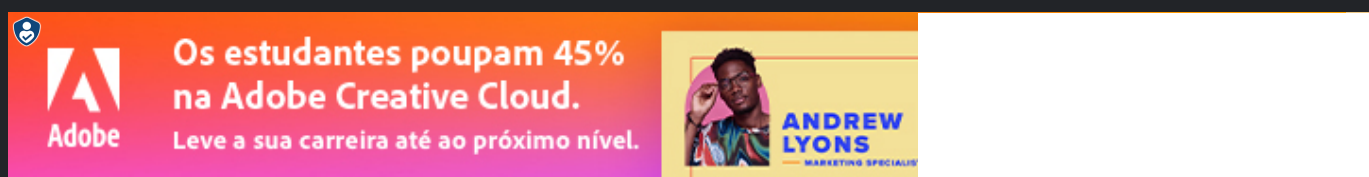


16 Métodos de array JavaScript que todo bom Desenvolvedor Web deve saber

Fundamental para evitar de reinventar soluções.

Sep 14, 2021



[JavaScript](#) fornece vários métodos integrados para lidar com matrizes/arrays. Vamos ver uma lista de **16** desses métodos que o ajudarão a melhorar suas habilidades de desenvolvimento de JavaScript.

1. `includes`

O método `includes` verifica na matriz e retorna `true` se o elemento fornecido está presente e `false` se não está.

```
const numbers = [1, 2, 3, 4, 5];
console.log(numbers.includes(3)); // true
console.log(numbers.includes(3, 3)); // false
```

O método `includes` também usa um segundo parâmetro que é a posição inicial para iniciar a verificação.

2. `map`

O método `map` pega um elemento individual da matriz fornecida e altera seu valor de acordo com a função passada, então retorna uma nova matriz com esses valores modificados.

```
const numbers = [1, 2, 3, 4, 5];
const squared = numbers.map((e) => e ** 2);
console.log(squared); // [ 1, 4, 9, 16, 25 ]
```

3. filter

O método **filter** verifica cada elemento na matriz em relação à condição na função passada e retorna apenas os elementos que atendem a essa condição.



```
const numbers = [1, 2, 3, 4, 5];
const result = numbers.filter((element) => element % 2 === 0);
console.log(result); // [ 2, 4 ]
```

4. forEach

O método **forEach** executa uma função uma vez para cada elemento da matriz.

```
const numbers = [1, 2, 3, 4, 5];
numbers.forEach((element) => {
  const message =
    element % 2 === 0
      ? `${element} é um número par`
      : `${element} é um número ímpar`;
  console.log(message);
});
// Saída:
// 1 é um número ímpar
// 2 é um número par
// 3 é um número ímpar
// 4 é um número par
// 5 é um número ímpar
```

5. reduce

O método **reduce** usa uma função com dois parâmetros, o primeiro parâmetro é para acumular o resultado e o segundo parâmetro para manter o valor do elemento atual durante a iteração. Esse valor acumulado é retornado após a execução das operações na função fornecida.

```
const numbers = [1, 2, 3, 4, 5];
const result = numbers.reduce(
  (totalSum, currentValue) => totalSum + currentValue
);
console.log(result); // 15
```


10. `indexOf`

O método `indexOf` retorna o índice do elemento fornecido na matriz se for encontrado e retorna `-1` se não for encontrado.

```
const numbers = [1, 2, 3, 4, 5];
const result = numbers.indexOf(3);
console.log(result); // 2
```

11. `push`

O método `push` adiciona novos elementos no final da matriz fornecida.

```
const numbers = [1, 2, 3, 4, 5];
numbers.push(6);
console.log(numbers); // [ 1, 2, 3, 4, 5, 6 ]
```

12. `pop`

O método `pop` remove o último elemento do final da matriz fornecida.

```
const numbers = [1, 2, 3, 4, 5];
numbers.pop();
console.log(numbers); // [ 1, 2, 3, 4 ]
```

13. `shift`

O método `shift` remove um elemento do início da matriz fornecida.

```
const numbers = [1, 2, 3, 4, 5];
numbers.shift();
console.log(numbers); // [ 2, 3, 4, 5 ]
```

14. `unshift`

O método `unshift` adiciona a lista de elementos fornecida ao array existente.

```
const numbers = [2, 3, 4, 5];
numbers.unshift(0, 1);
console.log(numbers); // [ 0, 1, 2, 3, 4, 5 ]
```

15. `splice`

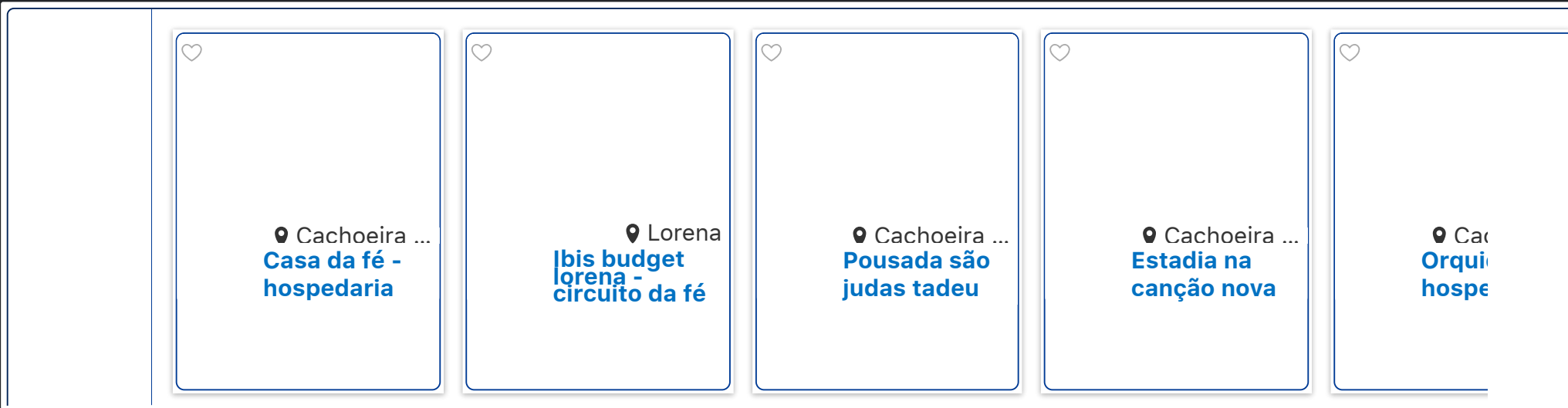
O método `splice` remove o número fornecido de elementos de um determinado índice e adiciona uma lista de elementos a partir do índice fornecido. Aqui, no exemplo, o método de splicing remove três elementos a partir do índice de `0` e adiciona o `"0"`, `"1"`, `"2"`, `"3"` elementos a partir do índice de `0`.

```
const numbers = [1, 2, 3, 4, 5];
numbers.splice(0, 3, "0", "1", "2", "3");
console.log(numbers); // [ '0', '1', '2', '3', 4, 5 ]
```

16. slice

O método **slice** cria e retorna uma nova matriz a partir da matriz fornecida contendo os elementos do índice inicial (incluído) ao índice final. Se nenhum índice final for fornecido, todos os elementos do índice inicial fornecido até o final da matriz serão retornados.

```
const numbers = [1, 2, 3, 4, 5];
console.log(numbers.slice(0, 2)); // [ 1, 2 ]
```



Esse artigo foi publicado originalmente por [plainenglish.io](#) em inglês .

javascript

desenvolvimentoweb

Compartilhe



Nosso canal no Youtube

Inscreva-se



Marcos Oliveira

Desenvolvedor de software

<https://github.com/terroo>

Artigos Relacionados



Lorena - Ibis Budget Lorena
- Circuito da Fé**

Anúncio Booking.com



24 Oct 2022

Como Obter o Diretório atual em 10
Linguagens de Programação diferentes



23 Sep 2022

10 Livros de JavaScript, TypeScript e
Node.js que vale à pena você ler



09 Sep 2022

Crow C++, um Framework Web: Fácil e
Veloz!

Crie Aplicativos Gráficos para Linux e Windows com C++

Aprenda C++ Moderno e crie Games, Programas CLI, GUI e TUI de forma fácil.

Saiba Mais

Receba as novidades no seu e-mail!

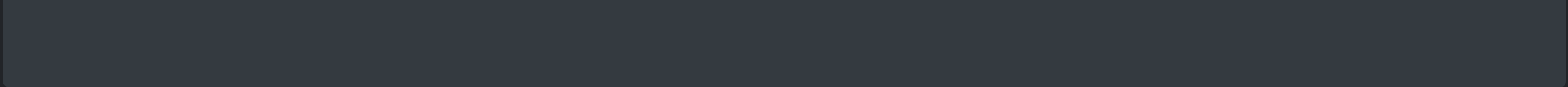
Após cadastro e confirmação do e-mail, enviaremos semanalmente resumos e também sempre que houver novidades por aqui para que você mantenha-se atualizado!

Digite seu e-mail

Quero receber novidades

caso queira entrar em contato conosco, [envie-nos um e-mail.](#)





Site sobre C++, Programação, Desenvolvimento de Games e Web.



Conheça também

[NewsTR](#) [English version](#)



[Terminal Root](#).[®] Todos os Direitos Reservados. Feito com [Jekyll](#), [Bootstrap](#) e ❤

