

Array数组

数组：一组有序数据的集合

创建数组的三种方法

- 方法一：
var 数组名 = new Array();
 数组名[0] = 值;
 数组名[1] = 值;
 ...
- 方法二：
var 数组名 = new Array(值,...)
- 方法三：
var 数组名 = [值,...]

遍历数组

属性

length 返回数组中元素的个数

遍历

- for循环
- for (var 索引 in 数组) { ... }
- for (var 值 of 数组) { ... }
- arr.forEach (function (值,索引) {
 ...
})

方法

- indexOf("元素")
 返回元素在数组中第一次出现的位置索引
 不存在返回-1
- lastIndexOf("元素")
 返回元素在数组中最后一次出现的位置索引
 不存在返回-1
- toString()
 用逗号将数组元素连接为字符串并返回
- join('连接符')
 用指定连接字符将数组元素连接为字符串并返回
- unshift(值,...) 头部追加新元素，返回新数组长度
- push(值,...) 尾部追加新元素，返回新数组长度

- shift() 删除第一个元素，返回被删除元素的值
- pop() 删除最后一个元素，返回被删除元素的值
- splice(开始位置,数量,新元素,新元素,.....)
删除元素或删除并插入新元素， 返回包括删除元素所组成的新数组
- slice(开始位置,结束位置)
截取时不包含结束位置元素， 返回截取元素组成的新数组， 不影响原数组
- sort() 将数组元素升序排序

```
sort(function(x,y){
    return x - y;
})
```

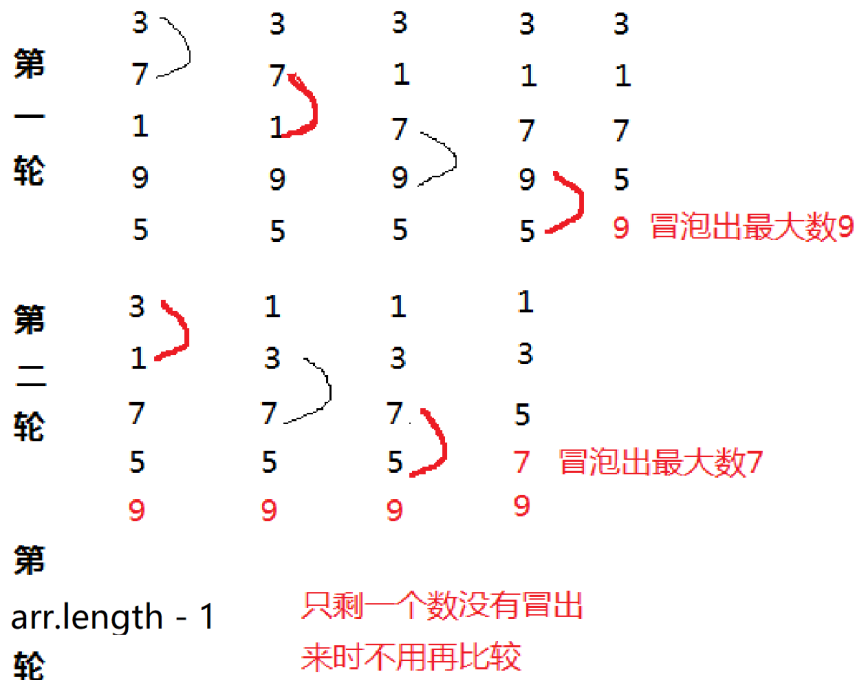
- reverse() 颠倒数组中的元素的顺序
- concat(arr1,...) 连接多个数组返回新数组,不影响原数组

冒泡排序

冒泡排序是一种的排序算法。它依次比较两个相邻元素， 如果发现大小顺序错误立即进行位置交换， 这样最大(或最小)的元素会经由交换慢慢“浮”出

已知 : var arr = [3,7,1,9,5];

要求 : 使用冒泡排序算法实现数组元素的升序排列



1. 每一个位置的数依次和其后位置的数进行比较，如果比后一位置的数大则互换位置
2. 通过每一轮的比较冒出当前剩余数中的最大数
3. 冒泡出来的最大数不再参与下一轮的比较