



# JavaScript reduce(), some(), and every() method explained

Next →

# .reduce()

Runs a calculation on every element of an array and passes the result of previous iterations. Returns the result once all elements have been processed

the result of  
previous iterations

the current  
element in the  
array



```
const nums = [10, 8, 2, 5]
```

```
const sum = nums.reduce((runningTotal, value) => {  
  return runningTotal + value  
}, 0);
```

```
console.log(sum) // 25
```

the result after  
all elements have  
been processed

initial runningTotal  
(before any iterations)

Next →

# .some()

Array.prototype.some()

Loops through an array and checks a condition that's provided by the given callback function against each element.

It exits early and returns true once it finds an element that passes the condition.



```
const fruit = ["pear", "banana", "apple", "melon"];

const containsApple = fruit.some((el) => {
  return el === "apple"
});

console.log(containsApple); // true
```

condition/test

true if any of the  
elements meet the  
condition, otherwise false

Next →

# .every()

Loops through an array and checks that every element meets a condition provided by the given callback function.

Returns true if all elements pass the condition, otherwise returns false.



```
const prices = [9.99, 25, 30, 49];  
  
const isAllBelowThreshold = prices.every((price) => {  
  return price < 50  
});  
  
console.log(isAllBelowThreshold); // true
```

condition/test

true if all elements  
meets the condition,  
otherwise false

Next →

**i post about Tech,  
Coding and Career**

Follow for more content like this

@alighouri