那么什么是类[数组](https://so.csdn.net/so/search?q=%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)对象呢？所谓类数组对象，最基本的要求就是具有length属性的对象。

let arrayLike = {

0: *'tom',*

1: *'65',*

2: *'男',*

3: [*'jane','john','Mary'],*

*'length': 4*

}

let arr = Array.from(arrayLike)

console.log(arr) // [*'tom','65','男',['jane','john','Mary']]*

那么，如果将上面代码中length属性去掉呢？实践证明，答案会是一个长度为0的空数组。

这里将代码再改一下，就是具有length属性，但是对象的属性名不再是数字类型的，而是其他字符串型的，代码如下：  
　　  
1、将类数组对象转换为真正数组：

let arrayLike = {

*'name': 'tom',*

*'age': '65',*

*'sex': '男',*

*'friends': ['jane','john','Mary'],*

length: 4

}

let arr = Array.from(arrayLike)

console.log(arr) // [ undefined, undefined, undefined, undefined ]

会发现结果是长度为4，元素均为undefined的数组

由此可见，要将一个类数组对象转换为一个真正的数组，必须具备以下条件：

**1、该类数组对象必须具有length属性，用于指定数组的长度。如果没有length属性，那么转换后的数组是一个空数组。  
2、该类数组对象的属性名必须为数值型或字符串型的数字  
ps: 该类数组对象的属性名可以加引号，也可以不加引号**

2、将Set结构的数据转换为真正的数组：

Set 对象允许你存储任何类型的唯一值，无论是原始值或者是对象引用。

let arr = [12,12,45,97,9797,564,134,45642]

let set = new Set(arr) {12,45,97,9797,564,134,4562} //去重

console.log(Array.from(set)) // [ 12, 45, 97, 9797, 564, 134, 45642 ]

3.　Array.from还可以接受第二个参数，作用类似于数组的map方法，用来对每个元素进行处理，将处理后的值放入返回的数组。如下：

let arr = [12,45,97,9797,564,134,45642]

let set = new Set(arr)

console.log(Array.from(set, item => item + 1)) // [ 13, 46, 98, 9798, 565, 135, 45643 ]

1. 将字符串转换为数组：

let str = *'hello world!';*

console.log(Array.from(str)) // ["h", "e", "l", "l", "o", " ", "w", "o", "r", "l", "d", "!"]

4、Array.from参数是一个真正的数组：

console.log(Array.from([12,45,47,56,213,4654,154]))

像这种情况，Array.from会返回一个一模一样的新数组