# Métodos de matriz de JavaScript

[❮ Anterior](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_arrays.asp&usg=ALkJrhi5m--e3K9kU3s86P0YhPIkOGvwFg)[Próximo ❯](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/js_array_sort.asp&usg=ALkJrhhbxaRDCRHQ7PqivHuSttJXzXnu-A)

## Convertendo Arrays para Strings

O método JavaScript **toString ()** converte uma matriz em uma seqüência de valores de matriz (separados por vírgulas).

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo").innerHTML = fruits.toString();

### Resultado

Banana,Orange,Apple,Mango

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_tostring&usg=ALkJrhhJ3bBjwvMEyqNGgNDx6J854TundQ)

O método **join ()** também junta todos os elementos da matriz em uma string.

Ele se comporta apenas como toString (), mas, além disso, você pode especificar o separador:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo").innerHTML = fruits.join(" \* ");

### Resultado

Banana \* Orange \* Apple \* Mango

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_join&usg=ALkJrhhqpcHn1QBzlD40Q51dd7vqh7zCXw)

## Popping and Pushing

Quando você trabalha com arrays, é fácil remover elementos e adicionar novos elementos.

Isto é o que aparecer e empurrar é:

Colocando itens **fora** de uma matriz ou empurrando itens **para** uma matriz.

## Popping

O método **pop ()** remove o último elemento de uma matriz:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.pop();              // Removes the last element ("Mango") from fruits

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_pop&usg=ALkJrhjLxJhg3f3un2ky7NglTI31JI1jbw)

O método pop () retorna o valor que foi "surgido":

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
var x = fruits.pop();      // the value of x is "Mango"

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_pop_out&usg=ALkJrhjtP4ZS8NkxxLMSKY15XQ61FwAvXA)

## Empurrando

O método **push ()** adiciona um novo elemento a uma matriz (no final):

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.push("Kiwi");       //  Adds a new element ("Kiwi") to fruits

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_push&usg=ALkJrhg-xs06AqulLe4dpOXP177adcKfHQ)

O método push () retorna o novo comprimento da matriz:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
var x = fruits.push("Kiwi");   //  the value of x is 5

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_push_length&usg=ALkJrhhlhbN2dVMtDWK98TS8z1WzxQ0Ndw)

## Elementos de Mudança

Shifting é equivalente a popping, trabalhando no primeiro elemento em vez do último.

O método **shift ()** remove o primeiro elemento de matriz e "desloca" todos os outros elementos para um índice mais baixo.

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.shift();            // Removes the first element "Banana" from fruits

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_shift&usg=ALkJrhivRlItw7PQUBjSrOi1wLHe0gYO6g)

O método shift () retorna a seqüência que foi "deslocada para fora":

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.shift();             // Returns "Banana"

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_shift_return&usg=ALkJrhg7SdTb5wObP4QVTenwG_QSlx4MLQ)

O método **unshift ()** adiciona um novo elemento a uma matriz (no início) e "destrói" os elementos mais antigos:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.unshift("Lemon");    // Adds a new element "Lemon" to fruits

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_unshift&usg=ALkJrhhndysFm84wNefoKgokmC98uNR9qg)

O método unshift () retorna o novo comprimento da matriz.

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.unshift("Lemon");    // Returns 5

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_unshift_return&usg=ALkJrhjFp61_B3aH6wlTG_kVFzn2yAvf5A)

## Elementos em mudança

Os elementos da matriz são acessados ​​usando seu **número de índice** :

Os **índices de** matriz começam com 0. [0] é o primeiro elemento de matriz, [1] é o segundo, [2] é o terceiro ...

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits[0] = "Kiwi";        // Changes the first element of fruits to "Kiwi"

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_change&usg=ALkJrhhKAXFVOHMrGg_IyG_boW5bkjTsXQ)

A propriedade length fornece uma maneira fácil de adicionar um novo elemento a uma matriz:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits[fruits.length] = "Kiwi";          // Appends "Kiwi" to fruit

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_change_add&usg=ALkJrhj6oF8uf4pgeqV5iyn7lD4VcfOhBQ)

## Eliminando elementos

Como os arrays de JavaScript são objetos, os elementos podem ser excluídos usando o **excluir do** operador JavaScript:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
delete fruits[0];           // Changes the first element in fruits to **undefined**

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_delete&usg=ALkJrhgJKWsav8xqi6lhm_eFVhZcwo_vZw)

O uso de **exclusão** pode deixar furos indefinidos na matriz. Use pop () ou shift () em vez disso.

## Splicing a Array

O método **splice ()** pode ser usado para adicionar itens novos a uma matriz:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.splice(2, 0, "Lemon", "Kiwi");

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_splice&usg=ALkJrhhGO9nA4Ab6B_rTgXHBjrKTK0CEJQ)

O primeiro parâmetro (2) define a posição **onde** novos elementos devem ser **adicionados** (emendados).

O segundo parâmetro (0) define **quantos** elementos devem ser **removidos** .

O resto dos parâmetros ("Limão", "Kiwi") definem os novos elementos a serem **adicionados** .

## Usando splice () para remover elementos

Com uma configuração de parâmetro inteligente, você pode usar splice () para remover elementos sem deixar "holes" na matriz:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.splice(0, 1);        // Removes the first element of fruits

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_remove&usg=ALkJrhg91OvSox844G38sjtoqY9r-HeMAg)

O primeiro parâmetro (0) define a posição onde novos elementos devem ser **adicionados** (emendados).

O segundo parâmetro (1) define **quantos** elementos devem ser **removidos** .

O resto dos parâmetros são omitidos. Nenhum novo elemento será adicionado.

## Mesclando (Concatenando) Arrays

O método **concat ()** cria uma nova matriz mesclando (concatenando) arrays existentes:

### Exemplo (mesclando duas matrizes)

var myGirls = ["Cecilie", "Lone"];  
var myBoys = ["Emil", "Tobias", "Linus"];  
var myChildren = myGirls.concat(myBoys);     // Concatenates (joins) myGirls and myBoys

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_concat&usg=ALkJrhjsD8mHL5Hf40d2Db_khFjwvDULIw)

O método concat () não altera os arrays existentes. Ele sempre retorna uma nova matriz.

O método concat () pode levar qualquer número de argumentos de matrizes:

### Exemplo (Mesclando Três Arrays)

var arr1 = ["Cecilie", "Lone"];  
var arr2 = ["Emil", "Tobias", "Linus"];  
var arr3 = ["Robin", "Morgan"];  
var myChildren = arr1.concat(arr2, arr3);     // Concatenates arr1 with arr2 and arr3

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_concat2&usg=ALkJrhjBHY4Vv6XlWZCohPqr7II4SqdCVg)

O método concat () também pode ter valores como argumentos:

### Exemplo (Fusão de uma matriz com valores)

var arr1 = ["Cecilie", "Lone"];  
var myChildren = arr1.concat(["Emil", "Tobias", "Linus"]);

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_concat3&usg=ALkJrhiR4RVjdWPfn5nP7dc4gQ_nje3irQ)

## Cortando uma matriz

O método **slice ()** corta um pedaço de uma matriz em uma nova matriz.

Este exemplo corta uma parte de uma matriz a partir do elemento da matriz 1 ("Laranja"):

### Exemplo

var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];  
var citrus = fruits.slice(1);

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_slice1&usg=ALkJrhj1FXOd8iIRbXmwsZ2tbC-jqzryIA)

O método slice () cria uma nova matriz. Não remove nenhum elemento da matriz de origem.

Este exemplo corta uma parte de uma matriz a partir do elemento da matriz 3 ("Apple"):

### Exemplo

var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];  
var citrus = fruits.slice(3);

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_slice3&usg=ALkJrhg5Bn50LyJJIQwbPNLMl04D7QTvgg)

O método slice () pode levar dois argumentos como fatia (1, 3).

O método seleciona elementos do argumento de início e até (mas não incluindo) o argumento final.

### Exemplo

var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];  
var citrus = fruits.slice(1, 3);

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_slice&usg=ALkJrhg8aSAbWv2le4ziTX2cUiDlkKwvWw)

Se o argumento final for omitido, como nos primeiros exemplos, o método slice () corta o resto da matriz.

### Exemplo

var fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];  
var citrus = fruits.slice(2);

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_slice2&usg=ALkJrhg4y0m1cdDc9ZQydbIFh8PWbkK92g)

## Automatic toString ()

O JavaScript converte automaticamente uma matriz em uma seqüência separada por vírgulas quando um valor primitivo é esperado.

Este é sempre o caso quando você tenta produzir uma matriz.

Estes dois exemplos produzirão o mesmo resultado:

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo").innerHTML = fruits.toString();

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_tostring&usg=ALkJrhhJ3bBjwvMEyqNGgNDx6J854TundQ)

### Exemplo

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo").innerHTML = fruits;

[Tente você mesmo "](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/js/tryit.asp%3Ffilename%3Dtryjs_array_automatic&usg=ALkJrhheTT5NSivEdWFnd3DG1kSAaWCa0Q)

Todos os objetos JavaScript possuem um método toString ().

## Localizar valores Max e Min em uma matriz

Não há funções internas para encontrar o valor mais alto ou mais baixo em uma matriz de JavaScript.

Você aprenderá como você resolve esse problema no próximo capítulo deste tutorial.

## Classes de ordenação

As matrizes de classificação são abordadas no próximo capítulo deste tutorial.

## Referência da Array Completa

Para obter uma referência completa, acesse nossa [Referência de matriz de JavaScript completa](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&prev=_t&rurl=translate.google.com.br&sl=en&sp=nmt4&tl=pt-BR&u=https://www.w3schools.com/jsref/jsref_obj_array.asp&usg=ALkJrhivtgEcC7JAmH_oIr8W2iPd-Y5QWQ) .

A referência contém descrições e exemplos de todas as propriedades e métodos da Array.