



HELLO
WORLD



ITERADÖES

COM ARRAYS

O QUE É?

O objeto Array do JavaScript é um objeto global usado na construção de 'arrays': objetos de alto nível semelhantes a listas.

```
let array = [1, 12, 2.5, null, 'John', true, 100]
```

	int	int	float	Null	string	bool	number
Elements: →	1	12	2.5	null	'John'	true	100
Index : → (position)	0	1	2	3	4	5	6

Javascript Array

MÉTODO MAP

Utilizamos essa função para percorrermos um vetor e retornar outro novo vetor com as alterações necessárias. Nesse exemplo percorremos o vetor `arr` multiplicando seus valores por dois, assim gerando um novo vetor chamado `newArr` com os valores multiplicados.

```
const arr = [1,2,3,4,5,6,7,8,9,10]
const newArr = arr.map(item => item * 2)
console.log(newArr) //[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
```

MÉTODO FILTER

Utilizamos essa função para filtrar o conteúdo de um vetor. Nesse exemplo vamos filtrar todos os valores pares do vetor arr, assim gerando um novo vetor chamado newArr com os valores positivos filtrados.

```
const arr = [1,2,3,4,5,6,7,8,9,10]
const newArr = arr.filter(item => item % 2 === 0)
console.log(newArr) //[2, 4, 6, 8, 10]
```

MÉTODO REDUCE()

Utilizamos essa função para reduzir uma vetor em uma única informação, geralmente um número, a função recebe uma variável acumuladora e o item da interação. Nesse exemplo vou somar os itens de um vetor.

```
const arr = [1,2,3,4,5,6,7,8,9,10]  
const sum = arr.reduce((accumulator, item) => accumulator + item)  
console.log(sum) //55
```

MÉTODO FIND

Utilizamos essa função para encontrar um item em um vetor. Nesse exemplo vou buscar um número em um vetor. Caso o item procurado não fosse encontrado o retorno seria undefined.

```
const arr = [1,2,3,4,5,6,7,8,9,10]
const found = arr.find((item) => item === 2)
console.log(found) //2
```

MÉTODO FINDINDEX

O método `findIndex` retorna o índice do primeiro elemento no array que satisfaz a função de teste fornecida. Caso contrário, `-1` é retornado.

```
const arr = [1, 2, 3, 4, 5];  
const foundIndex = arr.findIndex(item => item > 3);  
console.log(foundIndex); // 3
```


MÉTODO SOME

Utilizamos essa função para encontrar um item em um vetor, diferentemente da função find que retorna o item encontrado a função some retorna um resultado booleano, true se encontrou e false se não encontrou.

```
const arr = [1,2,3,4,5,6,7,8,9,10]
const found = arr.some((item) => item === 2)
console.log(found) //true
```

MÉTODO FOREACH

O método `forEach` executa uma função para cada elemento do array.

```
const arr = [1, 2, 3, 4, 5];  
arr.forEach(item => {  
    console.log(item);  
});
```

MÉTODO EVERY

O método every testa se todos os elementos do array passam pelo teste implementado pela função fornecida. Retorna um booleano.

```
const arr = [1, 2, 3, 4, 5];  
const allEven = arr.every(item => item % 2 === 0);  
console.log(allEven); // false
```

MÉTODO FOR.. LOOP

O for loop é uma maneira tradicional de iterar sobre um array. Ele permite controlar explicitamente o índice e a condição de parada.

```
const arr = [1, 2, 3, 4, 5];  
for (let i = 0; i < arr.length; i++) {  
  console.log(arr[i]);  
}
```

MÉTODO FOR...OF LOOP

O for...of loop é uma forma mais simples e moderna de iterar sobre os valores de um array.

```
const arr = [1, 2, 3, 4, 5];  
for (const item of arr) {  
  console.log(item);  
}
```

The background is a dark blue night sky. In the center is a large, pixelated yellow sun or moon. Below it is a pixelated burger with a yellow bun, an orange cheese slice, a pink and white checkered patty, and a brown ketchup slice. On either side of the burger are two pixelated clouds. In the foreground, there are two tall, pixelated buildings with vertical green and blue stripes and orange windows. At the bottom of each building is a blue bus with a red roof and black wheels.

DESAFIO

O objetivo deste exercício é praticar a iteração e manipulação de arrays em JavaScript. Você aprenderá a calcular a soma e a média dos elementos de um array, além de filtrar elementos com base em uma condição. Ao final, você retornará um objeto que contém várias informações derivadas do array original.



GAME RULES

01

Utilize o método reduce para calcular a soma de todos os elementos do array.

02

Com a soma obtida, divida-a pelo número de elementos do array para obter a média.

03

Com a soma obtida, divida-a pelo número de elementos do array para obter a média.



MEUS CONTATOS:



27 99500-7495



<https://beacons.ai/prismatech>



producaoprismatech@gmail.com



Avenida Jerônimo Monteiro 145, Vitória





THANK
YOU