

# Operadores Lógicos:

Vamos lá?

Conhecemos alguns operadores, como os **aritméticos**: `=`, `*`, `/`, `-` (Soma, Multiplicação, Divisão e Subtração) que vão nos permitir fazer operações matemáticas simples. E, conhecemos nesse módulo os operadores **relacionais**, por exemplo: `==`, `!=`, `>=`, `>` (Igualdade simples, Diferente, Maior ou igual, Maior) que nos permitiu criar condições em conjunto com o IF.

Porém, temos mais operadores para conhecer. Prometemos que esses serão os últimos! Iremos falar sobre operadores lógicos, e como eles podem ajudar em nossas condições.

## Operador And - &&

Esse operador nos permite **validar DUAS ou mais condições**. E ele só irá dar “verdadeiro” se as duas forem “verdadeiro” também. Veja o exemplo:

```
var dia = "domingo"
var tempo = "bom"

if(dia == "domingo" && tempo == "bom"){
    console.log("Hoje é dia de festa!")
}else{
    console.log("Hoje é um dia normal!")
}
```

Nesse exemplo, ele só irá imprimir **“Hoje é dia de festa”** caso o dia for domingo **E** se o tempo estiver bom. Se qualquer uma dessas alternativas fosse “falso”, seria executada a ação do else.

O operador AND também nos ajuda muito. Ele **evita alguns famosos “if alinhados”**, que são IF dentro de IF, já que podemos fazer duas condições em uma apenas usando o operador!

## Operador Or - ||

O operador OR nos ajuda quando podemos ter **mais opções**, e sendo qualquer uma delas verdadeira, já é o suficiente para executar uma ação. Pense que você está querendo um iPhone. Se a cor for prata ou preto tudo bem, porém qualquer coisa diferente disso você não irá querer. Veja isso no código:

```
var corIphone = "Preto"

if(corIphone == "Preto" || corIphone == "Prata"){
    console.log("Vou fechar a compra")
}else {
    console.log("Não quero mais esse iPhone")
}
```

Nesse caso, se a cor atende **qualquer uma** das condições, irá exibir **“Vou fechar a compra”**. Já se a cor for diferente de preto **OU** prata, será exibido **“Não quero mais esse iPhone”**.

Um ponto a se destacar é que, mesmo que as comparações sejam feitas na mesma variável, é necessário repetir o nome dela. Isso acontece porque cada condição é totalmente independente, e isso vale para todas as operações independente do operador!

```
//código errado
if(corIphone == "Preto" || "Prata")

//Código certo
if(corIphone == "Preto" || corIphone == "Prata")
```

## Operador NOT - !

Conhecido como **operador de negação**, serve para quando **queremos o valor falso da nossa condição**. Ele é representado pelo símbolo ! (Exclamação) e quando usado em uma variável, ele sempre vai antes do valor da mesma!

```
var feriado = false

if(!feriado){
    console.log("tenho que trabalhar!")
}else {
    console.log('Oba posso ficar em casa!')
}
```

Note que ele só irá executar a ação do IF caso o valor da variável feriado **for falso**, caso contrário, ele irá executar a ação do else. Além disso, observe que usamos o operador de negação (!) antes do nome da variável.