

# Array pop()

The `pop()` method is a method applied on arrays which removes the last element of the array. i.e it removes the element at the highest index value – `array[array.length - 1]`. When it is applied to an array, the length of the array changes – reduces by 1.

## Syntax

`array.pop()`

The method takes no arguments.

## Return value

The method has a return value which is the element it pops out (removes). If applied on an empty array, the return value is `undefined` (obviously, because the array doesn't even have any value in the first place);

## Example

```
let array1 = ["code", "source", {name: "website"}];
let popped = array1.pop()
console.log(array1, popped);
// Expected Output
// ["code", "source"];
// {name: "website"}
```

`array1` in the example above had three values. When the method was applied, only two values are left while the last value is removed (returned).

As seen in the code above, the object is the popped value which is returned by the method.

# Array push()

This method adds an element to the end of an array. When applied the arrays, the length increases by 1 where which the index of the element is `array.length - 1`

## Syntax

`array.push(value);`

The value which is an argument passed to the method could be a number, string, object or even array.

### Return Value

The return value is the new length of array.

### Example

```
let arr = [2, "code"];
let newArrLength = arr.push({name: "website"});
console.log(arr, newArrLength);
// Expected output
// [2, "code", {name: "website"}]
// 3
```

## Array shift()

This method, similar to `pop()` removes the first element of the array. When applied to arrays, this results in a decrease in length by 1

### Syntax

```
array.shift()
```

The method takes no argument.

### Return Value

The return value is the element which was removed from the array. If the array is empty, the return value would be `undefined`.

### Example

```
let arr = ["code", {name: "website"}, 56];
let shiftedValue = arr.shift();
console.log(arr, shiftedValue);
// Expected Output
// [{name: "website"}, 56]
// "code"
```

## Array unshift()

This method which is similar to `push()` adds an element to the beginning of the array. When applied, the array length increases by 1.

## Syntax

```
array.unshift(value);
```

The method takes an argument of value which could be object, array, string, etc.

## Return Value

The return value is the new length of the array after adding the element.

## Example

```
let arr = [{name: "website"}];
let newArrLength = arr.unshift([2, "code"]);
console.log(arr, newArrLength);
// Expected Output
// [[2, "code"], {name: "website"}]
// 2
```