

[Array]

In JavaScript

- O que é um array?
- Para que serve um array?
- Propriedade length
- Adicionando elementos em um array
- Acessando e interagindo nos elementos do Arrays
- Métodos do Javascript

O que é um array?

Os arrays são estruturas que servem para guardar dados, e organizá-los. Seu objetivo é ser um espaço fixo na memória do computador que armazena elementos. Esses elementos podem ser acessados por um tipo de indicação, que chamamos de índice.

A finalidade de um array é armazenar vários dados do mesmo tipo juntos.

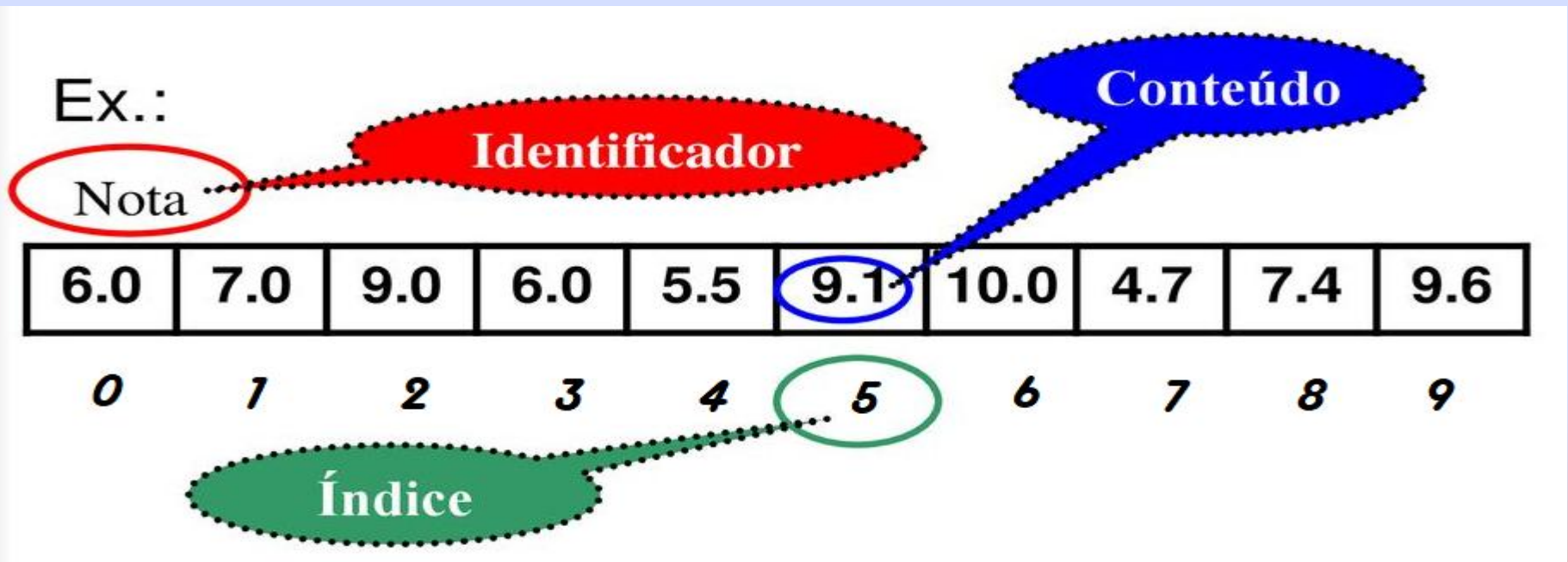


Para que serve um array?

Um array é uma estrutura de dados que serve para guardar elementos em um espaço da memória. Estes espaços da memória são chamados de variáveis.

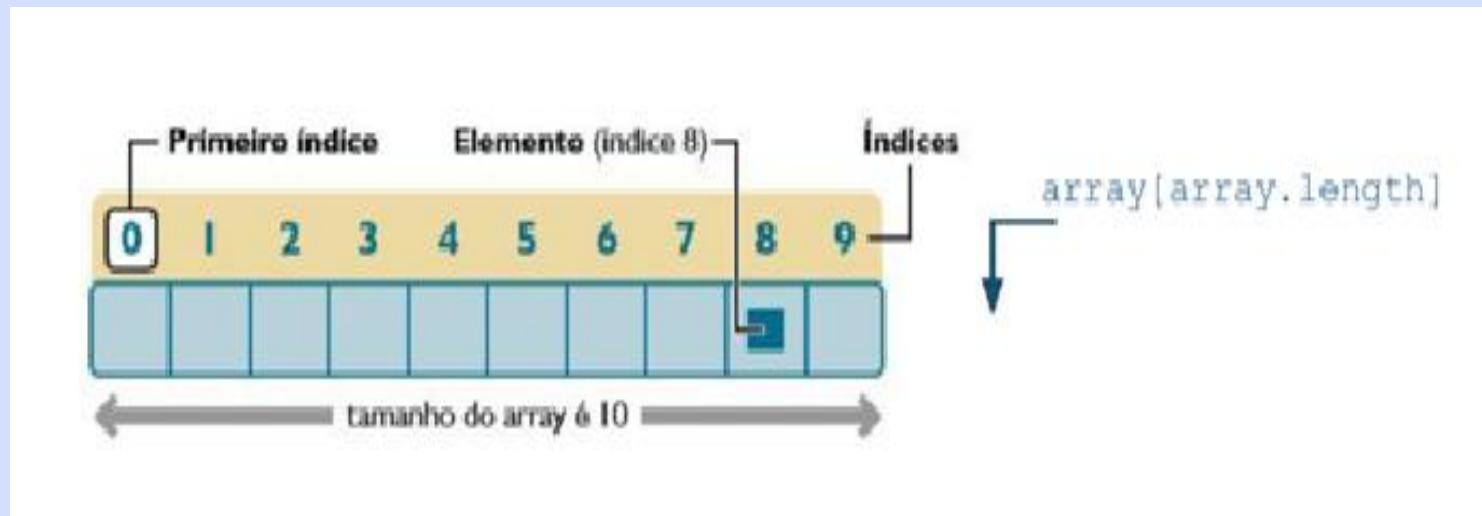
Arrays são usados para armazenar vários valores em uma única variável. Isso é comparado a uma variável que pode armazenar apenas um valor.

Cada item em um array tem um número anexado a ele, chamado de índice numérico, que permite acessá-lo.



Propriedade length

length (tamanho, em português) é uma propriedade dos arrays em JavaScript que retorna ou define o número de elementos em um determinado array.



Adicionando elementos em um array

O array é um dos tipos de dados mais utilizados quando estamos trabalhando com uma lista ordenada de valores.

Cada valor é referido como um elemento com um índice único. Ele armazena elementos de vários tipos de dados, que você pode acessar por meio de uma única variável.

Forma padrão para percorrer e adicionar elementos em um array usamos o for.

```
var dados = [3,5,6,8,11,3, 12,16];

for (var i = 0; i < dados.length; i++) {

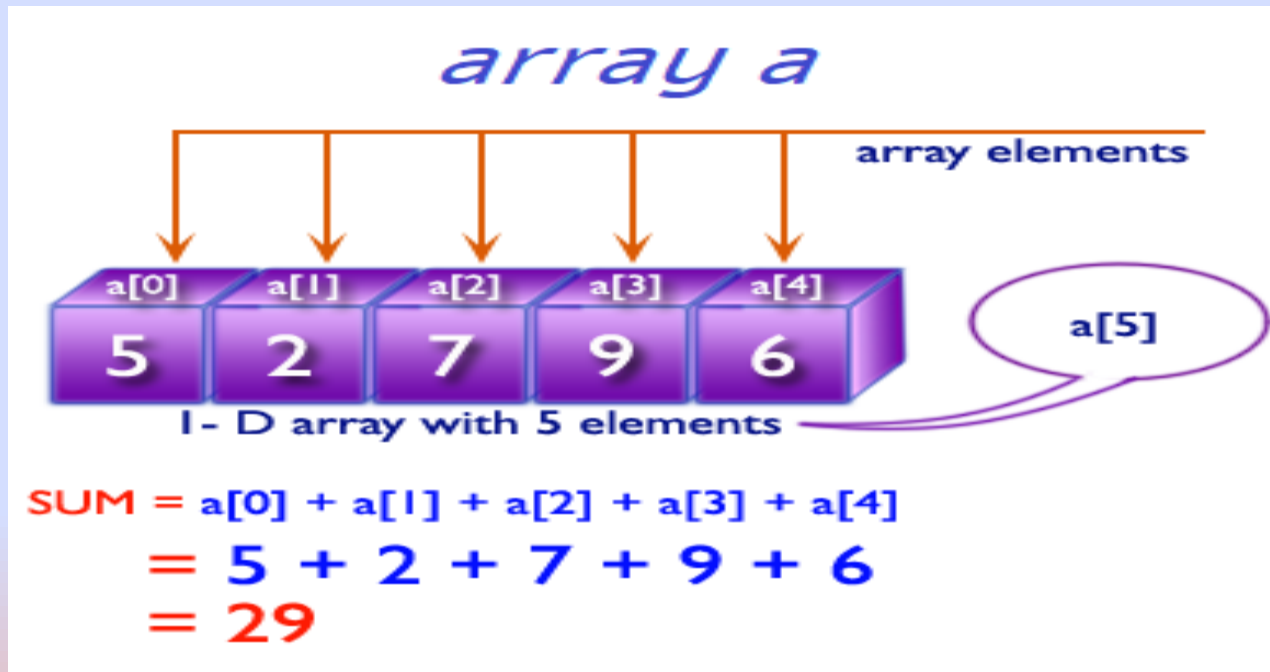
    console.log(dados[i]);

}
```

Acessando e interagindo nos elementos do Arrays

Para acessar um elemento do array usamos o nome da variável e indicando o índice da posição a ser acessada entre colchetes.

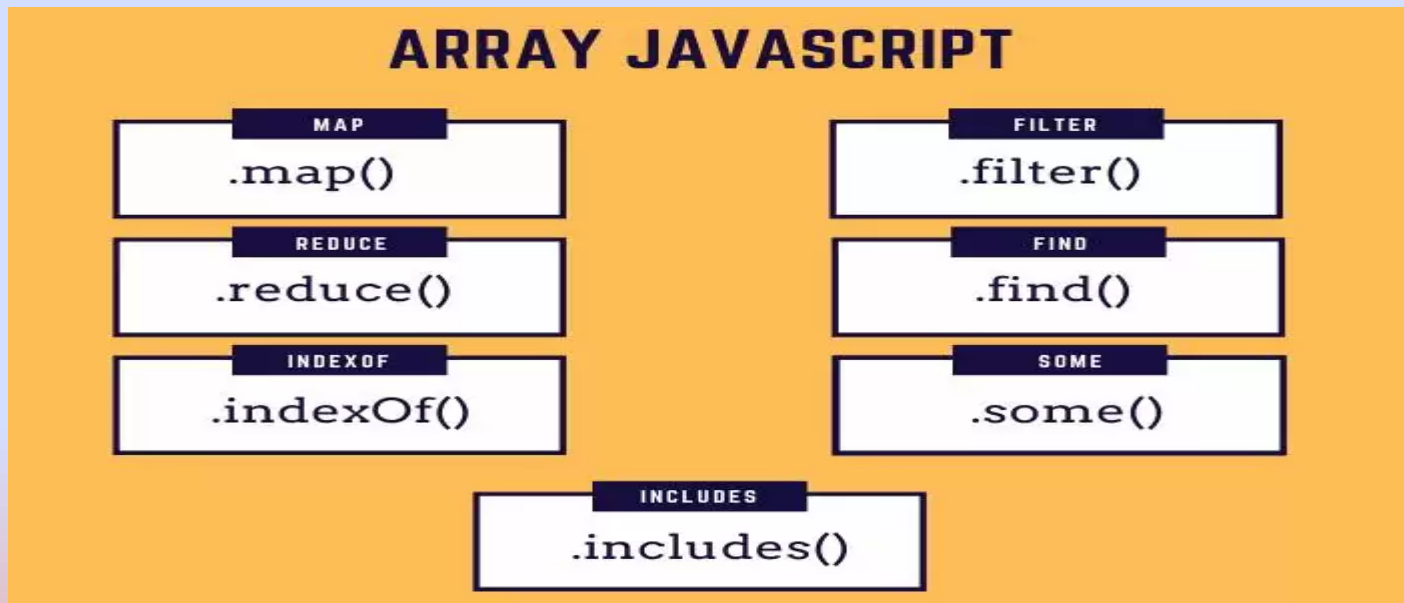
Além disso, sabemos que em JavaScript os índices dos arrays começam por 0(zero).



Métodos do Javascript

Em JavaScript praticamente qualquer tipo de dado é um objeto. Cada item dessa "coleção de valores", é chamado de propriedade. Cada propriedade é composta por um par de "nome: valor". Quando uma propriedade armazena uma função, ela se torna o que chamamos de método.

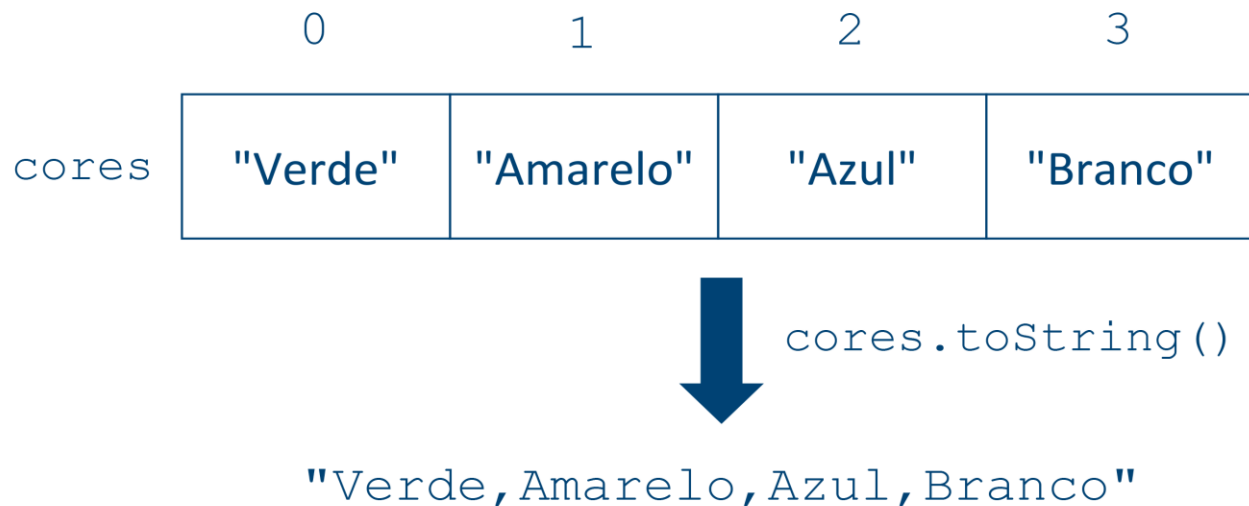
Existem alguns métodos oferecidos por JavaScript que são bastante úteis para trabalhar com arrays.



Alguns Métodos do Javascript

Método toString

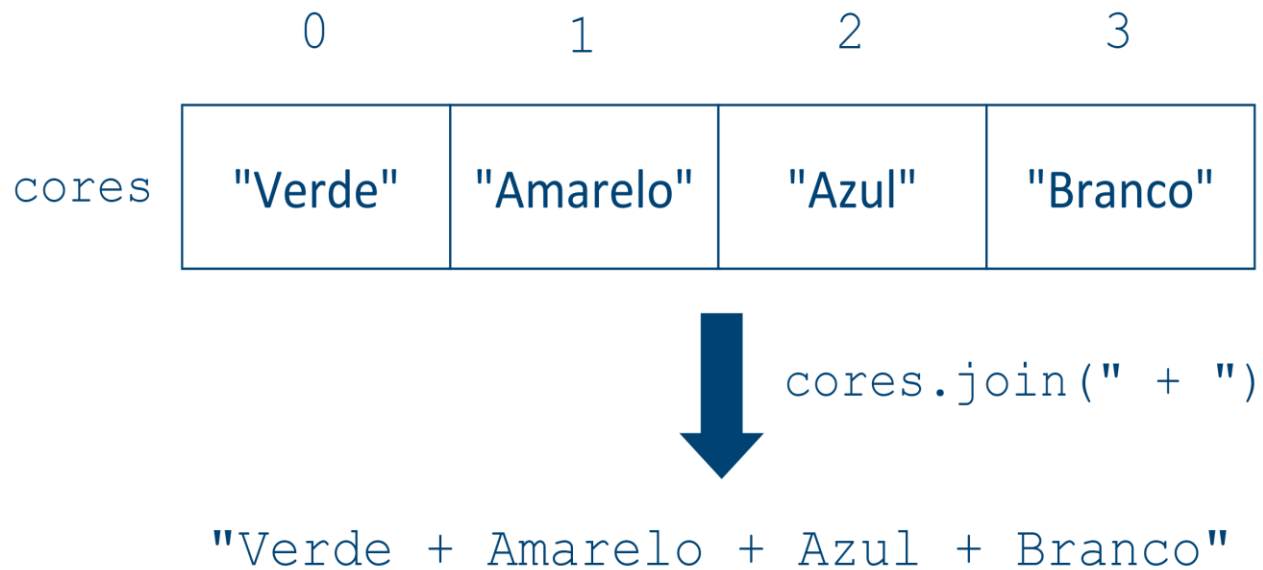
Retorna uma string contendo os elementos do array separados por vírgula.



Alguns Métodos do Javascript

Método join

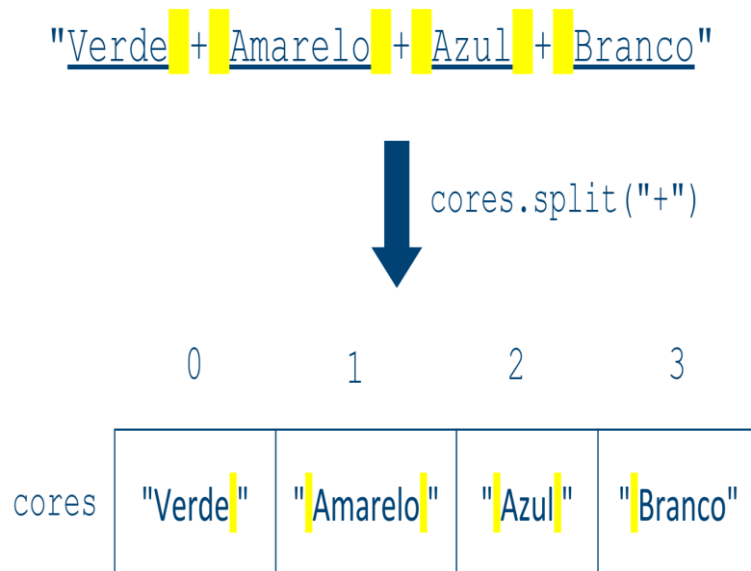
Tem o mesmo comportamento do método toString, mas utiliza o separador passado por você para separar os elementos do arrays na string resultante.



Alguns Métodos do Javascript

Método split

Recebe uma string que indica quem é o separador dos elementos e retorna um array contendo os elementos que estão separados pelo separador indicado pelo programador.



```
let string = "split";

string = '🍗🍰🐟';
string.split(' ') // [ "🍗", "🍰", "🐟" ]

string = '🍗🍕🍷🐟';
string.split(',') // [ "🍗", "🍕", "🍷", "🐟" ]

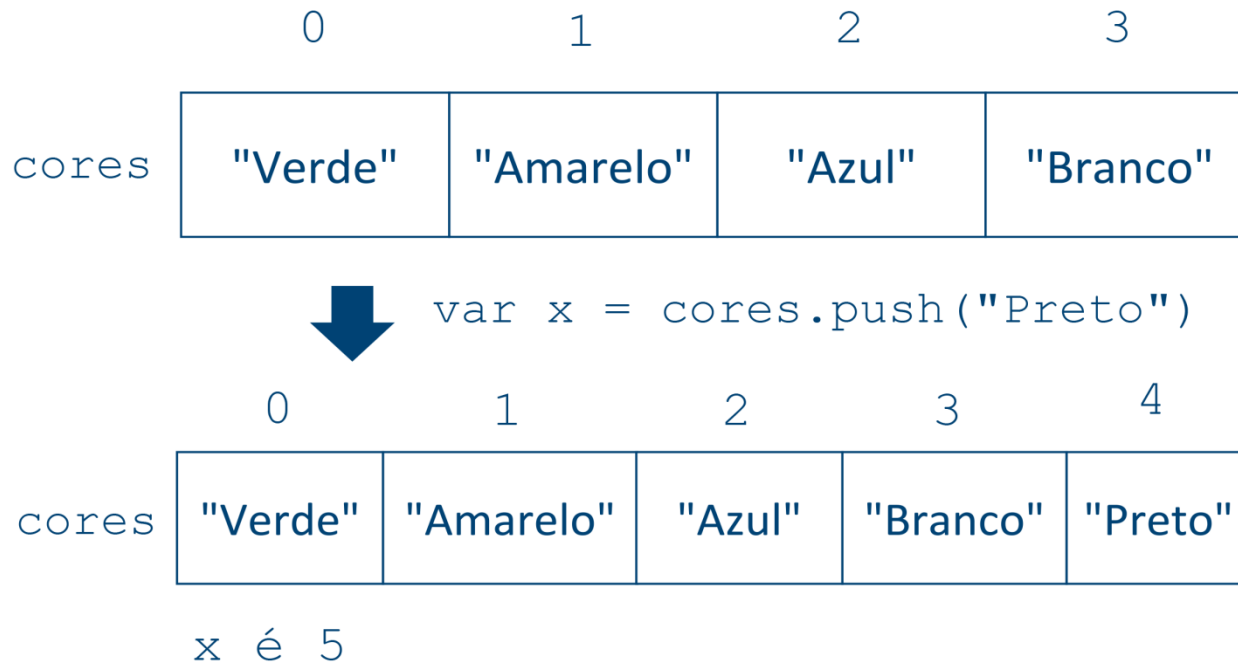
string = '🍰-🍷-🍕';
string.split('-') // [ "🍰", "🍷", "🍕" ]

result => ["s", "p", "l", "i", "t"]
```

Alguns Métodos do Javascript

Método push

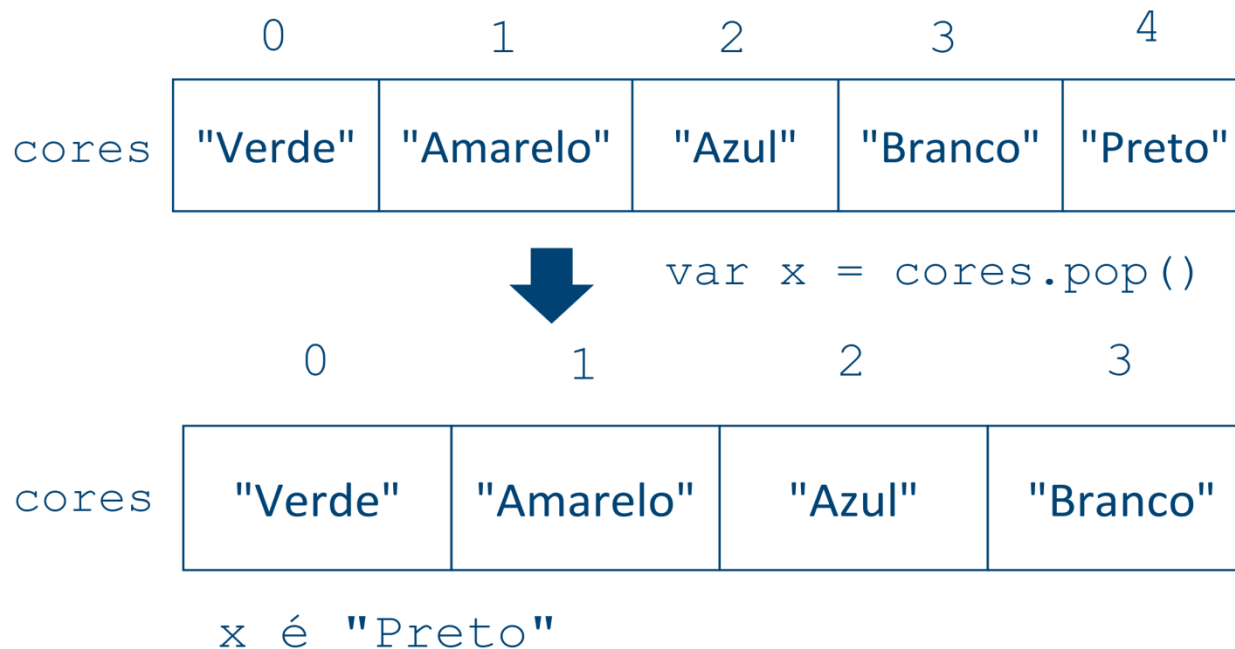
Adiciona o elemento passado no final do array e retorna o tamanho final do array.



Alguns Métodos do Javascript

Método pop

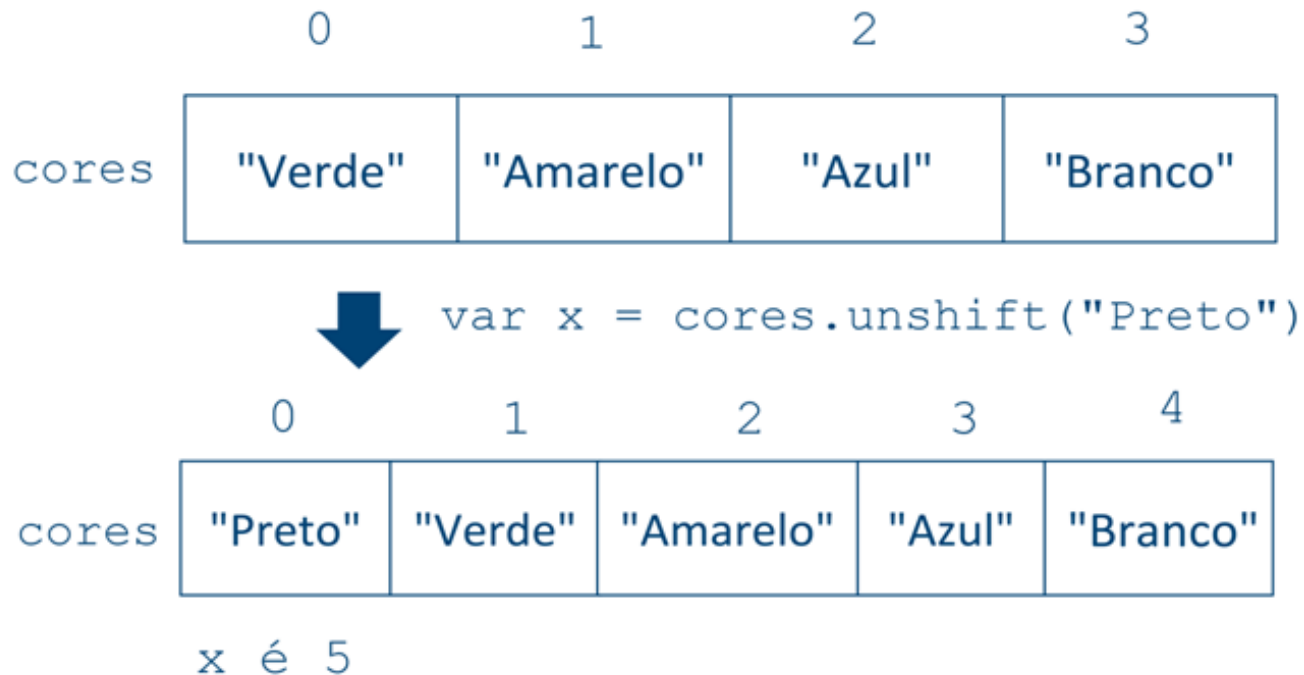
Remove o último elemento do array e retorna esse elemento.



Alguns Métodos do Javascript

Método unshift

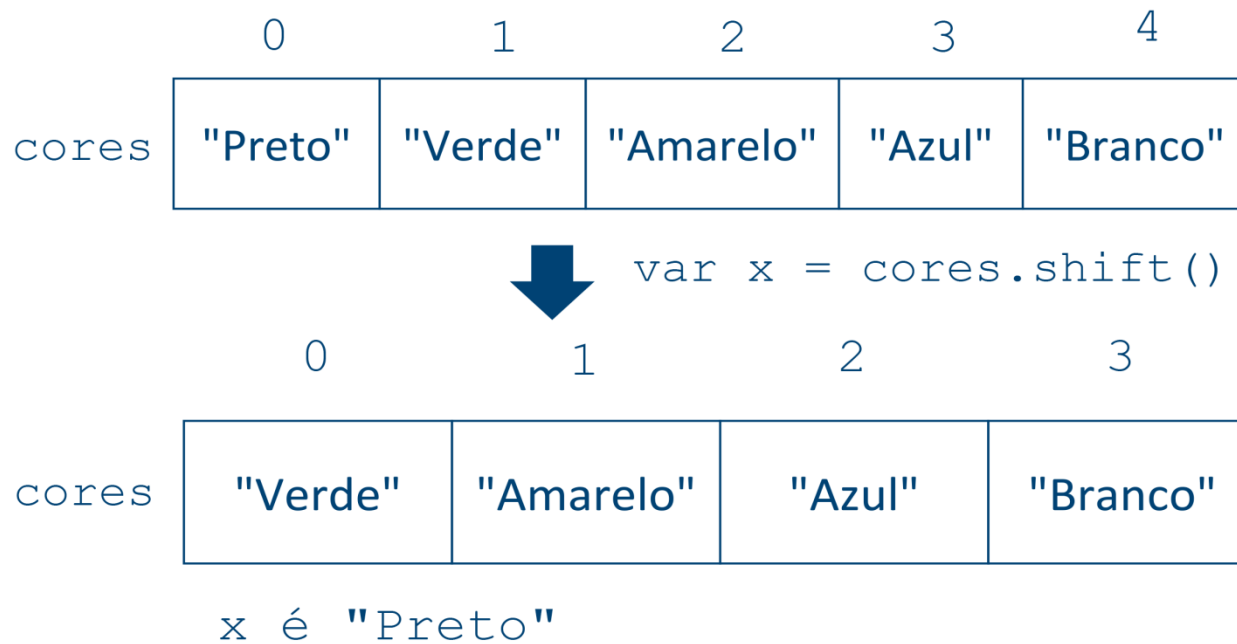
Adiciona o elemento passado no início do array e retorna o tamanho final do array.



Alguns Métodos do Javascript

Método shift

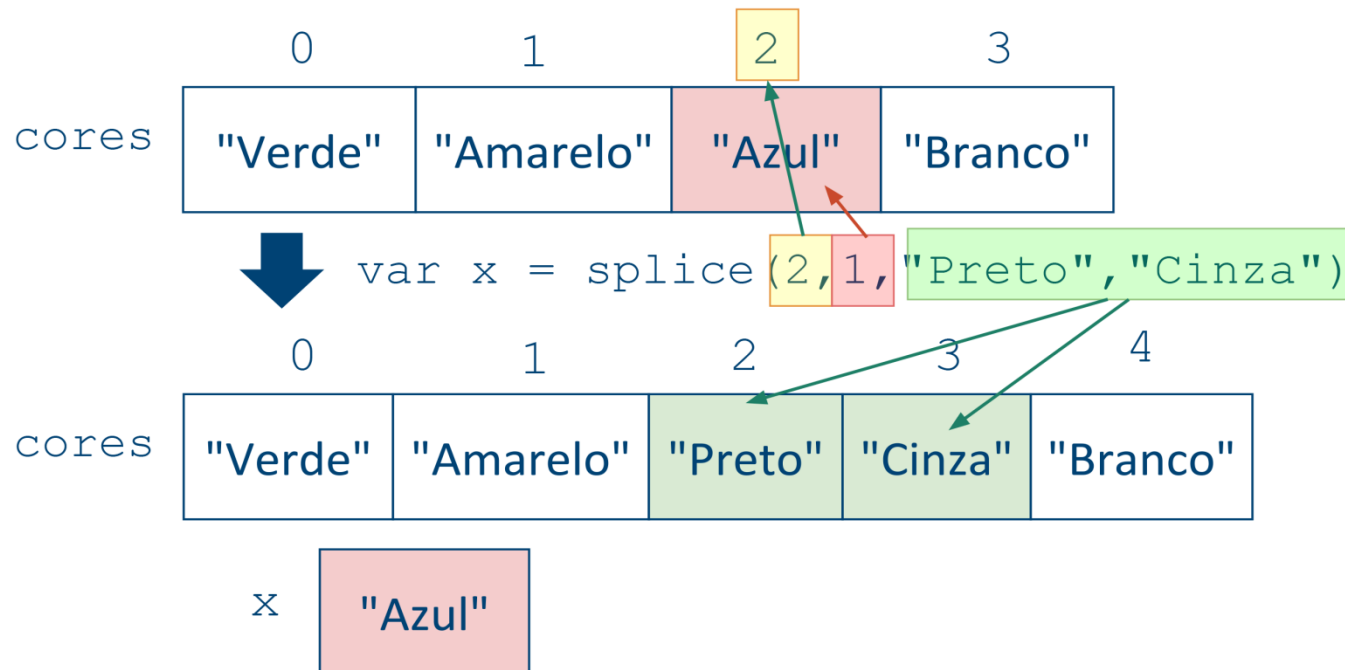
Remove o primeiro elemento do array e o retorna.



Alguns Métodos do Javascript

Método splice

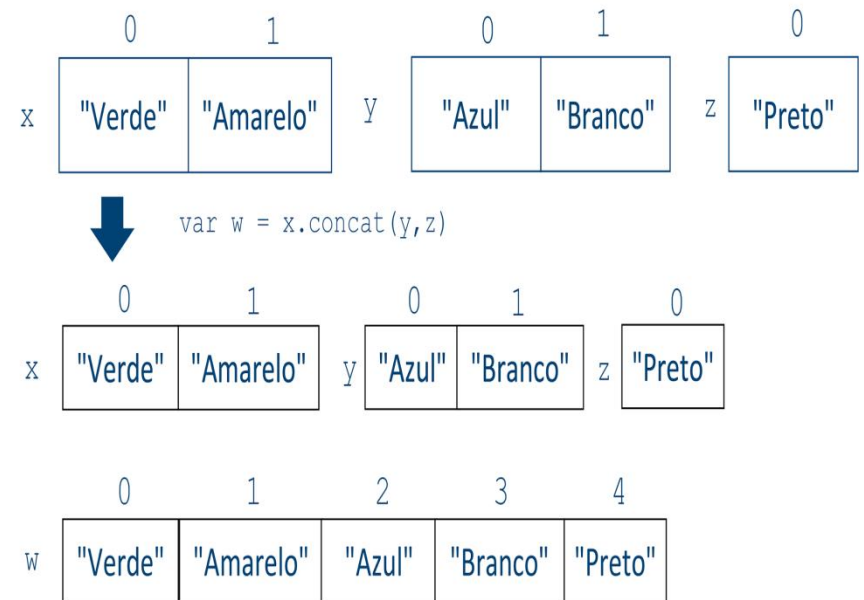
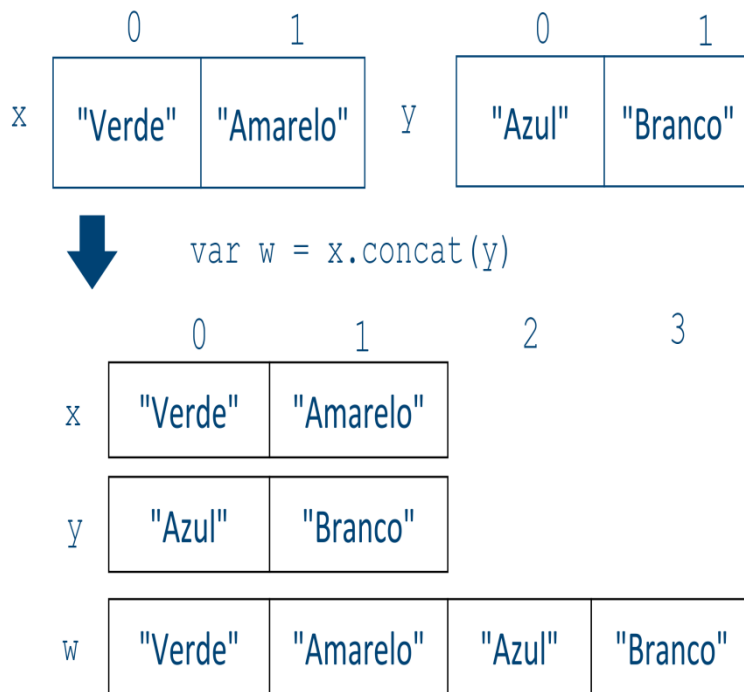
Também pode ser usado para adicionar e remover elementos do array. A sua principal diferença é que ele permite que isso seja feito dentro dos arrays e não apenas no início e no final.



Alguns Métodos do Javascript

Método concat

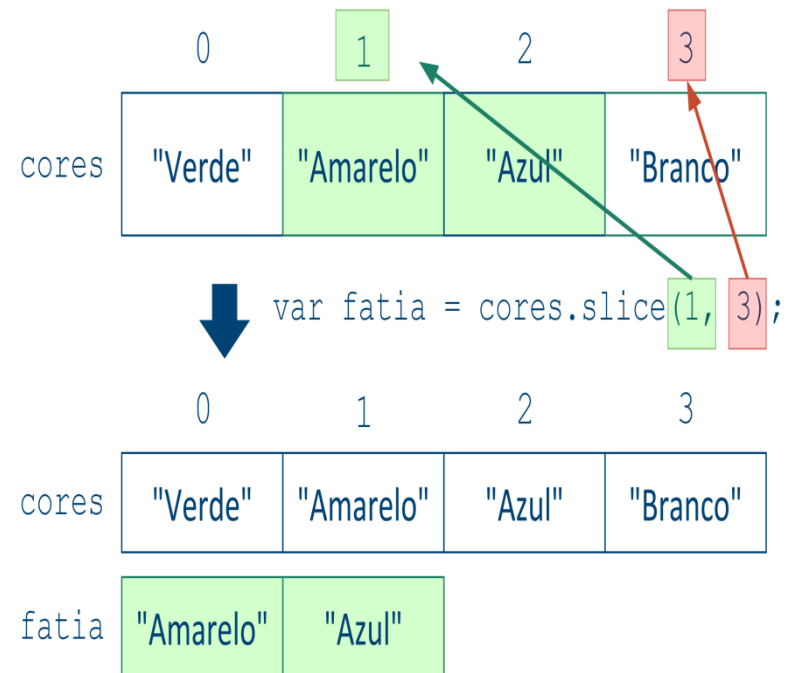
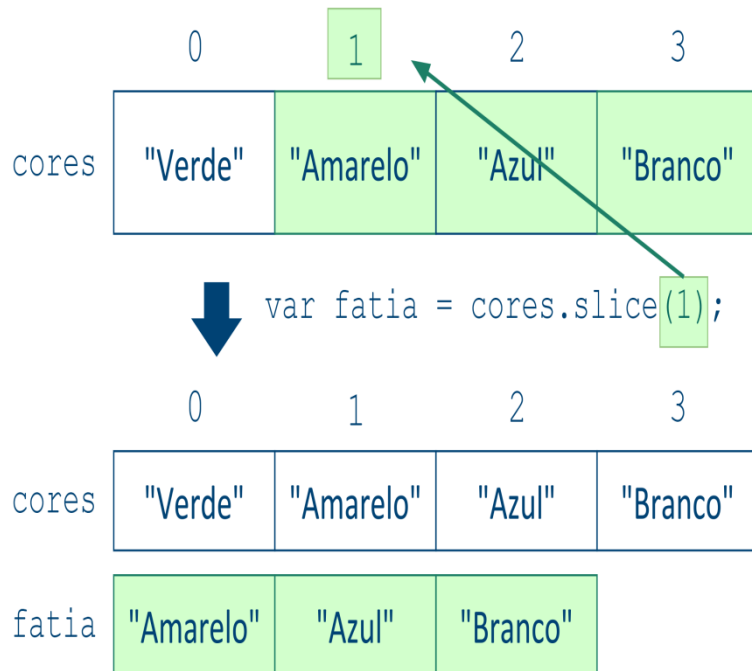
Concatena os arrays passados como parâmetros. Em outras palavras, esse método une os dois arrays. Não altera o array original.



Alguns Métodos do Javascript

Método slice

Retorna uma "fatia" (slice em inglês) do array à qual ele é aplicado. Assim como o método concat, ele não altera o array original, mas apenas retorna um novo array com o resultado da operação.



map([🐮, 🍠, 🐔, 🌽], cook)

=> [🍔, 🍟, 🍗, 🍿]

filter([🍔, 🍟, 🍗, 🍿], isVegetarian)

=> [🍟, 🍿]

reduce([🍔, 🍟, 🍗, 🍿], eat)










=> 💩






   .map(=>) =>    

   .filter() => 

   .every() => false

   .some() => true

   .fill(, 1) =>    

   .findIndex(el => el===) => 2

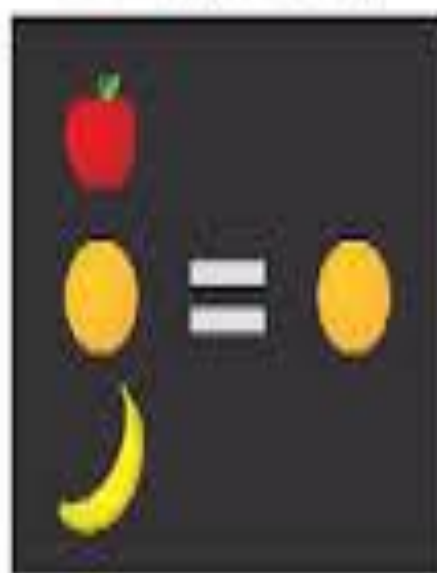
   .find() => 

   .reduce((acc, cur)=>acc+cur)=> 

`Array.map()`



`Array.filter()`



`Array.reduce()`

