



链滴

关联数组转换为树对象

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1596676537177>

来源网站: 链滴

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

2020-08-06

描述

将一个平铺的有相互关联关系的数组对象转换为一个使用递归嵌套的树形对象。这对嵌套评论非常有，例如：reddit.com。

提示

- 使用递归
- 使用 `Array.prototype.filter()` 过滤出同一链接关系的所有节点
- 使用 `Array.prototype.map()` 迭代每一个过滤出来的新对象，并为其添加一个 `children` 属性，属性值使用递归嵌套获取当前节点的所有子节点
- 第二个参数 `id` 的默认值为 `null`，用于表示一个没有链接到任何对象上的对象（如：一个顶层对象根节点）
- 第三个参数 `link` 的默认值为 `parent_id`，用于指定一个对象通过自身 `id` 链接到另一个对象的键

代码

```
const nest = (items, id = null, link = 'parent_id') =>
  items
    .filter(item => item[link] === id)
    .map(item => ({ ...item, children: nest(items, item.id, link) }));
```

示例

将平铺对象转换为嵌套的树形结构：

```
const comments = [
  { id: 1, parent_id: null },
  { id: 2, parent_id: 1 },
  { id: 3, parent_id: 1 },
  { id: 4, parent_id: 2 },
  { id: 5, parent_id: 4 }
];
const nestedComments = nest(comments); // [{ id: 1, parent_id: null, children: [{ children: [{ children: [{ children: [], id: 5, parent_id: 4 }], id: 4, parent_id: 2 }], id: 2, parent_id: 1 }, { children: [], id: 3, parent_id: 1 } ]}]
```

返回总目录

[每天 30 秒系列之 JavaScript 代码](#)