

数组对象和字符串对象

数组对象

```
new Array()
```

属性

- `arr.length` 保存的是数组的当前长度
- `constructor` 可以获取该对象的构造函数

```
console.log(aa.constructor)
```

- `instanceof` 判断该对象是不是该构造函数的实例

```
console.log(arr instanceof Array)
```

- `arr.push()`
 - 从末尾添加元素---返回值为新数组的长度，会对原数组造成影响
- `arr.unshift()`
 - 从开头添加元素---返回值为新数组的长度，会对原数组造成影响
- `arr.pop()`
 - 从末尾删除一个元素---返回值为删除的元素，会对原数组造成影响 (一次只能删除一个)
- `arr.shift()`
 - 从开头删除一个元素---返回值为删除的元素，会对原数组造成影响
- `arr.splice()`
 - 万能添加删除函数。。。从开头删除一个元素，返回删除的值---参数可以很多。第一个参数为要操作的位置(下标)，第二个为要删除的个数，第三个为要添加的元素，返回值为删除的元素组成的数组。同样会对原数组造成影响。
- `arr.slice()`
 - 截取---参数可以是两个，第一个参数为截取的开始位置，第二个为截取的结束位置，也可以不指定结束位置，(可以是负数)默认从开始位置，一直截取到最后，返回值为截取的数组，不会对原数组造成影响。
- `arr.join()`

- 把数组转化为字符串---默认用“,”逗号进行连接, 转化为字符串, 不会对原数组造成影响, 可以接受一个参数, 为指定连接的字符。

- `arr.concat()`

- 连接---可以同时连接多个数组, 返回值为新的数组, 不会对原数组造成影响。

- `arr.sort()`

- 排序---不传参数, 默认是以ASCII码的大小进行排序。若传参数, 必须是一个回调函数。回调函数接受两个参数

```
var arr = [3,6,2,7,2,8,9]

var result=arr.sort(function(m,n){

    // return m-n;    //升序

    return n-m;      //降序

});

console.log(result)

console.log(arr)
```

字符串对象

- `str.charAt()`

- 返回指定位置的字符, 不会对原字符串有影响

- `str.charCodeAt()`

- 返回指定位置的ASCII码或unicode编码, 不会对原字符串有影响

- `String.fromCharCode()`

- 将指定的unicode编码转化成字符

- `str.lastIndexOf()`

- 返回指定字符最后一次出现的位置, 不会对原字符串有影响

- `str.indexOf()`

- 返回指定字符第一次出现的位置, 不会对原字符串有影响, 如果没有, 返回-1。

- `str.replace()`

- 找到的替换掉, 找不到的原样输出 第一个值: 替换什么。第二个值: 替换成什么。

- `match()`

- `search()`

- `str.slice(1,-1)`

- 从开始的位置截取到结束的位置, 返回值为截取的字符串, 支持负数。 如果不指定结束位置, 都会截取到最后。

- `str.substring(1,-1)`

- 从开始的位置截取到结束的位置，返回值为截取的字符串，不支持负数。

- `str.substr(m,n)`

- 第一个值表示：下标。第二个值：截取的数量。若不指定长度，都会截取到最后。

- `str.split(2,3)`

- 第一个参数表示分割的字符，第二个表示数组的长度，返回数组，不会对原字符串有影响，可以不接受参数。

- `str.toUpperCase()`

- 不接受参数，转为大写。

- `str.toLowerCase()`

- 不接受参数，转为小写。

数学对象（Math 内置对象）

- `Math.random()`

- 1. 随机数(0—1的随机数，取不到0和1)

- `Math.ceil()`

- `Math.floor()`

- `Math.round(a)`

- `Math.abs(a)`

- `Math.min(a,b,c)`

- `Math.max(a,b,c)`

- `Math.sqrt(9)`

- `Math.pow(2,3)`

- `Math.sin()`

- `Math.cos()`

- `Math.PI`

- `toFixed(2)` 保留小数点后的几位