Day 4: Count Objects

by **AvmnuSng**

Problem	Submissions	Leaderboard	Discussions	Editorial 🔒	Tutorial

JavaScript: Iterating Over Arrays

In this article, we demonstrate some of the ways of iterating over an array's elements in JavaScript.

Basic *for* Loop

```
1  const arr = ['a', 'b', 'c', 'd'];
2
3  // 'i' is the index
4  for (let i = 0; i < arr.length; i++) {
5      console.log('arr[' + i + ']: ' + arr[i]);
6  }
```

Output

Run

The *for...in* Statement

OK

The *for...in* statement iterates over the enumerable properties of an object in an arbitrary order, which allows us to execute statements for each distinct property. In the case of an array, the property would be its elements. In the case of an object, that would be its properties.

Iterating Over an Object's Properties

```
 1  const o = {
 2      a: 1,
 3      b: 2,
 4      c: 3,
 5      d: 4
 6  };
 7
 8  console.log('property: value');
 9  // 'p' is the property
10  for (p in o) {
11      console.log(p + ': ' + o[p]);
12  }
```
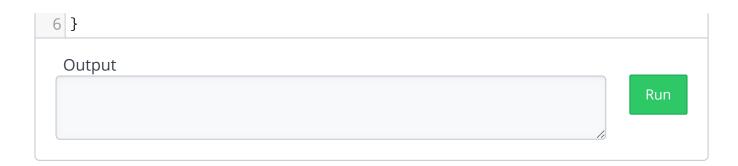
Output

[ ]   Run

─   EXAMPLE

Iterating Over an Array's Elements

When we use this loop to iterate over an array, the array's indices behave like properties.

```
 1  const o = ['first', 'second', false];
 2
```

```
6 }
```

Output

Run

The *forEach* Method

The *forEach* method iterates through an array and, for each element, it executes a function once.

−	EXAMPLE

```
1  const arr = ['a', 'b', 'c', 'd'];
2
3  arr.forEach((value, index, array) => {
4      console.log('index', index, 'has a value of', value,
5      'which correlates to array[' + index + ']:', array[index]);
6  });
7
8  arr.forEach((value, index) => {
9      console.log('index', index, 'has a value of', value);
10 });
11
12 arr.forEach((value) => {
```

Output

Run

Resources

-	L O O P S	Recommended Article

Loops

View

Check out our article on JavaScript loops for more information.