui (ex:  $x += 5$  é o mesmo que  $x = x + 5$ ).*
  - $-= \rightarrow$  *Subtração e atribui (ex:  $x -= 3$  é o mesmo que  $x = x - 3$ ).*
  - $= \rightarrow$  *Multiplicação e atribui (ex:  $x *= 2$  é o mesmo que  $x = x * 2$ ).*
  - $/= \rightarrow$  *Divisão e atribui (ex:  $x /= 4$  é o mesmo que  $x = x / 4$ ).*

De acordo com estes conceitos, responda as questões abaixo:

1. Qual dos seguintes operadores é um operador lógico em programação?

- a. +
- b. -=
- c. &&
- d. %

2. Qual operador é usado para atribuir um valor a uma variável?

- a. ==
- b. =
- c. +=
- d. !

3. O que o operador % retorna?

- a. A soma de dois números
- b. O produto de dois números
- c. O quociente de uma divisão
- d. O resto de uma divisão

4. Qual das alternativas abaixo descreve corretamente a função dos operadores em linguagens de programação?

- a. Os operadores são utilizados exclusivamente para realizar cálculos matemáticos entre valores e variáveis.
- b. Os operadores são usados apenas para comparar valores e determinar relações lógicas, como maior ou menor.
- c. Os operadores são essenciais para realizar operações em valores, variáveis e expressões, como cálculos matemáticos, comparações lógicas e atribuições de valores.
- d. Os operadores são usados apenas para atribuir valores a variáveis, não sendo capazes de realizar operações matemáticas ou lógicas.

Gabarito Atividade Prática

1. c. &&

Comentário: O operador && é usado para expressões lógicas em programação, representando o E lógico.

2. b. =

Comentário: O operador = é o operador de atribuição básico usado para definir valores em variáveis.

3. d. O resto de uma divisão.

Comentário: O operador % é o operador de módulo e retorna o resto de uma divisão entre dois números.

4. c. Os operadores são essenciais para realizar operações em valores, variáveis e expressões, como cálculos matemáticos, comparações lógicas e atribuições de valores.

Comentário: Operadores não são limitados a apenas uma função específica, mas abrangem várias tarefas, como realizar cálculos matemáticos, comparar valores, atribuir valores a variáveis, entre outras operações. A alternativa C é a mais precisa, pois abrange as principais funcionalidades dos operadores em programação.

**Ir para exercício**