**十、javascript事件绑定及深入**

**一、传统事件绑定的问题**

**1.【问题1】后面添加的事件会把前面的函数覆盖掉。**

window.onload = function(){

alert("Lee");

}

window.onload = function(){

alert("Mr Lee");

}

//结果为Mr Lee，Lee被覆盖掉了

【解决方法】想要同时显示的写法：将前一个事件赋值给一个变量



**2.【问题2】this指向的问题**



此时this显示结果为[object Window]，并不指向oDiv。

【解决方法】call 强制改变this的指向

【格式】函数.call(**第一个参数**this的指向);



**3.【问题3】too much recursion太多的递归**

【解决方法】





**4.【问题4】重复添加的同一个函数并不会进行过滤。**

【解决方法】



**二、传统事件绑定问题的解决（事件监听器）**

传统事件绑定：

1、覆盖

2、this

3、太多的递归

4、不能去重

DOM：

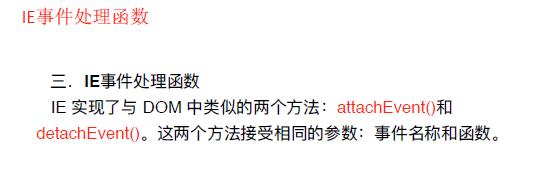
事件监听器：（有兼容问题）

1、addEventListener()

2、removeEventListener()



【添加事件，跨浏览器的兼容写法】





**三、阻止事件默认行为**

传统事件阻止默认行为

var oA = document.getElementById("a\_id");

oA.onclick = function(){

if(true){

return 10;

}

return false;

}

【缺点】：

1、return false必须写在最后，阻止事件默认行为以后，就不能继续操作了

2、前面的代码有可能导致执行不到return false;

**【解决】preventDefault**

