

```
//OPERADORES LÓGICOS:
//Operadores lógicos são usados quando desejamos fazer comparações lógicas entre 2 operandos ou sobre um único operando. Os operadores lógicos que nós utilizamos são o AND (Lógico e Binário), OR (Lógico e Binário), NOT (Lógico) e XOR (Apenas Binário).

//EXEMPLOS DE OPERADORES LÓGICOS:
const [a, b] = [true, false];
console.log(`1) (Operador AND) ${a} && ${a} = ${a && a}`); //Operador && é sempre lógico...
console.log(`1) (Operador AND) ${a} && ${b} = ${a && b}`);
console.log(`1) (Operador AND BINÁRIO) ${a} & ${a} = ${a & a}`); //O operador & traz o resultado binário 1(true) ou 0(false)...
console.log(`1) (Operador AND BINÁRIO) ${a} & ${b} = ${a & b}`);

console.log(`\n2) (Operador OR) ${a} || ${a} = ${a || a}`); //Operador || sempre trará um resultado lógico booleano...
console.log(`2) (Operador OR) ${a} || ${b} = ${a || b}`);
console.log(`2) (Operador OR) ${b} || ${b} = ${b || b}`);
console.log(`2) (Operador OR) ${a} | ${a} = ${a | a}`); //O operador | trará um resultado binário 1(true) ou 0(false)...
console.log(`2) (Operador OR) ${a} | ${b} = ${a | b}`);
console.log(`2) (Operador OR) ${b} | ${b} = ${b | b}`);

console.log(`\n3) (Operador NOT) !${a} = ${!a}`); //O operador de negação troca o valor booleano...
console.log(`3) (Operador NOT) !${b} = ${!b}`);

console.log(`\n4) (Operador CONVERSÃO BOOLEANA) !!1 = ${!!1}`); //O operador de dupla negação somente transforma um resultado 0(false) e qualquer valor positivo e negativo que não seja zero em valores booleanos (true)...
console.log(`4) (Operador CONVERSÃO BOOLEANA) !!0 = ${!!0}`);

console.log(`\n5) (Operador XOR BINÁRIO) ${a} ^ ${b} = ${a ^ b}`); //O operador XOR só trará resultados binários...
console.log(`5) (Operador XOR BINÁRIO) ${b} ^ ${b} = ${b ^ b}`);

//EXEMPLO DA APLICAÇÃO DE OPERADORES LÓGICOS NUMA FUNÇÃO:
//Uma função que determina que: Se recebermos 2 salários vamos comprar uma TV de 50" e tomar sorvete, mas se recebermos 1 salário só vamos comprar 1 TV de 32" e tomar sorvete, porém, se não recebermos nada não vamos comprar TV nenhuma nem comprar sorvete...
function comprar(salario1, salario2) {
    const comprarSorvete = salario1 || salario2; //Independente do salário, se recebermos pelo menos 1 vamos tomar sorvete...
    const comprarTV50 = salario1 && salario2; //A TV de 50" só será comprada se recebermos 2 salários...
```

```

    const comprarTV32 = !(salario1 ^ salario2); //A TV de 32 será comprada só se recebermos 1 salário ou outro...
    const manterSaudavel = !comprarSorvete; //Vamos manter nosso corpo saudável só se não comprarmos sorvete nenhum...
    return { //Perceba que no retorno de funções, quando criamos objetos, á partir de variáveis dentro da função, não precisamos de r
referenciar seus nomes para depois capturar os seus valores, por padrão podemos colocar somente o nome da variável que automaticamente
    ela se transforma na nossa chave e o seu valor vêm acompanhado dela...
        comprarSorvete,
        comprarTV50,
        comprarTV32,
        manterSaudavel}
}
console.log('\n6) Salario1 = true e Salario2 = true', compras(true, true));
console.log('\n7) Salario1 = true e Salario2 = false', compras(true, false));
console.log('\n8) Salario1 = false e Salario2 = true', compras(false, true));
console.log('\n9) Salario1 = false e Salario2 = false', compras(false, false));

```

RESULTADO NO CONSOLE...

```

[Running] node "c:\Users\User\Documents\JAVASCRIPT\ARQUIVOS_DAS_AULAS\048-Operadores_Logicos.js"
1) (Operador AND) true && true = true
1) (Operador AND) true && false = false
1) (Operador AND BINÁRIO) true & true = 1
1) (Operador AND BINÁRIO) true & false = 0

2) (Operador OR) true || true = true
2) (Operador OR) true || false = true
2) (Operador OR) false || false = false
2) (Operador OR) true | true = 1
2) (Operador OR) true | false = 1
2) (Operador OR) false | false = 0

3) (Operador NOT) !true = false
3) (Operador NOT) !false = true

4) (Operador CONVERSÃO BOOLEANA) !!1 = true
4) (Operador CONVERSÃO BOOLEANA) !!0 = false

```

```
5) (Operador XOR BINÁRIO) true ^ false) = 1
5) (Operador XOR BINÁRIO) false ^ false) = 0
```

```
6) Salario1 = true e Salario2 = true {
  comprarSorvete: true,
  comprarTV50: true,
  comprarTV32: false,
  manterSaudavel: false
}
```

```
7) Salario1 = true e Salario2 = false {
  comprarSorvete: true,
  comprarTV50: false,
  comprarTV32: true,
  manterSaudavel: false
}
```

```
8) Salario1 = false e Salario2 = true {
  comprarSorvete: true,
  comprarTV50: false,
  comprarTV32: true,
  manterSaudavel: false
}
```

```
9) Salario1 = false e Salario2 = false {
  comprarSorvete: false,
  comprarTV50: false,
  comprarTV32: false,
  manterSaudavel: true
}
```

```
[Done] exited with code=0 in 0.174 seconds
```