



链滴

# Array - toHash

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1561083506048>

来源网站: [链滴](#)

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

2019-06-21

## 描述

将给定的类似数组转换为一个对象值（键值存储）。

## 提示

- 给出对象需是一个可迭代的或者类似数组的结构
- 对提供的对象调用 `Array.prototype.reduce.call()`
- 键为提供的对象的 `key` 值，逐个元素迭代计算后返回最终结果
- 给定 `key` 的默认值为索引

## 代码

```
const toHash = (object, key) =>
  Array.prototype.reduce.call(
    object,
    (acc, data, index) => ((acc[!key ? index : data[key]] = data), acc),
    {}
  );
```

## 示例

将数组转换为对象：

```
toHash([4, 3, 2, 1]); // { 0: 4, 1: 3, 2: 2, 3: 1 }
toHash([ { a: 'label' } ], 'a'); // { label: { a: 'label' } }
```

返回管理员及其雇员的明细：

💡 `Array.prototype.map()` 的第二个参数为执行 `callback` 函数时使用的 `this` 值。

```
let users = [ { id: 1, first: 'Jon' }, { id: 2, first: 'Joe' }, { id: 3, first: 'Moe' } ];
let managers = [ { manager: 1, employees: [2, 3] } ];
managers.forEach(
  manager =>
    (manager.employees = manager.employees.map(function(id) {
      return this[id];
    }, toHash(users, 'id')))
);
managers; // [ { manager:1, employees: [ { id: 2, first: "Joe" }, { id: 3, first: "Moe" } ] } ]
```

## 返回总目录

[每天 30 秒系列之 JavaScript 代码](#)