

常用 JavaScript 字符串方法简述

indexOf(str) 返回字符串中参数字串第一次出现的位置（从左到右搜索,且下标从 0 开始）。如果没有匹配项，返回 -1。

```
var str="zhufengpeixun";
var index1 = str.indexOf('u'); //结果是 2
var index2 = str.indexOf('u',5); //从索引位置 5 开始，往右查找，结果是 11
var index3=str.indexOf('d');//结果是-1，找不到字符串 d
var index4=str.indexOf('pei',5)//结果是 7
注:indexof() 对大小写敏感。
```

charAt(index)

返回指定位置的字符。

```
var str=" zhufengpeixun" ;
var result = str.charAt(0); // result 的结果是 "z"
```

lastIndexOf(str)

返回字符串中参数字串最后一处出现的索引（从右到左搜索），如果没有匹配项，返回 -1。

```
var str="zhufengpeixun";
var index1 = str.lastIndexOf('u'); //结果是 11
var index2 = str.lastIndexOf('u',10); //从索引位置 10 开始，往左查找，结果是 2
```

substring(start,[end]) 查找指定索引位置的字符串，并将其返回。

返回从第一个参数的位置开始，到第二个参数位置的之前的字符串。第一个参数必填，第二个参数可以不写。如果第二个参数，表示结束的位置是最后。

```
var str="zhufengpeixun";
var result1=str.substring(7); //result1 结果是 "peixun"
var result2=str.substring(7,10); //result2 结果是 "pei"
```

substr(begin,[length])

返回字符串的一个子串，传入参数是起始位置和长度。第一个参数必写。第二个参数不写，则表示从结束位置到最后（和 substring 方法相同）。

```
var str="zhufengpeixun";
var result1=str.substr(7); //result1 结果是 "peixun"
var result2=str.substr(7,3); //第二个参数 3 是长度，result2 结果是 "pei"
```

toUpperCase 将整个字符串转成大写字母。

```
var result="zhufengpeixun".toUpperCase(); //结果是 ZHUFENGPEIXUN
```

toLowerCase 将整个字符串转成小写字母。

```
var result="CHINA".toLowerCase();//结果是 china
```

replace 字符串查找并替换的方法，功能很强大。在正则教材中会详解。

```
var str="a b c d e f g";//字符中间有空格
var result=str.replace(/\s+/g,'');//结果是 abcdefg, 把空格去掉了
```

search(reg) 用来查找字符的位置，和 `indexOf` 很像，它比 `indexOf` 的优点在于参数可以是一个正则表达式。

```
var str="珠峰培训成立于 2009 年 5 月 7 日";
var result=str.search(/\d+/);//结果是 7
```

split(separator,[limit]) 用指定的字符或正则以分隔符，把一个字符串分隔，结果作为一个数组返回。如果没有指定分隔符，则默认以逗号,为分隔符。

```
var str="2013-11-21 21:46:36";
var a1=str.split(/[- :]/);//a1 结果是
//['2013','11','21','21','46','36']
var a2=str.split(/[- :]/,3);//a2 的结果为['2013','11','21'],第二个参数用来指定返回值的长度
```

match(reg)

此方法功能强大，在正则中会详细介绍。它以正则参数，把正则匹配到的内容保存到数组里返回，如果没有匹配返回 `null`。

```
var reg = /^[a-z]+$;/;
var str1='zhufengpeixun';
var str2='0123456789'
var result1 = str1.match(reg); // 结果是一个类数组，保存着 "zhufengpeixun"
var result2 =str2.match(reg); //结果是 null
```

concat 将两个或多个字符的文本组合起来，返回一个新的字符串。用的不多。

```
var a = "hello";
var b = ",world";
var c=" ,zhufengpeixun"
var result = a.concat(b,c);
alert(result); //弹出 hello,word,zhuengpeixun
这个相当于 result=a+b+c;
```

slice 功能与 `substring` 相同，用的不多。

charCodeAt(index) 指定索引位置的字符的 Unicode 码

```
var str='abcd';
alert(str.charCodeAt(0))//弹出 97, 是 a 对应的 ASCII 码表中十进制的编码,也叫 Unicode 码
```