数组的方法

    arr.push()

    arr.unshift()

    arr.pop()

    arr.shift()

    arr.slice()

    arr.splice()

    arr.join()

arr.find()

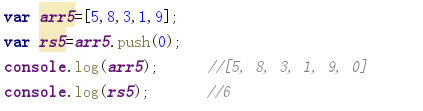
arr.indexOf()

arr.findIndex()

arr.reverse()

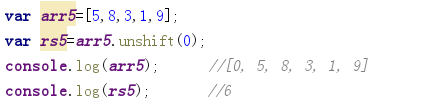
8.arr.push()

    往数组最后添加一个或几个元素，返回产生新数组的长度。



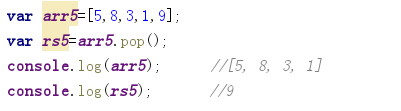
9.arr.unshift()

    往数组最前面添加一个或多个元素，返回产生新数组的长度。



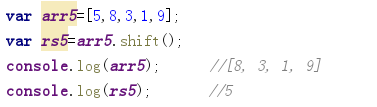
9.arr.pop()

    将数组最后一个元素删除，返回删除的元素。



10.arr.shift()

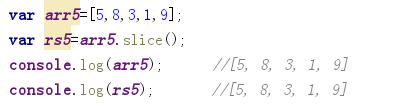
    将数组最前面的一个元素删除，返回删除的元素。



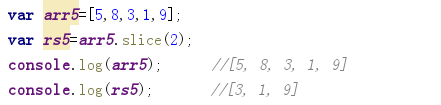
11.arr.slice()

    返回数组。截取数组中的元素，返回截取的元素组成的数组。不改变原数组。

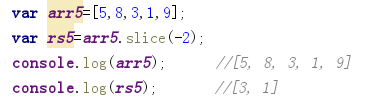
    **没有参数：**截取整个数组。



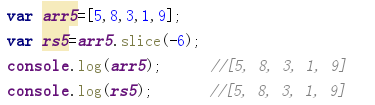
    **有一个参数：**从指定的位置开始截取到最后一位元素，包括指定的位置的元素。



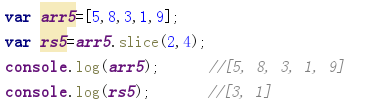
    **有一个参数，参数是负数：**从指定的位置开始截取到最后一位元素，包括指定的位置的元素。开始的位置是arr的长度加负数。



    负数绝对值大于arr的长度，则从0开始截取。



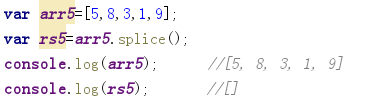
    **有两个参数：**从指定的位置开始截取到指定的位置，包括起始的指定位置的元素，不包括结束的指定位置的元素。



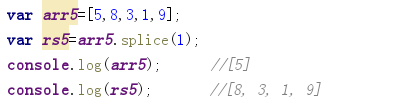
12.arr.splice()

    返回数组。删除或添加元素，返回删除的元素组成的数组。改变原数组。

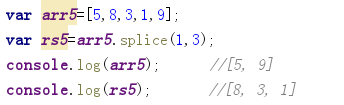
    **无参数：**不删除元素。



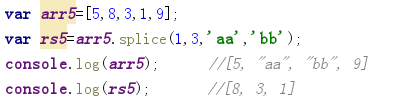
    **有一个参数：**从指定的位置开始删除到最后一位元素，包括指定的位置的元素。



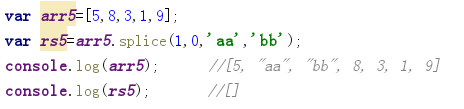
    **有两个参数：**从指定的位置开始删除，删除指定的个数，包括起始的指定位置的元素。



    **三个以上参数：**从指定的位置开始删除，删除指定的个数，包括起始的指定位置的元素，并且添加第三位以后的元素。

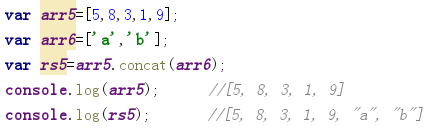


    **三个以上参数，第二个参数是0：**从指定的位置之前添加第三位以后的元素。



13.arr.concat()

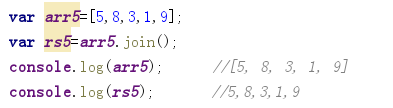
    返回合并之后的数值。



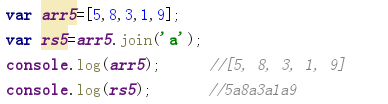
14.arr.join()

    将数值转出字符串，没有参数，默认用逗号隔开。

    **无参数：**



    **参数：**



15.arr.find()

找到满足回调函数条件的元素，没有则undefined。

循环数组，每循环一次，执行一次数组，当数组中的元素满足回调函数的条件时，停止循环，并返回该元素。

**var *arr5***=[5,8,3,1,9];

**var *rs5***=***arr5***.find(**function** (value) {

**return** value>4;

});

***console***.log(***arr5***); *//[5, 8, 3, 1, 9]*

***console***.log(***rs5***); *//5*

16.arr.indexOf()

判断数组中是否有指定的元素，没有则返回-1。

17.arr.findIndex()

找到满足回调函数条件的元素，没有则返回-1.。

**var *arr5***=[5,8,3,1,9];

**var *rs5***=***arr5***.findIndex(**function** (value) {

**return** value>10;

});

***console***.log(***arr5***); *//[5, 8, 3, 1, 9]*

***console***.log(***rs5***); *//1*