

OPERADORES RELACIONAIS, OPERADORES LÓGICOS E CONTROLE DE FLUXO (CONDICIONAIS)

(AULA 4)

CURSO BÁSICO DE PROGRAMAÇÃO COM JAVASCRIPT

MAYARA MARQUES

mmrosatab@gmail.com

SUMÁRIO

- Operadores relacionais
- Operadores lógicos
- Controle de fluxo - Condicionais
 - If
 - If-else
 - If-else-if
 - Switch
- Mão na massa

OPERADORES RELACIONAIS

Operadores relacionais comparam duas ou mais variáveis e determinam seu relacionamento com um resultado lógico, ou seja, true (para verdadeiro) ou false (para falso).

OPERADORES RELACIONAIS

OPERADOR	SINTAXE	FUNÇÃO
>	$a > b$	Avalia se a variável a é maior que a variável b (e retorna true ou false, dependendo dos valores de a e b)
>=	$a \geq b$	Avalia se a variável a é maior ou igual à variável b (e retorna true ou false, dependendo dos valores de a e b)
<	$a < b$	Avalia se a variável a é menor que a variável b (e retorna true ou false, dependendo dos valores de a e b)
<=	$a \leq b$	Avalia se a variável a é menor ou igual à variável b (e retorna true ou false, dependendo dos valores de a e b)

OPERADORES RELACIONAIS

OPERADOR	SINTAXE	FUNÇÃO
<code>==</code>	<code>a == b</code>	Avalia se o conteúdo da variável a é igual à variável b (e retorna true ou false, dependendo dos valores de a e b)
<code>===</code>	<code>a === b</code>	Avalia se o conteúdo e o tipo da variável a é igual à variável b (e retorna true ou false, dependendo dos valores de a e b)
<code>!=</code>	<code>a != b</code>	Avalia se a variável a é diferente da variável b (e retorna true ou false, dependendo dos valores de a e b)
<code>!==</code>	<code>a !== b</code>	Avalia se o conteúdo e o tipo da variável é diferente da variável b (e retorna true ou false, dependendo dos valores de a e b)

OPERADORES LÓGICOS

Operadores lógicos avaliam um ou mais operandos lógicos que geram um único valor lógico (true ou false) como resultado final da expressão avaliada.

OPERADORES LÓGICOS

OPERADOR	SINTAXE	FUNÇÃO
&&	(a > b) && (a > d)	AND - E lógico
	(a > b) (a > d)	OR - Ou lógico
!	!a	NOT - negação Nega logicamente uma expressão ou variável

OPERADORES LÓGICOS

- `&&` (and lógico) e `&` (and binário):

OPERANDO A	OPERANDO B	RESULTADO
verdadeiro	verdadeiro	verdadeiro
verdadeiro	falso	falso
falso	verdadeiro	falso
falso	falso	falso

- `||` (or lógico) e `|` (or binário):

OPERANDO A	OPERANDO B	RESULTADO
verdadeiro	verdadeiro	verdadeiro
verdadeiro	falso	verdadeiro
falso	verdadeiro	verdadeiro
falso	falso	falso

OPERADORES LÓGICOS

- ! (negação):

OPERANDO A	RESULTADO
verdadeiro	falso
falso	verdadeiro

CONTROLE DE FLUXO - CONDICIONAIS

Estruturas que permitem a seleção de partes do código para execução. Possibilitam fazer desvios para executar um trecho de código ou não. (estruturas de decisão)

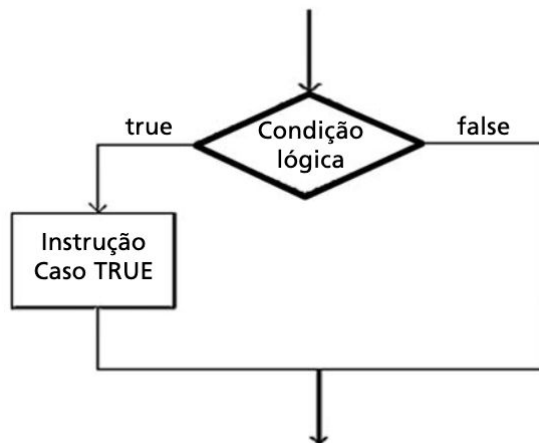
Existem quatro formas de estruturas de decisão em Javascript:

- if
- if-else
- if-else-if
- switch

CONDICIONAIS

- If

A declaração if dentro de um bloco de comando determina que as instruções daquele bloco só serão executadas se a expressão lógica associada ao if for verdadeira.



CONDICIONAIS

```
if (condition)
```

```
    instruction
```

```
if (condition) {
```

```
    // block of code to be executed if the condition is true
```

```
    // instructions
```

```
}
```

CONDICIONAIS

```
if (18 > 16) {  
  console.log("18 is greater than 16")  
}
```

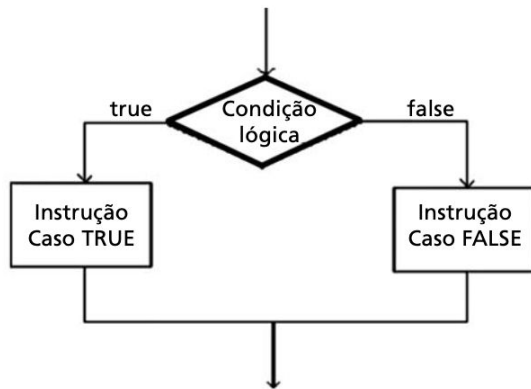
```
let ninhoPrice = 18
```

```
if (ninhoPrice <= 18) {  
  console.log("Buy Ninho")  
}
```

CONDICIONAIS

- If-else

A declaração if-else em dois blocos de comando associados determina que as instruções dentro do bloco ao qual o if pertence serão executadas se a expressão lógica associada ao if for verdadeira. As instruções dentro do bloco ao qual o else pertence serão executadas se a expressão for falsa.



CONDICIONAIS

```
if (condition) {  
    // block of code to be executed if the condition is true  
} else {  
    // block of code to be executed if the condition is false  
}
```

CONDICIONAIS

```
let ninhoPrice = 20

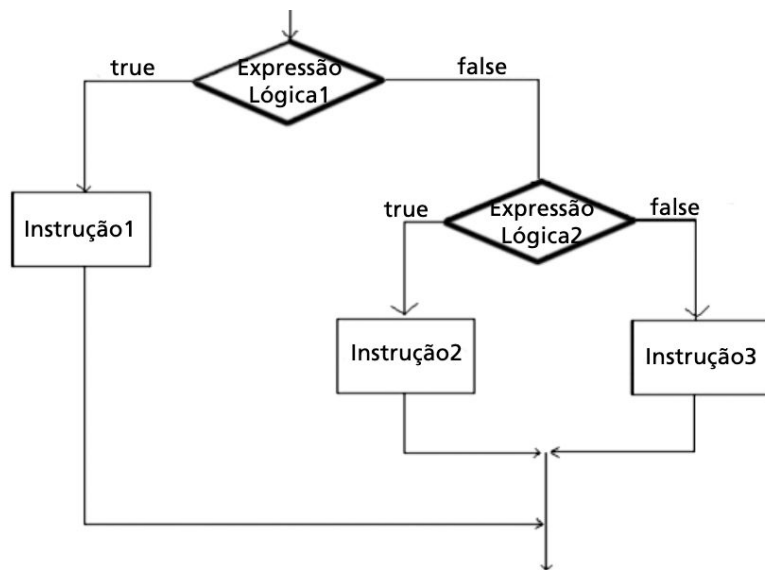
if (ninhoPrice <= 18) {
  console.log("Buy Ninho.")
} else {
  console.log("Buy Elegê.")
}

// Outputs "Buy Elegê."
```


CONDICIONAIS

- If -else-if

Utilizado para fluxos mais complexos, com intuito de tratar mais de dois fluxos possíveis.



CONDICIONAIS

```
if (condition1) {  
    // block of code to be executed if condition1 is true  
} else if (condition2) {  
    // block of code to be executed if the condition1 is false and condition2 is true  
} else {  
    // block of code to be executed if the condition1 is false and condition2 is false  
}
```

CONDICIONAIS

```
let ninhoPrice = 20

let elegePrice = 16

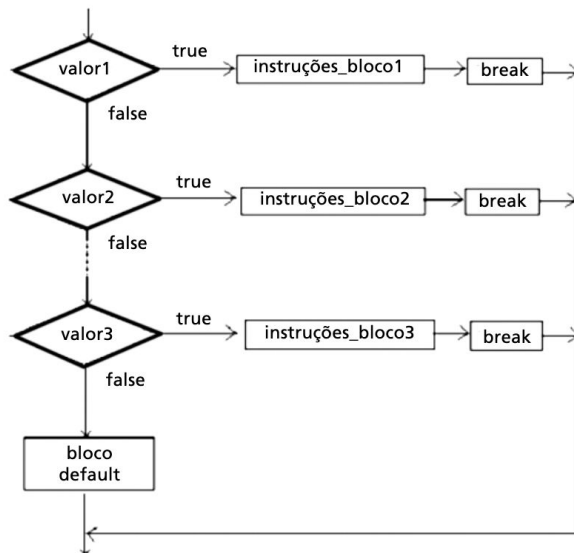
if (ninhoPrice <= 18) {
  console.log("Buy Ninho")
} else if (elegePrice <= 16) {
  console.log("Buy Elegê.")
} else {
  console.log("Buy Italac.")
}

// Outputs "Buy Elegê."
```

CONDICIONAIS

- Switch

Uma outra opção ao if-else é o Switch. Ele permite selecionar um bloco de código a ser executado.



CONDICIONAIS

```
switch (expression) {  
  case x:  
    // code block  
    break  
  case y:  
    // code block  
    break  
  default:  
    // code block  
}
```

CONDICIONAIS

```
let day = 4

switch (day) {

  case 6:

    console.log("Today is Saturday")

    break

  case 7:

    console.log("Today is Sunday")

    break

  default:

    console.log("Looking forward to the Weekend")

}

// Outputs "Looking forward to the Weekend"
```

CONDICIONAIS

- Operador ternário

O operador ternário em JavaScript é uma **forma concisa de escrever uma estrutura condicional** que avalia uma expressão e retorna um valor com base nessa avaliação.

CONDICIONAIS

- Operador ternário
 - Sintaxe
 - condição ? valor_se_verdadeiro : valor_se_falso
 - A condição é avaliada primeiro.
 - Se a condição for verdadeira, o valor após o ? (valor_se_verdadeiro) é retornado.
 - Se a condição for falsa, o valor após o : (valor_se_falso) é retornado.

CONDICIONAIS

- Operador ternário

Exemplo

```
const numero = 7

const resultado = numero % 2 === 0 ? "Par" : "Ímpar"

console.log(resultado) // Output: "Ímpar"
```

CONDICIONAIS

- Operador ternário

Exemplo

```
const numero = 10

const resultado = numero > 0 ? "Positivo" : (numero < 0 ? "Negativo" :
"Zero")

console.log(resultado) // Output: "Positivo"
```



MÃO NA MASSA

MÃO NA MASSA



1. Crie um programa que lê duas notas de prova, calcule a média e imprima a palavra 'APROVADO(A)' se a média for maior ou igual a 7 e 'REPROVADO(A)' se a média for menor de 7.

2. Crie um programa que, dado um número inteiro lido via prompt, classifique qual o dia da semana é. Baseie-se na associação abaixo:

DOMINGO = 1, SEGUNDA = 2, TERÇA = 3, QUARTA = 4, QUINTA = 5, SEXTA = 6 e SÁBADO = 7

OBS: Caso o usuário digite um valor fora do intervalo de 1 a 7, o programa deve exibir uma mensagem de erro.

3. Crie um programa que, dado um número inteiro lido via prompt, classifique se é um dia de final de semana ou não. Baseie-se na associação abaixo:

DOMINGO = 1, SEGUNDA = 2, TERÇA = 3, QUARTA = 4, QUINTA = 5, SEXTA = 6 e SÁBADO = 7

OBS: Caso o usuário digite um valor fora do intervalo de 1 a 7, o programa deve exibir uma mensagem de erro.

MÃO NA MASSA

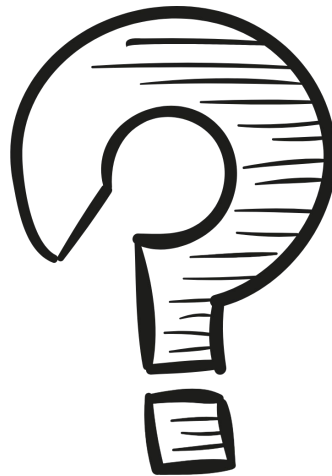


4. Cria um programa que, dado um número inteiro, classifique se este número é par ou ímpar.
5. Crie um programa que lê dois números inteiros e classifique qual é o maior.
6. Crie um programa que lê três números inteiros e classifique qual é o maior.



7. O que será impresso no console após a execução do código abaixo?

```
var texto = "Yennefer and Geralt"  
  
console.log(texto)  
  
if(5 + 5 === 10){  
    var texto = "Siri"  
    console.log(texto)  
}  
  
console.log(texto)
```





8. O que será impresso no console após a execução do código abaixo?

```
var texto = "Yennefer and Geralt"

console.log(texto)

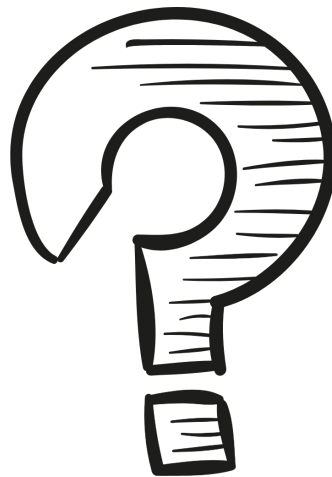
if(5 + 5 === 10){

    let texto = "Siri"

    console.log(texto)

}

console.log(texto)
```



REFERÊNCIAS

- Apostila Caelum Estruturação de páginas usando HTML e CSS
- <https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript>
- <https://www.w3schools.com/js/default.asp>
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide/Expressions_and_operators