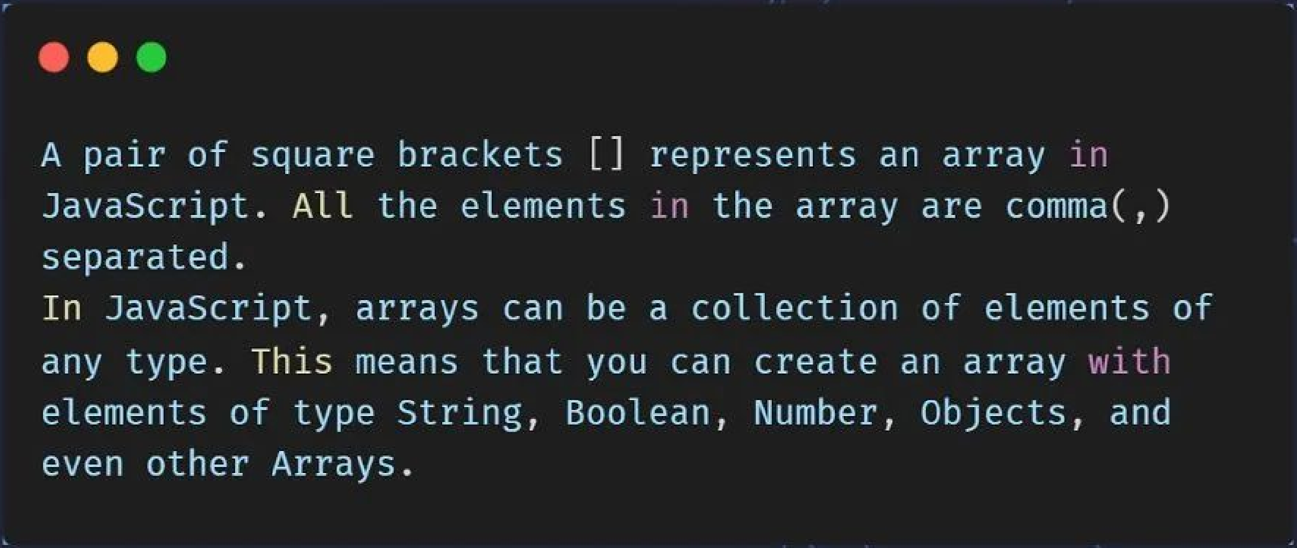
# TOP JAVASCRIPT ARRAY METHODS WITH EXAMPLES

@javascript.js

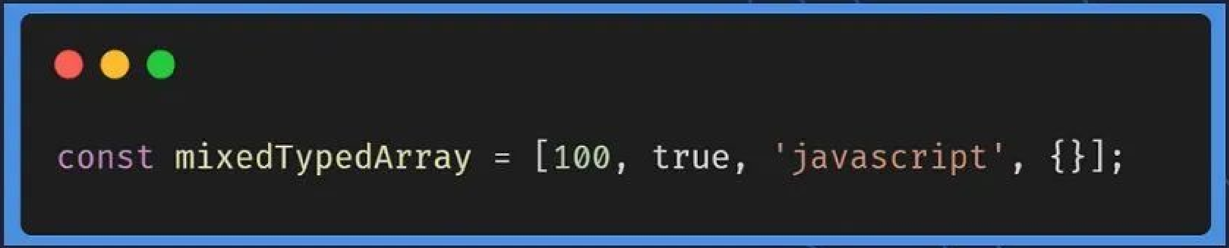
1

# What is an Array in JavaScript?



A pair of square brackets [] represents an array in JavaScript. All the elements in the array are comma(,) separated.

In JavaScript, arrays can be a collection of elements of any type. This means that you can create an array with elements of type String, Boolean, Number, Objects, and even other Arrays.

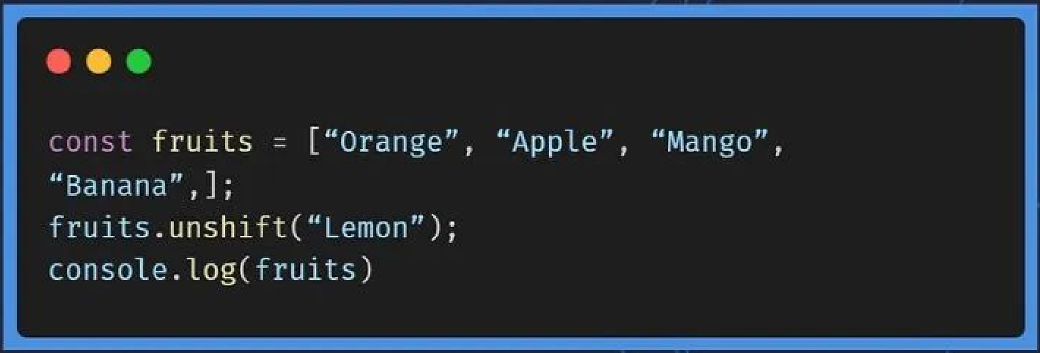


```
const mixedTypedArray = [100, true, 'javascript', {}];
```

2

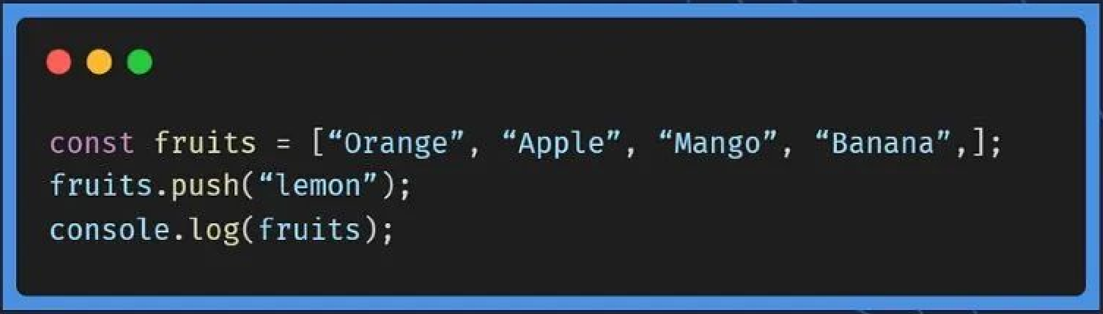
# Add Element in Array

The **unshift( )** the method adds a new element to an array (**at the beginning**), and "unshifts" older elements:



```
const fruits = [“Orange”, “Apple”, “Mango”,
“Banana”,];
fruits.unshift(“Lemon”);
console.log(fruits)
```

The **push( )** the method adds a new element to an array (**at the end**):

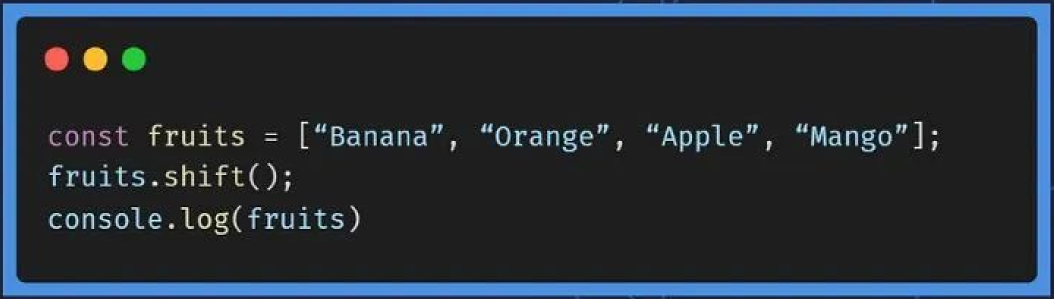


```
const fruits = [“Orange”, “Apple”, “Mango”, “Banana”,];
fruits.push(“lemon”);
console.log(fruits);
```

3

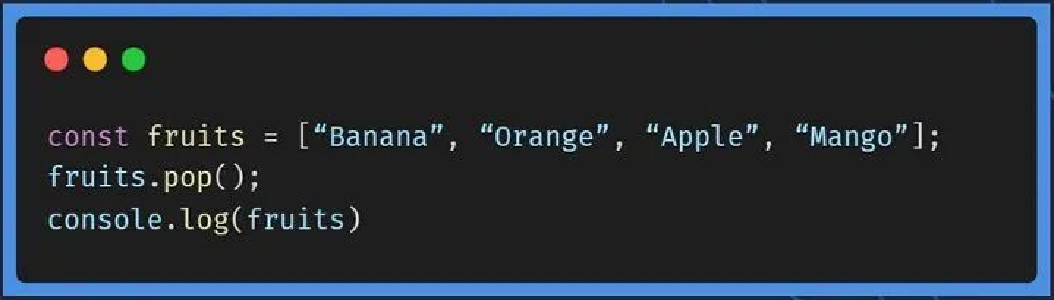
# Delete Element

The **shift( )** the method removes the first array element and "shifts" all other elements to a lower index.



```
const fruits = [“Banana”, “Orange”, “Apple”, “Mango”];
fruits.shift();
console.log(fruits)
```

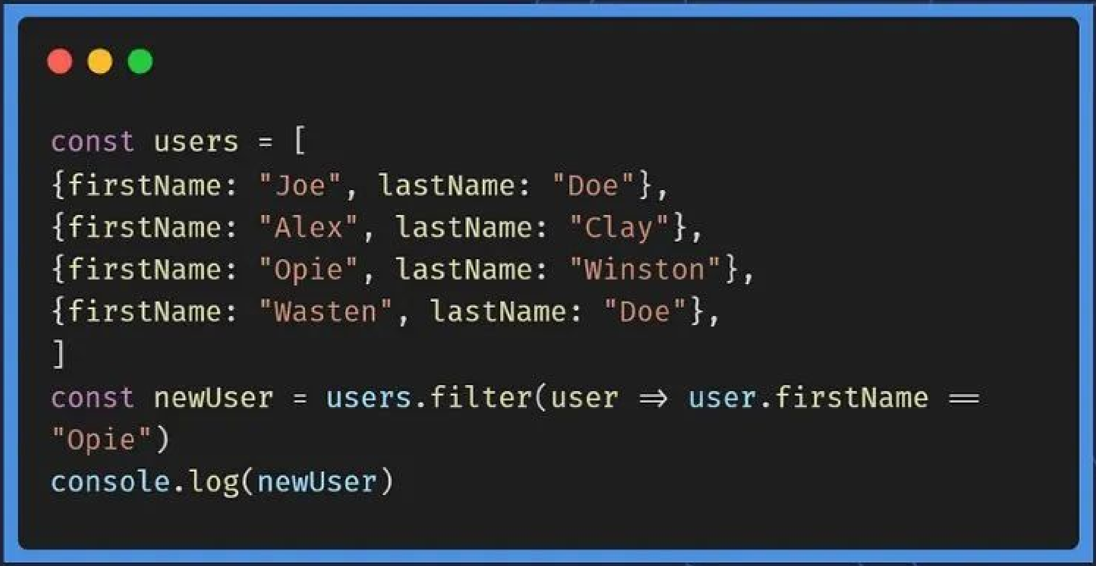
The **pop( )** the method removes the last element from an array:



```
const fruits = [“Banana”, “Orange”, “Apple”, “Mango”];
fruits.pop();
console.log(fruits)
```

# Filter Elements

The **filter( )** the method creates a new array filled with elements that pass a test provided by a function.
The **filter( )** the method does not execute the function for empty elements.

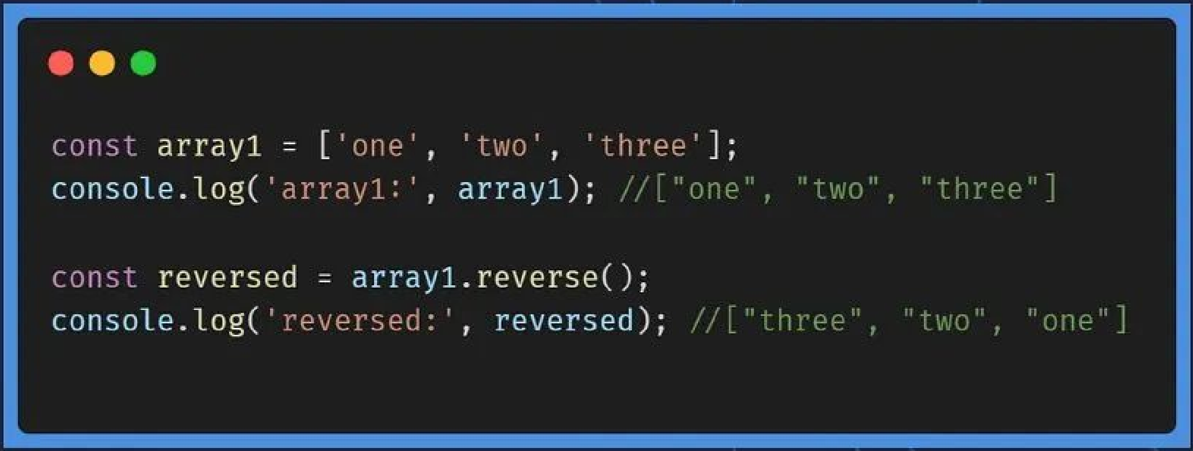


```
const users = [
{firstName: "Joe", lastName: "Doe"},
{firstName: "Alex", lastName: "Clay"},
{firstName: "Opie", lastName: "Winston"},
{firstName: "Wasten", lastName: "Doe"},
]
const newUser = users.filter(user => user.firstName ==
"Opie")
console.log(newUser)
```

5

# Reverse in JavaScript

The **reverse( )** method reverses the order of the elements in an array.
The **reverse( )** method overwrites the original array.

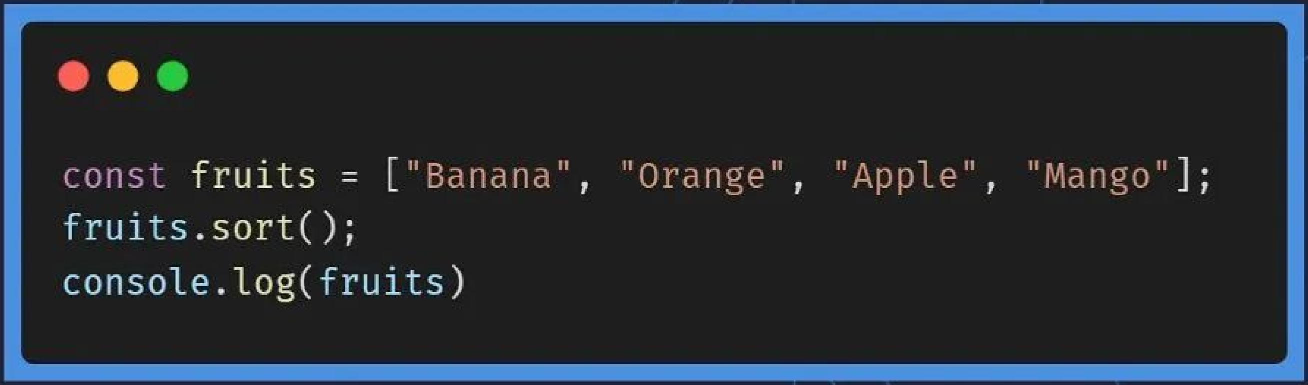


```
const array1 = ['one', 'two', 'three'];
console.log('array1:', array1); //["one", "two", "three"]

const reversed = array1.reverse();
console.log('reversed:', reversed); //["three", "two", "one"]
```

# SORT in JavaScript

The **sort( )** method sorts an array alphabetically:



```
const fruits = ["Banana", "Orange", "Apple", "Mango"];
fruits.sort();
console.log(fruits)
```