

Array (Array) in JavaScript

Array object - Array helps you store multiple values in a single variable. It stores a set of fixed-size ranges of elements in the same type (type). An array is used to store a data set, but it is often more useful to think of an array as a collection of variables in the same type.

Array object - **Array** helps you store multiple values in a single variable. It stores a set of fixed-size ranges of elements in the same type (type). An array is used to store a data set, but it is often more useful to think of an array as a collection of variables in the same type.

Syntax

Use the following syntax to create an **Array** object:

```
var fruits = new Array ( "apple" , "orange" , "mango" );
```

The Array parameter is a list of strings or integers. When you define a single numeric parameter with the Array constructor, you specify the initialization length of the array. The maximum permitted length for an array is 4,294,967,295.

You can create an array by simply assigning the following values:

```
var fruits = [ "apple" , "orange" , "mango" ];
```

You will use the sequence number to access and set the values inside an array as follows:

```
fruits [ 0 ] is the first element fruits [ 1 ] is the second element fruits [
```

Array properties

Here are the properties of the Array object and their description.

Properties Description

constructor

Returns a reference to the array function that creates that object. index The property represents the zero-based index of the match in the string. input This attribute is present only in arrays created by regular Expression matches.

length

Reflect the number of elements in an array.

prototype

The prototype property allows you to add properties and methods to an object.

Array methods

Below is a list of the methods of the Array object along with their description.

Method Description

`concat ()`

Returns a new array that includes this array in combination with other arrays and / or values.

`every ()`

Returns true if each element in this array satisfies the given test function.

`filter ()`

Create a new array with all elements of this array, for which the filter function returns true.

`forEach ()`

Call a function for each element in the array.

`indexOf ()`

Returns the first (lowest) index of an element in the array equivalent to the given value, or -1 if not found.

`join ()`

Combine all elements in an array into a string.

`lastIndexOf ()`

Returns the last (largest) index of an element in the array equivalent to the given value, or -1 if not found.

`map ()`

Create a new array with the results of calling a given function on each element of this array.

`pop ()`

Remove the last element from an array and return that element.

`push ()`

Add one or more elements to the end of an array and return the new length of the array.

`reduce ()`

Apply a function simultaneously with two values ??of the array (from left to right) when reducing it to a single value.

`reduceRight ()`

Apply a function simultaneously with two values ??of the array (from right to left) when it is reduced to a single value.

`reverse ()`

Reverse the order of the elements of an array - First become the last and finally become the first.

`shift ()`

Remove the first element from an array and return that element.

slice ()

Extract - extract an area of an array and return a new array.

some ()

Returns true if at least one element in this array satisfies the given test function.

toSource ()

Represents the source code of an object.

sort ()

Arrange the elements of an array.

splice ()

Add and / or remove elements from an array.

toString ()

Returns a string representing the array and its elements.

unshift ()

Add one or more elements to the beginning of an array and return the new length of the array.

According to Tutorialspoint

Previous article: String object in JavaScript

Next lesson: Date object in JavaScript

You finished reading the article "**Array (Array) in JavaScript**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.