

# JavaScript

Para facilitar seu aprendizado, tente fazer os exercícios mentalmente e somente em caso de dúvidas utilize alguma ferramenta, como o terminal de comando com o Node.js rodando, Console do navegador ou sites como JS Fiddle:

## Operadores

Operador	Função
Aritméticos: retornam o resultado de uma operação	
+ - * / % ++ --	somar subtrair multiplicar dividir resto de divisão incremento decremento
Comparadores matemáticos: teste lógico, retorno booleano (true / false)	
< > <= >=	menor que maior que menor ou igual maior ou igual
Comparadores Lógicos: teste lógico, retorno booleano (true / false)	
== != === !==	igualdade entre sentenças (valor) diferença entre sentenças (valor) igualdade entre sentenças (valor e tipo) diferença entre sentenças (valor e tipo)
Operadores de lógica e junção lógica	
! && 	NÃO (NOT) E (AND) OU (OR)

O sinal de exclamação (!) é o operador NOT (não), utilizado para negar a sentença que vem na sequência.

Exemplos:

```
a != b      // o valor de a é diferente de b
x !== y     // o valor e o tipo de x são diferentes de y
!a == b     // o valor de a não é igual ao valor de b
```

As condições lógicas são convertidas em números binários:

**true** é equivalente a 1

**false** é equivalente a 0

## Operador lógico de atribuição

Tem a capacidade de atribuir valor a uma variável a partir de uma condição lógica, economiza IFs

Exemplo:

```
var meuCarro = cor == "preto" ? "preto" : "branco";
```

## Exercícios:

Preencha os resultados das operações e o tipo de dado

### Exemplos

```
8 + 6 = 14 (number)
"8" + "6" = "86" (string)
"8.6" + 4 = "8.64" (string)
"8" * 4 = 32 (number)
"8" - 4 = 4 (number)
"8" / 3 = 2.6666666666666665 (float)
5 + true = 6 (number)
"teste" + true = "testetrue" (string)
"8" == 8 = true (boolean)
"8" == 4 = false (boolean)
8 === "8" = false (boolean)
8 !== "8" = true (boolean)
8 < 4 = false (boolean)
8 > 4 = true (boolean)
```

## Exercícios:

### 1. Resolva as operações:

- $10 + 15 = 25$  (number)
- $"10" + 2 = 102$  (string)
- $"10" * 2 = 20$  (number)
- $"10" / 3 = 3.3333333333333335$  (float)
- $"10" \% 3 = 1$  (number)
- $10 + \text{true} = 11$  (number)
- $10 == "10" = \text{true}$  (boolean)
- $10 === "10" = \text{false}$  (boolean)
- $10 < 11 = \text{true}$  (boolean)
- $10 > 12 = \text{false}$  (boolean)
- $10 \leq 10.1 = \text{true}$  (boolean)
- $10 > 9.99 = \text{true}$  (boolean)
- $10 \neq \text{"dez"} = \text{true}$  (boolean)
- $10 + \text{true} = 11$  (number)
- $\text{"dez"} + \text{true} = \text{"deztrue"}$  (string)
- $10 + \text{false} = 10$  (number)
- $10 * \text{false} = 0$  (number)
- $\text{true} + \text{true} = 2$  (number)
- $10++ = \text{aumenta}$
- $10-- = \text{diminui}$
- $1 \& 1 = 1$  (number)
- $1 \& 0 = 0$  (number)
- $0 \& 0 = 0$  (number)
- $1 \& 0 = 0$  (number)
- $0 / 1 = 0$  (number)
- $5 + 5 == 10 = \text{true}$  (boolean)
- $"5" + "5" == 10 = \text{false}$  (boolean)
- $"5" * 2 > 9 = \text{true}$  (boolean)
- $(10 + 10) * 2 = 40$  (number)
- $10 + 10 * 2 = 30$  (number)

### 2. Responda as perguntas de acordo com as variáveis.

```
var branco = "preto";  
var preto = "cinza";  
var cinza = "branco";  
var carro = "preto";  
var valor = 30000;  
var prestacao = 750;
```

- a) `branco == "branco" = false (boolean)`
- b) `branco == cinza = false (boolean)`
- c) `carro == branco = true (boolean)`
- d) `var cavalo = carro == "preto" ? "cinza" : "marron"; cavalo = "cinza"`
- e) Quantas prestações são necessárias para pagar o valor do carro com uma entrada de 3.000? Demonstre a operação.  
$$R = \text{var entrada} = 3000;$$
$$(\text{valor} - \text{entrada}) / 750;$$
$$36 \text{ prestações}$$
- f) Somando as variáveis de cores é formada uma string de quantos caracteres?  
`R = pretocinzabrancopreto = 21 caracteres`