# **[javascript逻辑运算符“||”和“&&”](http://www.cnblogs.com/pigtail/archive/2012/03/09/2387486.html)**

1. 先来说说||（逻辑或）,从字面上来说，只有前后都是false的时候才返回false，否则返回true。

alert(true||false); // true  
alert(false||true); // true  
alert(true||true); // true  
alert(false||false); // false

1. 但是，从深层意义上来说的话，却有另一番天地，试下面代码

alert(0||1);

显然，我们知道，前面0意味着false,而后面1意味着true,那么上面的结果应该是true,而事实返回的结果是1。再看下面代码：

alert(2||1);

我们知道，前面2是true,后面1也是true,那返回结果又是什么呢？测试结果是2,继续看：

alert('a'||1);

同样，前面'a'是true，后面1也是true;测试结果是'a',下面

alert(''||1);

由上，我们知道前面''是false,后面1是true,而返回结果是1。再看下面

alert('a'||0);

 前面'a'是true，而后面0是false,返回结果是'a'，继续下面

alert('a'||'b');

 前面'a'是true,后面'b'是false,返回结果是'a'，我们继续下面

alert(''||0);

 前面''是false,后面0同样是false,返回结果是0

alert(0||'');

前面0是false，后面''是false，返回结果是''

这就意味

1、只要“||”前面为false,不管“||”后面是true还是false，都返回“||”后面的值。

2、只要“||”前面为true,不管“||”后面是true还是false，都返回“||”前面的值。  
二、下面说说&&（逻辑与）,从字面上来说，只有前后都是true的时候才返回true，否则返回false。

alert(true&&false); // false  
alert(true&&true); // true  
alert(false&&false); // false  
alert(false&&true); // false

然后，根据上面经验，我们看看“&&”号前后，不单单是布尔类型的情况。

alert(''&&1);

结是返回'',“&&”前面''是false,后面是1是true。

alert(''&&0);

结是返回'',“&&”前面''是false,后面是0也是false。

alert('a'&&1);

结是返回1,“&&”前面''a是true,后面是1也是true。

alert('a'&&0);

结是返回0,“&&”前面''a是true,后面是0是false。

alert('a'&&'');

结是返回'',“&&”前面''a是true,后面是''是false。

alert(0&&'a');

结是返回0,“&&”前面''0是false,后面是'a'是true。

alert(0&&'');

结是返回0,“&&”前面''0是false,后面是''也是false。

这意味着

1、只要“&&”前面是false，无论“&&”后面是true还是false，结果都将返“&&”前面的值;

2、只要“&&”前面是true，无论“&&”后面是true还是false，结果都将返“&&”后面的值;

让我们总结一下：

1、只要“||”前面为false，无论“||”后面是true还是false，结果都返回“||”后面的值。

2、只要“||”前面为true，无论“||”后面是true还是false，结果都返回“||”前面的值。

3、只要“&&”前面是false，无论“&&”后面是true还是false，结果都将返“&&”前面的值;

4、只要“&&”前面是true，无论“&&”后面是true还是false，结果都将返“&&”后面的值;

由上两个测试可知，逻辑运算符，“||”和“&&”都是遵行短路原则，只要确定符号前面的真假，既可确定返回值。

需要说明的是“&&”的优先级是高于“||”的，下面测试：

alert(1||'a'&&2);

返回结果是1。

根据反证原理，我们假设“||”的优先级不低于“&&”（这里只所以用“不低于”，是为了同时证明等级相同的情况）。

根据上面我们得出的结论（1），（1||'a'）将返回前面的值1，(1&&2)根据结论(4)应该返回后面的值2。这显然不对，由此可知“&&”的优先级是高于“||”的。