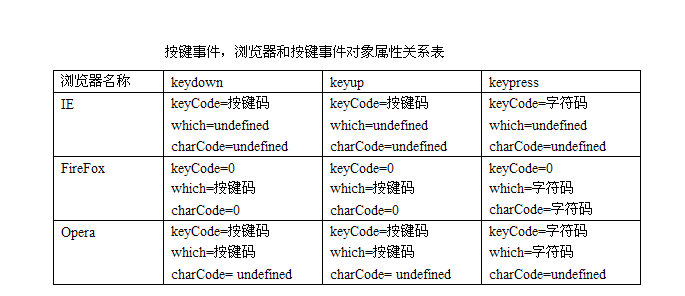
**3.1 按键事件的按键码和字符码**

按键事件的按键码和字符码缺乏浏览器间的可移植性，对于不同的浏览器和不同的案件事件，按键码和字符码的存储方式都是不同的，按键事件，浏览器和按键事件对象属性关系如下表：



如表所示：

在IE中，只有一个keyCode属性，并且它的解释取决于事件类型。对于keydown来说，keyCode存储的是 按键码，对于 keypress事件来说，keyCode存储的是一个字符码。而IE中没有which和charCode属性，所以which和charCode属性始 终为undefined。

FireFox中keyCode始终为0，时间keydown/keyup时，charCode=0，which为按键码。事件keypress时，which和charCode二者的值相同，存储了字符码。

在Opera中，keyCode和which二者的值始终相同，在keydown/keyup事件中，它们存储按键码，在keypress时间中，它们存储字符码，而charCode没有定义，始终是undefined。

**3.2 用keydown/keyup还是keypress**

第一部分已经介绍了keydown/keyup和keypress的区别，有一条比较通用的规则，keydown事件对于功能按键来说是最有用的，而keypress事件对于可打印按键来说是最有用的[3]。

键盘记录主要是针对于可打印字符和部分功能按键，所以keypress是首选，然而正如第一部分提到的，IE中keypress不支持功能按键，所以应该用keydown/keyup事件来进行补充。

8：退格键  
46：delete  
37-40： 方向键  
48-57：小键盘区的数字  
96-105：主键盘区的数字  
110、190：小键盘区和主键盘区的小数点

189、109：小键盘区和主键盘区的负号

