

CONTROLE DE FLUXO - LAÇOS DE REPETIÇÃO ***(AULA 5)***

CURSO BÁSICO DE PROGRAMAÇÃO COM JAVASCRIPT

MAYARA MARQUES

SUMÁRIO

- Laços de repetição
 - While
 - Do while
 - For
 - While vs For
- Declaração continue
- Declaração break
- Mão na massa

LAÇOS DE REPETIÇÃO

Laços de repetição ou estruturas de repetição são recursos que permitem que um determinado trecho de código seja executado várias vezes.

LAÇOS DE REPETIÇÃO

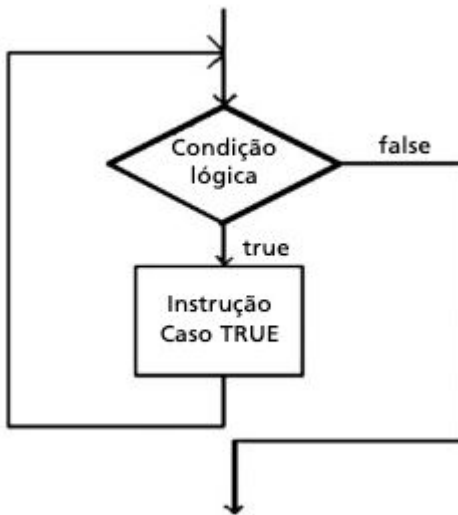
Algumas das estruturas de repetição do Javascript

- While
- Do While
- For

LAÇOS DE REPETIÇÃO

- While

O while permite executar um bloco de código **enquanto** uma condição está sendo satisfeita.



LAÇOS DE REPETIÇÃO

- *While*

```
while (condition) {  
    // code block to be executed  
}
```

LAÇOS DE REPETIÇÃO

- *While*

```
let i = 0

while (i < 5) {

  console.log(i)

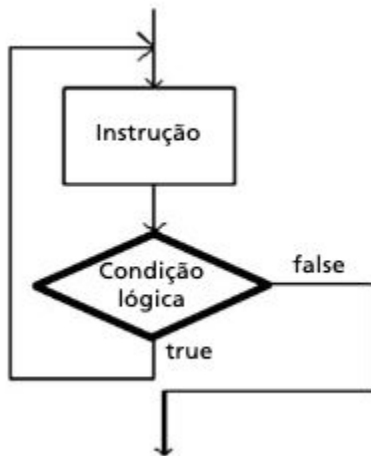
  i++

}
```

LAÇOS DE REPETIÇÃO

- Do while

O do while permite executar um bloco de código **enquanto** uma condição está sendo satisfeita, contudo a condição é testada **após** o bloco de código.



LAÇOS DE REPETIÇÃO

- Do while

```
do {  
  
    // code block to be executed  
  
}while (condition)
```

LAÇOS DE REPETIÇÃO

- Do while

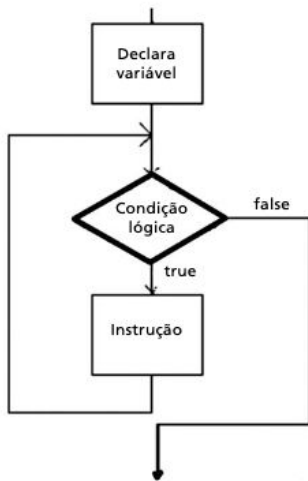
```
let i = 0

do {
  console.log(i)
  i++
}while (i < 5)
```

LAÇOS DE REPETIÇÃO

- For

Similar ao while e ao do while, permite executar um bloco de comando repetidamente segundo uma condição.



LAÇOS DE REPETIÇÃO

- For

```
for (statement 1; statement 2; statement 3) {  
    // code block to be executed  
}
```

LAÇOS DE REPETIÇÃO

- For

```
for (let i = 0; i < 5; i++) {  
  console.log(i)  
}
```

While vs For

- A principal diferença entre o while e for está na forma como eles iteram ou repetem um conjunto de instruções. Enquanto o **for repete um conjunto de instruções em um número específico de vezes**, o while nos dá a possibilidade de não definir exatamente a quantidade de vezes que um bloco de instruções será executado. No while, **a expressão booleana associada à condição é quem determinará a quantidade de vezes que o mesmo será executado.**

DECLARAÇÃO CONTINUE

- Continue

A declaração continue é utilizada para **saltar a interação corrente** de uma estrutura de repetição (while, do while ou for).

DECLARAÇÃO CONTINUE

```
for (let i = 0; i < 10; i++)  
{  
  if (i == 4) {  
    continue  
  }  
  console.log(i)  
}
```

```
let i = 0  
while (i < 10) {  
  
  if (i == 4) {  
    i++  
    continue  
  }  
  console.log(i)  
  i++  
  
}
```


DECLARAÇÃO BREAK

- Break

A declaração break é utilizada para **interromper o bloco de comandos e sair** de uma estrutura de repetição (while, do while ou for) ou de uma estrutura de decisão switch case.

DECLARAÇÃO BREAK

```
let i = 0

while (i < 10) {
  console.log(i)
  i++
  if (i == 4) {
    break
  }
}
```

```
for (let i = 0; i < 10; i++)
{
  if (i == 4) {
    break
  }
  console.log(i)
}
```



MÃO NA MASSA

MÃO NA MASSA



1. Crie um programa que imprima todos os números de 150 a 300.
2. Crie um programa que imprima a soma de 1 até 100.
3. Crie um programa que imprima todos os números pares de 1 até 100.
4. Crie um programa que imprima todos os números ímpares de 1 até 100.
5. Crie um programa que imprima todos os múltiplos de 3, entre 1 e 100.
6. Crie um programa que calcule o fatorial de um número lido.
7. Crie um programa que determine se um número lido é primo ou não.
8. Crie um programa que calcule e imprima o fatorial dos números entre 1 e 10.
9. Crie um programa que determine e imprima os números primos entre 1 e 20.

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