



```
document.getElementById("demo1").innerHTML = fruits; //  
Banana, Orange, Apple, Mango  
fruits.pop(); // remove Mango  
document.getElementById("demo2").innerHTML = fruits; // Banana, Orange, Apple
```

pop() (popped out) .

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo1").innerHTML = fruits; //  
Banana, Orange, Apple, Mango  
document.getElementById("demo2").innerHTML = fruits.pop(); // Mango  
document.getElementById("demo3").innerHTML = fruits; //  
Banana, Orange, Apple
```

## Pushing

push() 가 .

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo").innerHTML = fruits;  
  
function myFunction() {  
  fruits.push("Kiwi");  
  document.getElementById("demo").innerHTML = fruits; //  
Banana, Orange, Apple, Mango, Kiwi  
}
```

push() :

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];  
document.getElementById("demo1").innerHTML = fruits; //  
Banana, Orange, Apple, Mango  
  
function myFunction() {  
  document.getElementById("demo2").innerHTML = fruits.push("Kiwi"); // 5  
  document.getElementById("demo1").innerHTML = fruits; //  
Banana, Orange, Apple, Mango, Kiwi  
}
```

## Shifting Elements

Shifting popping , .

shift() , ( ) “

```

let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Apple, Mango
fruits.shift(); // remove first element "Banana" from fruits
document.getElementById("demo2").innerHTML = fruits; // Orange, Apple, Mango

```

shift() 가 :

```

let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Apple, Mango
document.getElementById("demo2").innerHTML = fruits.shift(); // Banana
document.getElementById("demo3").innerHTML = fruits; //
Orange, Apple, Mango

```

unshift() 가 , (“unshifts” older elements):

```

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

function myFunction() {
  fruits.unshift("Lemon");
  document.getElementById("demo").innerHTML = fruits; //
Lemon, Banana, Orange, Apple, Mango
}

```

unshift() .

```

let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Apple, Mango
document.getElementById("demo2").innerHTML = fruits.unshift("Lemon"); // 5
document.getElementById("demo3").innerHTML = fruits; //
Lemon, Banana, Orange, Apple, Mango

```

## Changing Elements

(index number)

0                    . [0]                    , [1]                    , [2]                    ...

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Apple, Mango
fruits[0] = "Kiwi"; // index[0] Banana "Kiwi"
document.getElementById("demo2").innerHTML = fruits; //
Kiwi, Orange, Apple, Mango
```

length                    가                    .

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

function myFunction() {
  fruits[fruits.length] = "Kiwi"; // fruits                    'Kiwi'
  가
  document.getElementById("demo").innerHTML = fruits; //
Banana, Orange, Apple, Mango, Kiwi
}
```

## Deleting Elements

JavaScript                    , JavaScript                    delete                    :

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = "The first fruit is: " +
fruits[0]; // The first fruit is: Banana
delete fruits[0]; // index[0] Banana
document.getElementById("demo2").innerHTML = "The first fruit is: " +
fruits[0]; // The first fruit is: undefined
console.log(fruits); // (4) [empty, "Orange", "Apple", "Mango"]
```

delete                    (holes)                    .                    pop()  
shift()                    .

## Splicing an Array

`splice()` (items) 가 .

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = "Original Array: " + fruits;
// Original Array: Banana,Orange,Apple,Mango
function myFunction() {
  fruits.splice(2, 0, "Lemon", "Kiwi");
  document.getElementById("demo2").innerHTML = "New Array: " + fruits; //
  New Array: Banana,Orange,Lemon,Kiwi,Apple,Mango
}
```

(2) 가 가 ( ) .  
 (0) .  
 ("Lemon", "Kiwi") 가 .

`splice()` .

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = "Original Array: " + fruits; //
Original Array: Banana,Orange,Apple,Mango

function myFunction() {
  let removed = fruits.splice(2, 2, "Lemon", "Kiwi");
  document.getElementById("demo2").innerHTML = "New Array: " + fruits; //
  New Array: Banana,Orange,Lemon,Kiwi
  document.getElementById("demo3").innerHTML = "Removed Items: " + removed;
  // Removed Items: Apple,Mango
}
```

## Using splice() to Remove Elements

, splice() “ ” ,

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

```
function myFunction() {  
  fruits.splice(0, 1);  
  document.getElementById("demo").innerHTML = fruits; // Orange,Apple,Mango  
}
```

(0) 가 가 ( )

(1)

가 가

## Merging (Concatenating) Arrays

concat() ( )

### Merging Two Arrays

```
let myGirls = ["Cecilie", "Lone"];  
let myBoys = ["Emil", "Tobias", "Linus"];  
let myChildren = myGirls.concat(myBoys);  
  
document.getElementById("demo").innerHTML = myChildren; //  
Cecilie,Lone,Emil,Tobias,Linus
```

'concat()'

'concat()'

### Merging Three Arrays

```
let arr1 = ["Cecille", "Lone"];  
let arr2 = ["Emil", "Tobias", "Linus"];  
let arr3 = ["Robin", "Morgan"];  
  
let myChildren = arr1.concat(arr2, arr3);  
  
document.getElementById("demo").innerHTML = myChildren;
```

```
//Cecille,Lone,Emil,Tobias,Linus,Robin,Morgan
```

concat()

### Merging an Array with Values

```
let myChildren = arr1.concat("Peter");
document.getElementById("demo").innerHTML = myChildren; //
Emil,Tobias,Linus,Peter
```

### Slicing an Array

slice()

1 ("Orange")

```
let fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
let citrus = fruits.slice(1); // index[1]~index[4]
citrus
document.getElementById("demo1").innerHTML = fruits; //
Banana,Orange,Lemon,Apple,Mango
document.getElementById("demo2").innerHTML = citrus; //
Orange,Lemon,Apple,Mango
```

'slice()'

3 ("Apple")

```
let fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
let citrus = fruits.slice(3); // 3,4 citrus
document.getElementById("demo1").innerHTML = fruits; //
Banana,Orange,Lemon,Apple,Mango
document.getElementById("demo2").innerHTML = citrus; // Apple,Mango
```

slice()

slice(1, 3)

( )

```
let fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
let citrus = fruits.slice(1, 3); // 1,2 citrus
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Lemon, Apple, Mango
document.getElementById("demo2").innerHTML = citrus; // Orange, Lemon
```

, end 가 slice() .

```
let fruits = ["Banana", "Orange", "Lemon", "Apple", "Mango"];
let citrus = fruits.slice(2); // 2, 3, 4
citrus
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Lemon, Apple, Mango
document.getElementById("demo2").innerHTML = citrus; // Lemon, Apple, Mango
```

## Automatic toString()

JavaScript (primitive value) , .

가 .

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo2").innerHTML = fruits.toString(); //
Banana, Orange, Apple, Mango
```

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits; //
Banana, Orange, Apple, Mango
```

JavaScript toString() 가 .

# Finding Max and Min Values in an Array

JavaScript 가 가

(JS Array Sort)

## Sorting Arrays

(sorting arrays)

(JS Array Sort)

## Complete Array Reference

[Complete JavaScript Array Reference](#)

가

, [Javascript](#), [Array](#), [Methods](#)

From: <http://rwiki.repia.com/> -

. - 2023.12

Permanent link: [http://rwiki.repia.com/doku.php?id=wiki:javascript:javascript\\_note:js\\_array\\_methods&rev=1619494024](http://rwiki.repia.com/doku.php?id=wiki:javascript:javascript_note:js_array_methods&rev=1619494024)

Last update: 2022/03/10 19:52

