



ES10

JSON.stringify()

JSON.stringify 在 ES10 修复了一些超出范围的 Unicode 展示错误的问题。因为 JSON 都是被编码成 UTF-8，所以遇到 0xD800–0xDFFF 的字符会因为无法编码成 UTF-8 进而导致显示错误。在 ES10 它会用转义字符的方式来处理这部分字符而非编码的方式，这样就会正常显示了。

```
JSON.stringify('\u{D800}') === '"\\ud800"' // true
```

Array.prototype.flat()

flat() 方法会按照一个可指定的深度递归遍历数组，并将所有元素与遍历到的子数组中的元素合并为一个新数组返回。

语法

```
const newArray = arr.flat(depth)
```

解释

参数	含义	必选
depth	指定要提取嵌套数组的结构深度，默认值为 1	N

示例

```
const numbers = [1, 2, [3, 4, [5, 6]]]
console.log(numbers.flat())
// [1, 2, 3, 4, [5, 6]]
```

[TIP]

此时 flat 的参数没有设置，取默认值 1，也就是说只扁平化向下一级，遇到 [3, 4, [5, 6]] 这个数组会扁平化处理，不会再继续遍历内部的元素，是否还有数组

```
const numbers = [1, 2, [3, 4, [5, 6]]]
console.log(numbers.flat(2))
// [1, 2, 3, 4, 5, 6]
```

当 flat 的参数大于等于 2，返回值就是 [1, 2, 3, 4, 5, 6] 了。

Array.prototype.flatMap()

flatMap() 方法首先使用映射函数映射每个元素，然后将结果压缩成一个新数组。从方法的名字上也可以看出来它包含两部分功能一个是 map，一个 flat（深度为1）。

语法

```
const new_array = arr.flatMap(function callback(currentValue[, index[, array]]) {
  // 返回新数组的元素
}, thisArg)
```

解释

参数	含义	必选
----	----	----