



Métodos de Arrays



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



No **Javascript**, o **array** é muito importante.

Por isso, temos uma série de **métodos muito úteis** para trabalhar com eles.





.push()

Adiciona um ou mais **elementos** ao **final** do array

- **Recebe** um ou mais elementos como parâmetros
- **Retorna** o novo comprimento do array





{ }

```
let cores = ["Roxo", "Laranja", "Azul"]

cores.push("Violeta") // Retorna 4

console.log(cores)
// ["Roxo", "Laranja", "Azul", "Violeta"]

cores.push("Cinza", "Ouro") // Retorna 6

console.log(cores)
// ["Roxo", "Laranja", "Azul", "Violeta", "Cinza", "Ouro"]
```



.pop()

Elimina o último elemento de um array

- **Não recebe** parâmetro
- **Retorna** o elemento eliminado





```
let cores = ["Roxo", "Laranja", "Azul"]
```

```
let ultimaCor = cores.pop()
```

```
console.log(cores) // ["Roxo", "Laranja"]
```

```
console.log(ultimaCor) // Azul
```

```
{}
```



.shift()

Elimina o **primeiro** elemento de um array

- **Não recebe** parâmetro
- **Retorna** o elemento eliminado





```
let cores = ["Roxo", "Laranja", "Azul"]
```

```
let primeiraCor = cores.shift()
```

```
console.log(cores) // ["Laranja", "Azul"]
```

```
console.log(primeiraCor) // Roxo
```

```
{}
```




.unshift()

Adiciona um ou mais **elementos** ao **início** de um array

- **Recebe** um ou mais elementos como parâmetro
- **Retorna** o novo comprimento do array





```
let cores = ['Roxo', 'Laranja', 'Azul'];  
cores.unshift('Violeta'); // Retorna 4
```

```
console.log(cores);  
// ["Violeta", "Roxo", "Laranja", "Azul"]
```

```
{}
```

```
cores.unshift('Cinza', 'Ouro') // Retorna 6
```

```
console.log(cores);  
// ["Cinza", "Ouro", "Violeta", "Roxo", "Laranja", "Azul"]
```



.join()

Junta os elementos de um array usando um separador que especificamos. Se não o especificar, use vírgulas.

- **Recebe** um separador (string), opcional
- **Retorna** uma string com os elementos unidos





{}

```
let cores = ['Roxo', 'Laranja', 'Azul'];

let separadosPorVirgula = cores.join()

console.log(separadosPorVirgula)
// "Roxo,Laranja,Azul"

let separdosPorTraco = cores.join(" - ")

console.log(separdosPorTraco)
// "Roxo - Laranja - Azul"
```



.lastIndexOf()

Similar ao **.indexOf()**, exceto que ele começa procurando pelo elemento no **final do array** (de trás para frente).

Se houver elementos repetidos, ele retorna a posição do primeiro que encontrar (ou seja, o último, se olharmos desde o início).





```
let cores = ['Roxo', 'Laranja', 'Azul', 'Laranja'];
```

```
cores.lastIndexOf("Laranja")
```

```
// Ele encontrou o que procurava.
```

```
// Retorna 3, que é o índice do elemento encontrado.
```

```
cores.indexOf("Abacaxi")
```

```
// Não encontrou o que procurava. Retorno -1.
```

```
{}
```



.includes()

Também similar ao **.indexOf()**, mas retorna um booleano

- **Recebe** um item para pesquisar no array
- **Retorna** verdadeiro se encontrou o que estava procurando, falso se não





```
let cores = ['Roxo', 'Laranja', 'Azul'];
```

```
cores.includes("Laranja")
```

// Ele encontrou o que procurava. Retorna true

```
{}
```

```
cores.includes("Abacaxi")
```

// Não encontrou o que procurava. Retorna false

DigitalHouse>
Coding School