Array

JavaScript array is an object that represents a collection of similar type of elements.

There are 3 ways to construct array in JavaScript

By array literal

By creating instance of Array directly (using new keyword)

By using an Array constructor (using new keyword)

1) JavaScript array literal

The syntax of creating array using array literal is given below:

var arrayname=[value1,value2.....valueN];

As you can see, values are contained inside [ ] and separated by , (comma).

Let's see the simple example of creating and using array in JavaScript.

<script>

var emp=["Sonal","Vishal","Rakesh"];

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

document.write(emp[i] + "<br/>");

}

</script>

--------------------------------------------------------------------------------------------------

2) JavaScript Array directly (new keyword)

The syntax of creating array directly is given below:

var arrayname=new Array();

Here, new keyword is used to create instance of array.

Let's see the example of creating array directly.

<script>

var i;

var emp = new Array();

emp[0] = "Arun";

emp[1] = "Varun";

emp[2] = "John";

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

document.write(emp[i] + "<br>");

}

</script>

--------------------------------------------------------------------------------------------

3) JavaScript array constructor (new keyword)

Here, you need to create instance of array by passing arguments in constructor so that we don't have to provide value explicitly.

The example of creating object by array constructor is given below.

<script>

var emp=new Array("Jai","Vijay","Smith");

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

document.write(emp[i] + "<br>");

}

</script>

-------------------------------------------------------------------------------------------------

JavaScript Array concat() Method

The JavaScript array concat() method combines two or more arrays and returns a new string. This method doesn't make any change in the original array.

Syntax

The concat() method is represented by the following syntax:

array.concat(arr1,arr2,....,arrn)

Parameter

arr1,arr2,....,arrn - It represent the arrays to be combined.

Return

A new array object that represents a joined array.

JavaScript Array concat() Method Example

Let's see some examples of concat() method.

Example 1

Here, we will print the combination of two arrays.

<script>

var arr1=["C","C++","Python"];

var arr2=["Java","JavaScript","Android"];

var result=arr1.concat(arr2);

document.writeln(result);

</script>

-------------------------------------------------------------------

JavaScript Array entries() Method

The entries() method creates a new iterator object of an array, holding the key/value pairs for every value in the array. A key represents the index number carrying an item as its value. It does not affect the original array.

Syntax

The following syntax represents the entries() method:

array.entries()

Parameters

It does not have any parameters.

Return

It returns the newly created array iterator object. The iterator object represents each element of the array with keys assigned to them.

<html>

<head> <h5> Array Methods </h5>

<body>

<script>

var arr=['John','Michael','Embrose','Herry','Lewis'];

var itr=arr.entries();

document.write("After applying the entries method:"+"<br>");

for(var e of itr) //for loop using var.

{

document.write(e+"</br>");

}

</script>

</body>

</head>

</html>

-------------------------------------------------------------------------------------

JavaScript Array sort() method

The JavaScript array sort() method is used to arrange the array elements in some order. By default, sort() method follows the ascending order.

Syntax

The sort() method is represented by the following syntax:

array.sort(compareFunction)

Parameter

compareFunction - It is optional. It represents a function that provides an alternative sort order.

Example 1

Let's see a simple example to sort the array of string elements.

<script>

var arr=["AngularJS","Node.js","JQuery","Bootstrap"]

var result=arr.sort();

document.writeln(result);

</script>

----------------------------------------

Example 2

<!DOCTYPE html>

<html>

<body>

<script>

var arr=[2,4,1,8,5];

var result=arr.sort(function compare(a,b)

{

a-b; return

});

document.writeln(result);

</script>

</body>

</html>

-----------------------------------------------

<!DOCTYPE html>

<html>

<body>

<script>

var arr=[2,4,1,8,5];

var result=arr.sort(function compare(a,b)

{

return b-a;

});

document.writeln(result);

</script>

</body>

</html>

----------------------------------------------

Example 5

Let's see an example to extract minimum value from an array.

<script>

var arr=[2,4,1,8,5]

var result=arr.sort(); //1,2,4,5,8

document.writeln(arr[0]);

</script>

--------------------------------------

example 6: Reverse ()

<!DOCTYPE html>

<html>

<body>

<script>

var arr=[2,4,1,8,5]

var result=arr.sort().reverse(); // 8,5,4,2,1

document.writeln(arr);

</script>

</body>

</html>

--------------------------------------------------