



LÓGICA DE PROGRAMAÇÃO JAVASCRIPT



ALEX SANDER RESENDE DE DEUS PROFESSOR

- Há 25 anos ensinando programação, é um apaixonado por tecnologia. Atualmente é coordenador de cursos na ETEC Albert Einstein do Centro Paula Souza.
- É professor da FIAP School e FIAP Corporate, lecionando C#, Banco de Dados e Desenvolvimento Mobile.

AGENDA

1 AULA 1 Definição e histórico. Fundamentos Básicos da Programação – Tipos de dados, variáveis e estruturas básicas; Usando o GIT

2 AULA 2 Estruturas condicionais, operadores lógicos e switch case

AULA 3 Laços de Repetição: For, while

AULA 4 Objetos nativos JavaScript – Date, String, Math

AGENDA

AULA 5 Arrays, Filter, Map Reduce, Splite, for Each, for in, for of **AULA 6** Funções: tipos, declarações, escopo, retorno, clousers, call-back, IIFE, factory **AULA 7** DOM – Document Object Model **AULA 8** DOM – Document Object Model

AGENDA

9 AULA 9 Eventos

Objetos: Criação, prototype, getters, setters

AULA 10 Classes JavaScritp Assíncrono

AULA 3

LAÇOS DE REPETIÇÃO



Uma das situações mais recorrentes na vida de um desenvolvedor de sistemas é precisar que um determinado trecho do código seja repetido diversas vezes.

Quem resolve essa questão são os loops. Loops são estruturas de programação capazes de repetir a execução de um determinado trecho do código.

Está difícil de imaginar a utilidade? Então vamos ver um caso prático.

LOOP WHILE



```
while (condição) {
     //instruções que serão repetidas enquanto a condição for verdadeira
}
```

A sintaxe do loop While permite que indiquemos uma condição e um conjunto de instruções que devem ser repetidas enquanto a condição for verdadeira.

COM UM LAÇO DE REPETIÇÃO VEJA A DIFERENÇA DE CÓDIGO

```
JS script.js > ...
    var num=parseInt(prompt("Digite um número para ver a tabuada"));
   var cont=1;
   while(cont<=10){
        let res=num*cont;
        document.write(res.toString() + "<br>");
        cont++;
                            !DOCTYPE html>
                        1
8
                             <html lang="pt-br">
                                 <head>
                                     <title>Desvios condicionais</title>
                                 </head>
                            <body>
                                 <script type="text/javascript" src="js/script.js"></script>
                             </body>
                            </html>
                       11
```

LOOP FOR

Já sabemos o que são Loops: estruturas de programação capazes de repetir a execução de um determinado trecho do código.

Enquanto um loop While é baseado em uma condição e é executado enquanto a condição é satisfeita, temos diferentes tipos de loops para diferentes cenários (alguns que só exploraremos no curso de Orientação a Objetos, por exemplo).

Hoje conheceremos o loop For, que é baseado em um contador.

VAMOS REFAZER O MESMO PROGRAMA (TABUADA) UTILIZANDO O LOOPING FOR

```
JS script.js > ...
    var num=parseInt(prompt("Digite um número para ver a tabuada"));
    for (let cont=1;cont<=10;cont++){
        let res=num*cont;
        document.write(res.toString() + "<br>");
                              !DOCTYPE html>
7
                              <html lang="pt-br">
                                  <head>
                                      <title>Desvios condicionais</title>
                                  </head>
                              <body>
                                   <script type="text/javascript" src="js/script.js"></script>
                              </body>
                              </html>
                          10
                          11
```



```
for (variavel contadora; condição de parada; incremento){
    //instruções que serão repetidas
}
```

Essa é a sintaxe do loop For. Com ele, o programador não precisa se preocupar em criar antecipadamente a variável contadora e nem corre o risco de esquecer o incremento e acabar criando um loop infinito.



```
for (variavel contadora; condição de parada; incremento){
    //instruções que serão repetidas
}
```

No primeiro parâmetro desse loop devemos indicar qual é a variável contadora e qual é seu valor inicial.

Se indicarmos algo como i=0, a variável i será nossa contadora e iniciará valendo zero, porém terá que ser criada anteriormente pelo programador.

Se indicarmos int i=0, a variável i será criada automaticamente no início do loop e destruída após ele, economizando espaço em memória.



```
for (variavel contadora; condição de parada; incremento){
    //instruções que serão repetidas
}
```

A condição é similar à que escreveríamos em um loop while.

A lógica é: enquanto a condição for verdadeira, o loop continua sendo executado.



for (variavel contadora; condição de parada; incremento){
//instruções que serão repetidas

O incremento, por sua vez, deve apresentar quanto será somado ou subtraído da variável contadora a cada volta do loop.

Se escrevermos i++, i=i+1, i=i+2 ou qualquer outra adição, a variável será incrementada a cada volta do loop.

Se escrevermos i--, i=i-1, i=i-2, por outro lado, a variável será decrementada a cada volta do loop.

LOOP DO WHILE

Um loop do while é uma instrução de fluxo de controle que executa um bloco de código pelo menos uma vez e depois executa o bloco repetidamente ou para de executá-lo, dependendo de uma determinada condição booleana no final do bloco.

VAMOS AGORA ANALISAR O PROGRAMA TABUADA UTILIZANDO O LOOPING DO WHILE

```
JS script.js > ...
   var num=parseInt(prompt("Digite um número para ver a tabuada"));
   var cont=1;
   dof
       let res=num*cont;
       document.write(res.toString() + "<br>");
       cont++
   }while(cont<=10);</pre>
                              DOCTYPE html>

<html lang="pt-br">

                                  <head>
                                       <title>Desvios condicionais</title>
                                   </head>
                          7 v <body>
                                   <script type="text/javascript" src="js/script.js"></script>
                              </body>
                         10
                               </html>
```

MELHORANDO O PROJETO



Agora nosso script está em uma função.

```
function calcularTabuada() {
         var num = parseInt(document.getElementById("num").value);
         var cont = 1;
         var result = "";
                                                                 Lendo o conteúdo da input
         do {
                                                                         no html.
             let res = num * cont;
             result += res + "<br>";
             cont++
         } while (cont <= 10);
         document.getElementById("resultado").innerHTML = result;
10
                                                              Criação de uma string
 Exibição do resultado na
                                                                 com o resultado.
      tag p do html.
```

MOMENTO HANDS ON

Utilizando o looping while, crie um programa que leia um número qualquer fornecido pelo usuário e calcule o fatorial deste número. Exemplo:

5!=5x4x3x2x1=120

Utilizando o looping for, elabore um programa que leia dois números quaisquer e exiba o somatório de todos os números que estão entre os dois digitados. Lembre-se que o primeiro número deve ser sempre menor que o segundo.

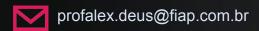
Utilizando o looping do while, elabore um programa que leia dois números quaisquer e exiba a quantidade de números pares entre os dois valores digitados. Lembre-se que o primeiro número deve ser sempre menor que o segundo.

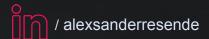
LEMBRE-SE DE VERSIONAR SEU PROJETO NO GIT

REFERÊNCIAS

- https://developer.mozilla.org/pt-BR/docs/Aprender/JavaScript
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/String
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Math
- https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Array
- https://developer.mozilla.org/pt-BR/docs/DOM/Referencia_do_DOM
- https://www.w3schools.com/

OBRIGADO







Copyright © 2021 | Professor Alex Sander Resende de Deus

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.

