**十二 、javascript-ECMA5**

**一、严格模式**

严格模式，使用该模式以后，语法规范会非常严格。

确定在哪个范围内使用严格模式。

【注】对应的作用域： "use strict"，设置以后，在该作用域内必须使用严格模式

function func(){

max = 10; //如果一个变量在声明的时候，没有使用var声明，那么这个变量会被声明称全局变量。

}

func();

alert(max);//10

**二、bind**

**this指向问题**

function show(){

alert(this); //object Window

}

setTimeout(function(){

show.call(document);

}, 2000);//调用函数，强制this指向为document

setTimeout(show.bind("hello world"), 2000);在调用函数之前，强制this指向为hello world，与call区别

**三、数组的额外方法**

数组新增函数：

**1.indexOf**

【格式】数组.indexOf(元素, start);

alert([1, 2, 3, 4, 5, 4].indexOf(4, 4));

//和字符串的应用一样，第一个参数用来查找元素第一次出现的位置，第二个参数为开始查找的下标。

**2.数组的遍历forEach**

for循环

for...in**//i为字符串**

forEach

【格式】数组.forEach(函数);

var arr = [10, 5, 3, 2, 6, 7];

arr.forEach(function(item, index, array){

/\*

item 每一次遍历到的元素

index 每一次遍历到的元素的下标

array 数组

\*/

//alert(item + ", " + index + ", " + array);

});

**3.map 映射**

var arr = [10, 5, 3, 2, 6, 7];

var newArr = arr.map(function(item, index, array){

return item + 10;

})

alert(newArr); ////20,15,13,12,16,17

**4.reduce 归并**

var tmp = [1, 2, 3, 4].reduce(function(pre, next, index, array){

/\*

next 下标1遍历

pre 每一次循环，上一次return后面表达式结果

\*/

//alert(pre + ", " + next);

return pre + next;

})

//alert(tmp);

**5.filter**

【注】挑出数组中符合条件的元素

return 部分是筛选的条件

var res = [50, 60, 30 , 70, 80, 90].filter(function(item, index, array){

return item > 50;

})

alert(res);

**6.some**

【返回值】true或者false。

【功能】判断return后面的表达式是否成立。

【注】如果判断条件成立，循环就停止了。

var tmp = ["One", "Two", "Three"].some(function(item, index, array){

alert(item);

return item == "Two";

})

alert(tmp);

**7.every**

【注】判断每一个条件都符合才返回true。

【注】只要有一个条件不符合条件，循环就停止了。

var tmp = [10, 20, 30, 40].every(function(item, index, array){

alert(item);

return item > 10;

})

alert(tmp);

**四、trim**

【功能】删除字符串，首尾空格。

【格式】字符串对象.trim();