



Workshop Javascript

@wwcoderecife

CRONOGRAMA

Welcome.

O que são algoritmos?

O que é JavaScript?

Variáveis.

Arrays.

Objetos.

Estruturas de decisão (if/else).

Estruturas de repetição (for/map/filter).

Funções.

Desafio final.

Quem Sou eu?

*Simone
Amorim*



@simoneas02



@samorim02



simoneas02.github.io



Quem Sou eu?

*Willany
Silva*



@willany



@willanyasilva



Algoritmos??



Ada Lovelace: 1842

Algoritmos



Algoritmos

Comprar um vestido

- Entro na loja
- Começo a procurar um vestido que me agrade
- Enquanto não acho, continuo procurando firme e forte
- Achei um bonitinho, vou provar né
- Se ficar bom eu levo, senão eu vou continuar procurando
- Eita, prestou não :(
- Continuo procurando
- Achei outro e tá na promoção, tô sentindo que agora vai
- Vou provar de novo, se ficar bom eu levo, senão vou continuar procurando
- Esse ficou #topper :)
- Vou comprar!

Tempo

Como vocês fazem para chegar nas aulas do WDP?



2



- **Brendan Eich: 1995**
- É atualmente a principal linguagem para programação client-side em navegadores web.
- JavaScript permite páginas da Web interativas e, portanto, é uma parte essencial dos aplicativos da web.
- Versão atual: **ES9**

What's??

ES6/2015, ES7/2016, ES8...

variáveis



cômoda = memória do computador
gaveta = label
conteúdo = valor

variáveis

```
var nome = 'Beyonce';
```

palavra reservada

label

valor

variáveis

- **Palavras:** embora a maioria dos navegadores já reconheça uma variedade de caracteres UTF-8 (como palavras com acentos e "ø_ø", por exemplo), o recomendado é o uso apenas letras de MAIÚSCULAS e minúsculas, sem acentos e espaços. (case-sensitive)

```
var meuNOME = "Karina";  
var meunome = "Karina";  
var meuNome = "Karina";  
var ø_ø = "JavaScript é legal!"; // Não recomendado
```

- **numeros:** desde que sejam precedidos por uma ou mais letras.

```
var camisa9 = "Pipico";  
var camisa10atacante = "Neymar";
```

- **underline e cifrão:** "_" e "\$" também são permitidos em qualquer posição, mas pouco usados.

```
var _nome = "Willany"  
var segundo_nome = "Silva"  
var $ultimo_nome_ = "Melo"
```

variáveis

```
var nome = prompt('Digite seu nome: ');  
  
alert(nome + ', seja bem vinda');
```

Agora é sua vez

Cálculo de idade:

- Declara o ano atual para fazer o cálculo
- Pede que o usuário digite o ano em que nasceu
- Calcula a idade do usuário e armazena na variável idade
- Mostra ao usuário a idade que ele possui

variáveis

```
let name = 'Simone';
```

Disponível no escopo do bloco onde foi declarada

```
const name = 'Simone';
```

Não pode ser reassinada para um novo valor

Array



cômoda = array
gaveta = item do array
conteúdo = valor do array

Array

```
let names = ["Simone", "Will", "Karina", "Gabi", "Milena"];
```

0 1 2 3 4

```
let names = [{ name: "Simone", age: 38 }, "Single", 2019];
```

Array

`names.push("Maria");` ➡ Add um novo valor no fim do array

`names.pop();` ➡ remove um valor do fim do array

`names.unshift("Karla");` ➡ Add um novo valor no início do array

`names.shift();` ➡ remove um valor do início do array

Array

`names.length` → Tamanho do array

`names.reverse()` → Retorna os valores do array em ordem reversa

`array1.concat(array2);` → Usado para unir dois ou mais arrays

`names.slice(início, fim);` → Retorna um novo array com os valores indicado do início e fim do array original

`names.splice(início, nº de itens delatados, itens adicionados);`

Agora é sua vez

- Criar um array com 5 elementos
- Adicionar itens no final e no início do array
- Remover itens no final e no início do array
- Qual o tamanho do seu array?
- Mudar a ordem do array
- Criar um novo array
- Criar um terceiro array com os valores dos dois arrays que você criou inicialmente
- Remover elementos do array usando `slice()` e `splice()`

Objeto

- É uma coleção de propriedades
- Uma propriedade é uma associação entre um nome (ou chave) e um valor.
- Um valor de propriedade pode ser uma função, número, string ou até outro objeto.

Objeto

```
let person = {  
  name: "Beyonce",  
  age: 37,  
  born: "September/04/81",  
  kids: ["Blue Ivy Carter", "Rumi Carter", "Sir Carter"],  
}
```

Objeto

```
person.songs = ["Halo", "Single Ladies"];
```

```
person.age = 35;
```

```
console.log(person[name]);
```

Agora é sua vez

- Criar um Objeto para guardar os dados de um projeto:
 - O nome é uma string
 - O time é um array de nome
 - A data de entrega é uma string
- Adicione mais uma pessoa no time
- Exclua a uma pessoa do time
- Altere o nome do projeto

Tempo



Estrutura de decisão IF/ELSE

São comandos da linguagem que permitem desviar o fluxo do programa, dependendo de uma condição.

Estrutura de decisão IF/ELSE

```
if (<condição>) {  
    <código a ser executado caso a condição  
    seja verdadeira>  
}
```

Estrutura de decisão IF/ELSE

```
const numero = 11;  
if (numero >= 7 || numero <10) {  
  console.log("o número é maior que 10");  
}
```


Estrutura de decisão IF/ELSE

```
const numero = 11;
if (numero > 10) {
  console.log("o número é maior que 10");
}else if(numero <10 && numero > 0){
  console.log("o numero é menor que 10 e maior que
0");
}else {
  console.log("o número é menor que 10");
}
```

Agora é sua vez

Média do aluno:

- Criar duas constantes com valores que representam notas
- calcular a média e armazenar em uma constante
- se a média for maior ou igual a sete, mostrar “Aprovado”
- senão mostrar “Reprovado”

*** bônus

- se a média for igual a dez, mostrar “Aprovado, top da balada”

Estrutura de repetição FOR

Formado por três partes:
inicialização, condição e
incremento.

Estrutura de repetição FOR

```
for (contador; condicao; incremento++) {  
    acao  
}
```

Estrutura de repetição FOR

```
for (let contador=0; contador<100; contador++) {  
    console.log(contador);  
}
```

Estrutura de repetição FOR

```
let names = ["Simone", "Will", "Karina", "Gabi", "Milena"];

for (let contador=0; contador<names.length; contador++) {
  console.log(names[contador]);
}
```

Agora é sua vez

Número par:

- Criar uma instrução de repetição de vai de 1 até 100
- verifique se o número atual é par
- se for par, mostre o número

Use o operador Modulus (%) para verificar o resto da divisão

MAP

- Mapeia os elementos de um array para um novo array
- Repassando o resultado de uma função aplicado a cada elemento do array.
- Aplicado apenas para arrays

Map

```
const numbers = [1, 4, 9, 16];  
const add = numbers.map(number => number + 1);  
const multiply = numbers.map(number => number * 2);
```

Links úteis

[Rocketseat - curso gratuito](#)

[The 10 Best JavaScript ES6 Features](#)

[JavaScript Variables: The Beginner's Guide](#)

[Declaring a Winner 🏆 Between JavaScript's var, let and const](#)

[JavaScript brief history and ECMAScript\(ES6,ES7,ES8,ES9\) features](#)

[What's new in ECMAScript 2019 \(ES2019\) / ES10](#)

[Simplify your JavaScript – Use .map\(\), .reduce\(\), and .filter\(\)](#)