第 **8** 章 基础事件

学习要点：

1

2

3

.绑定事件

.简写事件

.复合事件

JavaScript 有一个非常重要的功能，就是事件驱动。当页面完全加载后，用户通过鼠标

或键盘触发页面中绑定事件的元素即可触发。jQuery 为开发者更有效率的编写事件行为，封

装了大量有益的事件方法供我们使用。

一．绑定事件

在 JavaScript 课程的学习中，我们掌握了很多使用的事件，常用的事件有：click、dblclick、

mousedown、mouseup、mousemove、mouseover、mouseout、change、select、submit、keydown、

keypress、keyup、blur、focus、load、resize、scroll、error。那么，还有更多的事件可以参考

手册中的事件部分。

jQuery 通过.bind()方法来为元素绑定这些事件。可以传递三个参数：bind(type, [data], fn)，

type 表示一个或多个类型的事件名字符串；[data]是可选的，作为 event.data 属性值传递一个

额外的数据，这个数据是一个字符串、一个数字、一个数组或一个对象；fn 表示绑定到指

定元素的处理函数。

//使用点击事件

$

('input').bind('click', function () {

//点击按钮后执行匿名函数

alert('点击！');

});

/

/普通处理函数

('input').bind('click', fn);

function fn() {

alert('点击！');

$

//执行普通函数式无须圆括号

}

//可以同时绑定多个事件

$

('input').bind('mouseout mouseover', function () {

//移入和移出分别执行一次

$

('div').html(function (index, value) {

return value + '1';

});

});

//通过对象键值对绑定多个参数

$('input').bind({

//传递一个对象

'mouseout' : function () {

//事件名的引号可以省略

alert('移出');

},

'mouseover' : function () {

alert('移入');

}

});

//使用 unbind 删除绑定的事件

$

('input').unbind();

//删除所有当前元素的事件

//删除当前元素的 click 事件

//使用 unbind 参数删除指定类型事件

$

('input').unbind('click');

/

/使用 unbind 参数删除指定处理函数的事件

function fn1() {

alert('点击 1');

}

function fn2() {

alert('点击 2');

}

$

$

('input').bind('click', fn1);

('input').bind('click', fn2);

$

('input').unbind('click', fn1);

//只删除 fn1 处理函数的事件

二．简写事件

为了使开发者更加方便的绑定事件，jQuery 封装了常用的事件以便节约更多的代码。我

们称它为简写事件。

简写事件绑定方法

方法名

click(fn)

触发条件

鼠标

描述

触发每一个匹配元素的 click(单击)事件

dblclick(fn)

mousedown(fn)

mouseup(fn)

mouseover(fn)

mouseout(fn)

鼠标

触发每一个匹配元素的 dblclick(双击)事件

触发每一个匹配元素的 mousedown(点击后)事件

触发每一个匹配元素的 mouseup(点击弹起)事件

触发每一个匹配元素的 mouseover(鼠标移入)事件

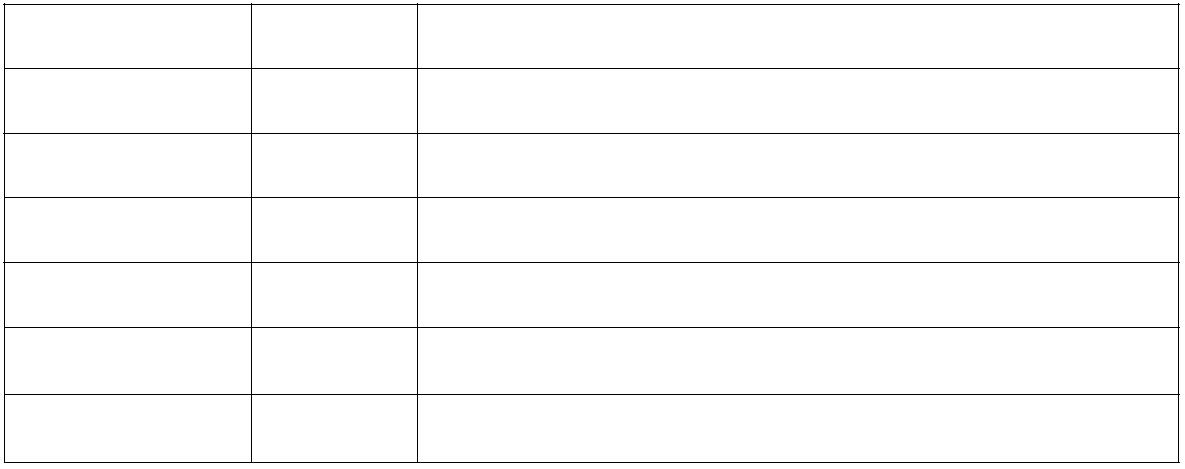
触发每一个匹配元素的 mouseout(鼠标移出)事件

鼠标

鼠标

鼠标

鼠标



mousemove(fn)

mouseenter(fn)

mouseleave(fn)

keydown(fn)

keyup(fn)

鼠标

鼠标

鼠标

键盘

键盘

键盘

文档

文档

文档

表单

表单

表单

表单

表单

表单

表单

触发每一个匹配元素的 mousemove(鼠标移动)事件

触发每一个匹配元素的 mouseenter(鼠标穿过)事件

触发每一个匹配元素的 mouseleave(鼠标穿出)事件

触发每一个匹配元素的 keydown(键盘按下)事件

触发每一个匹配元素的 keyup(键盘按下弹起)事件

触发每一个匹配元素的 keypress(键盘按下)事件

当卸载本页面时绑定一个要执行的函数

keypress(fn)

unload(fn)

resize(fn)

触发每一个匹配元素的 resize(文档改变大小)事件

触发每一个匹配元素的 scroll(滚动条拖动)事件

触发每一个匹配元素的 focus(焦点激活)事件

触发每一个匹配元素的 blur(焦点丢失)事件

触发每一个匹配元素的 focusin(焦点激活)事件

触发每一个匹配元素的 focusout(焦点丢失)事件

触发每一个匹配元素的 select(文本选定)事件

触发每一个匹配元素的 change(值改变)事件

触发每一个匹配元素的 submit(表单提交)事件

scroll(fn)

focus(fn)

blur(fn)

focusin(fn)

focusout(fn)

select(fn)

change(fn)

submit(fn)

注意：这里封装的大部分方法都比较好理解，我们没必要一一演示确认，重点看几个需

要注意区分的简写方法。

.

mouseover()和.mouseout()表示鼠标移入和移出的时候触发。那么 jQuery 还封装了另外

一组：.mouseenter()和.mouseleave()表示鼠标穿过和穿出的时候触发。那么这两组本质上有

什么区别呢？手册上的说明是：.mouseenter()和.mouseleave()这组穿过子元素不会触发，

而.mouseover()和.mouseout()则会触发。

//HTML 页面设置

<div style="width:200px;height:200px;background:green;">

<p style="width:100px;height:100px;background:red;"></p>

</div>

<strong></strong>

//mouseover 移入

$

('div').mouseover(function () {

//移入 div 会触发，移入 p 再触发

$

('strong').html(function (index, value) {

return value+'1';

});

});