



链滴

# Array - mapObject (advanced)

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1558082057214>

来源网站: [链滴](#)

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

## 描述

把数组中的值转换为一个 key-value 对象，这个对象使用原始值作为 key，对原始值使用提供的方法返回的结果作为对应的 value。

## 提示

- 使用一个匿名的内部函数作用域来声明一个没有定义的内存空间
- 使用闭包来存储返回值
- 使用一个新的 `Array` 来存储原始数组及其每一个元素经过指定函数运行的结果值，如：`[arr, arr.map(fn)]`
- 使用 `Array.prototype.reduce` 把新数组转换为相应的 key-value 对象
- 使用强制运算符 `()` 获取最后一个表达式的运算结果
- 由于闭包和特定的操作，我们就不需要把一个上下文挪动到另一个上下文中

## 代码

```
const mapObject = (arr, fn) =>
  (a => (
    (a = [arr, arr.map(fn)], a[0].reduce((acc, val, ind) => ((acc[val] = a[1][ind]), acc), {})
  ))());
```

## 示例

获取数组中每个元素及其平方的对象：

```
const squareIt = arr => mapObject(arr, a => a * a);
squareIt([1, 2, 3]); // { 1: 1, 2: 4, 3: 9 }
```

## 返回总目录

[每天 30 秒系列之 JavaScript 代码](#)