

数组方法 `map()` 和 `forEach()` 有什么区别？

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1547790109416>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

2019-01-18

回答

这两种方法都是对数组中的元素进行迭代。`map()` 通过每个元素的回调函数将其映射到新的元素上，最终返回一个新的数组。`forEach()` 虽然也为每一个元素准备了回调函数，但却不返回新的数组。`forEach()` 在每一次迭代的使用中都会产生副作用，因此 `map()` 是编程技术中常用的方法。

加分回答

- 如果你需要迭代一个数组，使其本身发生变化且不需要返回一个新数组时，可以使用 `forEach()`
- 如果你只是对数字进行遍历时，也可以使用 `forEach()`
- `map()` 是保持原有数组不变的正确选择，他可以让原始数组的每一个值都映射到新的数组上
- `map()` 运行的较快，且返回的新数组可以让你继续使用 `map()`、`filter()`、`reduce()` 等方法，如：

```
const arr = [1, 2, 3, 4, 5];
const arr1 = arr.map(num => num * 2).filter(num => num > 5);
console.log(arr); // [1, 2, 3, 4, 5]
console.log(arr1); // [6, 8, 10]
```

返回总目录

每天 30 秒