

shift:删除原数组第一项，并返回删除元素的值；如果数组为空则返回undefined

```
a.shift();
```

unshift:将参数添加到原数组开头，并返回数组的长度

```
a.unshift(-2,-1);
```

注:在IE6.0下测试返回值总为undefined，FF2.0下测试返回值为7，所以这个方法的返回值不可靠，需要用返回值时可用splice代替本方法来使用。

pop:删除原数组最后一项，并返回删除元素的值；如果数组为空则返回undefined

```
a.pop();
```

push:将参数添加到原数组末尾，并返回数组的长度

```
a.push(6,7);
```

concat:返回一个新数组，是将参数添加到原数组中构成的

```
a.concat(6,7); //a:[1,2,3,4,5] b:[1,2,3,4,5,6,7]
```

splice(start,deleteCount,val1,val2,...):从start位置开始删除deleteCount项，并从该位置起插入val1,val2,...

Javascript代码

```
var a = [1,2,3,4,5];
```

```
var b = a.splice(2,2,7,8,9); //a:[1,2,7,8,9,5] b:[3,4]
```

```
var b = a.splice(0,1); //同shift
```

```
var b = a.splice(a.length-1,1); //同pop
```

```
a.splice(a.length,0,6,7); var b = a.length; //同push
```

reverse:将数组反序

```
a.reverse(); //a:[5,4,3,2,1] b:[5,4,3,2,1]
```

sort(orderfunction):按指定的参数对数组进行排序

```
b = a.sort(); //a:[1,2,3,4,5] b:[1,2,3,4,5]
```

slice(start,end):返回从原数组中指定开始下标到结束下标之间的项组成的新数组

```
b = a.slice(2,5); //a:[1,2,3,4,5] b:[3,4,5]
```

`join(separator)`:将数组的元素组起一个字符串，以`separator`为分隔符，省略的话则用默认用逗号为分隔符

```
var a = [1,2,3,4,5];
var b = a.join("|"); //a:[1,2,3,4,5] b:"1|2|3|4|5"
```

数组是JavaScript提供的一个内部对象，它是一个标准的集合，我们可以添加(`push`)、删除(`shift`)里面元素，我们还可以通过`for`循环遍历里面的元素，那么除了数组我们在JavaScript里还可以有别的集合吗？

由于JavaScript的语言特性，我们可以向通用对象动态添加和删除属性。所以Object也可以看成是JS的一种特殊的集合。下面比较一下Array和Object的特性：

Javascript代码

```
//Array:
/*新建:*/var ary = new Array(); 或 var ary = [];
/*增加:*/ary.push(value);
/*删除:*/delete ary[n];
/*遍历:*/for ( var i=0 ; i < ary.length ; ++i ) ary[i];

//Object:
/*新建:*/var obj = new Object(); 或 var obj = {};
/*增加:*/obj[key] = value; (key为string)
/*删除:*/delete obj[key];
/*遍历:*/for ( var key in obj ) obj[key];
```

从上面的比较可以看出Object完全可以作为一个集合来使用，在使用Popup窗口创建无限级Web页菜单(3)中我介绍过Eric实现的那个__MenuCache__，它也就是一个模拟的集合对象。

如果我们要在Array中检索出一个指定的值，我们需要遍历整个数组：

```
var keyword = ;
1. for ( var i=0 ; i < ary.length ; ++i )
    {
    }
// todo
{
if ( ary[i] == keyword )
```

```
{  
for ( var i=0 ; i < ary.length ; ++i )
```

而我们在Object中检索一个指定的key的条目，只需要是要使用：

```
var key = '';  
var value = obj[key];  
// todo
```