



链滴

# Array - unzipWith (advanced)

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1561945854495>

来源网站: [链滴](#)

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

2019-06-30

## 描述

创建一个由一系列元素组成的数组，通过 [zip](#) 和提供的函数来分解原始数组中的元素。

## 提示

- 使用 `Math.max.apply()` 获取数组中最长的子数组长度
- 使用 `Array.prototype.map()` 为每一个元素创建一个数组
- 使用 `Array.prototype.reduce()` 和 `Array.prototype.forEach()` 来对数组中的元素进行元素的重分组
- 使用 `Array.prototype.map()` 和扩展运算符 `...` 让每一组中的元素都使用提供的 `fn` 函数

## 代码

```
const unzipWith = (arr, fn) =>
  arr
    .reduce(
      (acc, val) => (val.forEach((v, i) => acc[i].push(v)), acc),
      Array.from({
        length: Math.max(...arr.map(x => x.length))
      }).map(x => [])
    )
    .map(val => fn(...val));
```

## 示例

对数组中的同列元素进行相加：

```
unzipWith([[1, 10, 100], [2, 20, 200]], (...args) => args.reduce((acc, v) => acc + v, 0)); // [3, 30, 00]
```

## 返回总目录

[每天 30 秒系列之 JavaScript 代码](#)