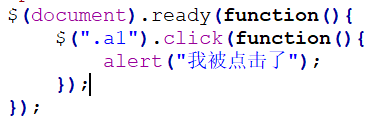
# JQuery事件

1）在JQuery中，对于大多数的 DOM事件都提供了等效的JQuery方法。如下：

|  |  |  |  |
| --- | --- | --- | --- |
| click( ) | 单击事件 | mousedown ( ) | 鼠标键按下事件 |
| dblclick( ) | 双击事件 | mouse move ( ) | 鼠标键松开事件 |
| change( ) | 状态改变事件 | mouse leave ( ) | 鼠标离开事件 |
| blur( ) | 失去焦点事件 | submit ( ) | 表单提交事件 |
| focus( ) | 获取焦点事件 | load ( ) | 元素加载完成事件 |
| keydown( ) | 键盘被按事件 |  |  |
| keyup () | 键盘松开事件 |  |  |
| mouse enter () | 鼠标进入事件 |  |  |

以上只是列出了部分常用的事件。

2）使用这是JQuery事件方法的语法如下：



在这里表示为class属性包含了a1的元素添加了单击事件。其中，click( )的参数为

一个函数，这个函数是该事件的处理程序。**当你没有为事件方法传入事件处理程序，则表示使用代码的方式来主动触发事件。**

# 常用的事件

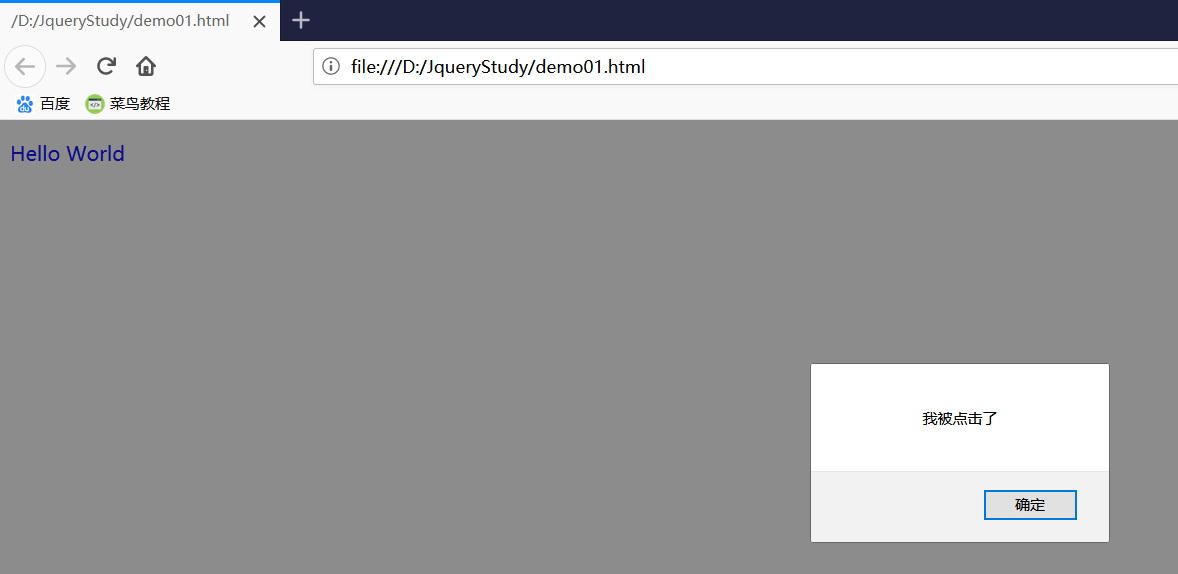
## 鼠标事件

### 单击事件 click( )

元素绑定了单击事件click( )后，点击该元素时执行该事件的处理程序。如下：

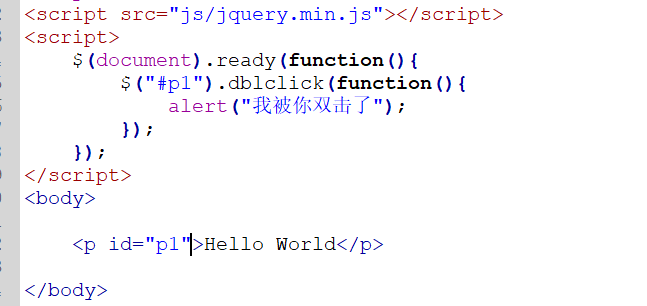


单击该元素时，执行结果如下：

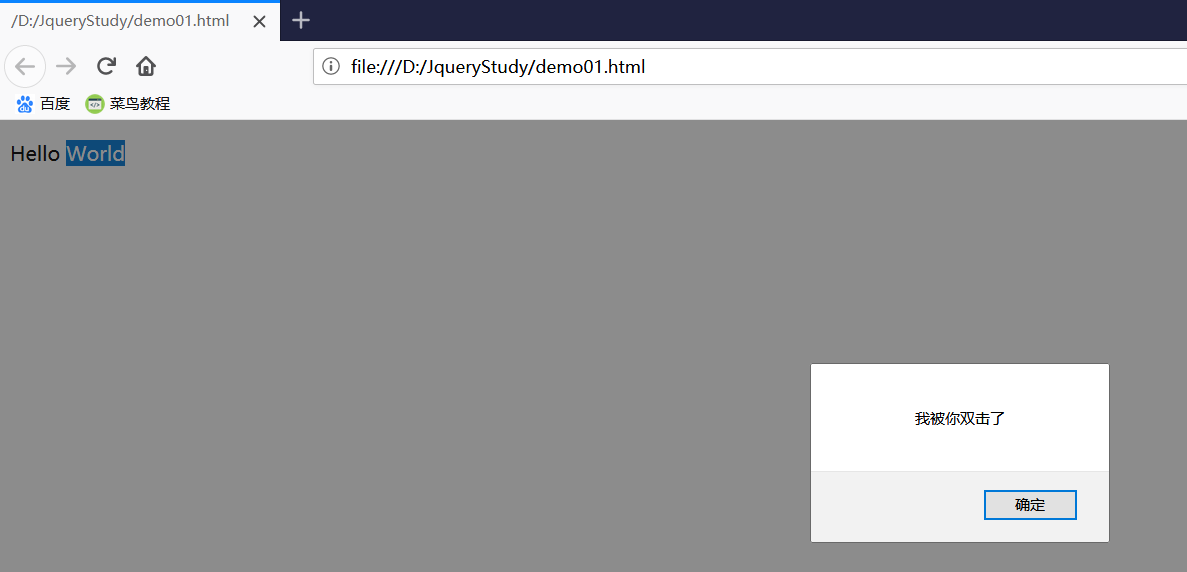


### 双击事件 dblclick( )

元素绑定了双击事件dblclick( )后，双击该元素时执行该事件的处理程序。如下：



双击该元素时，执行结果如下：

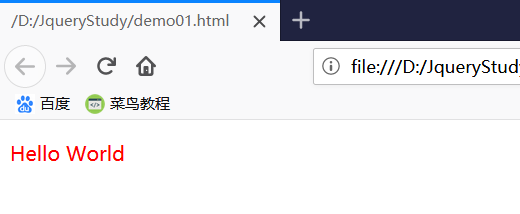


### 鼠标进入事件 mouseenter( )

元素绑定鼠标进入事件mouseenter( )后，将鼠标移动到该元素上时执行该事件的处理程序。如下：

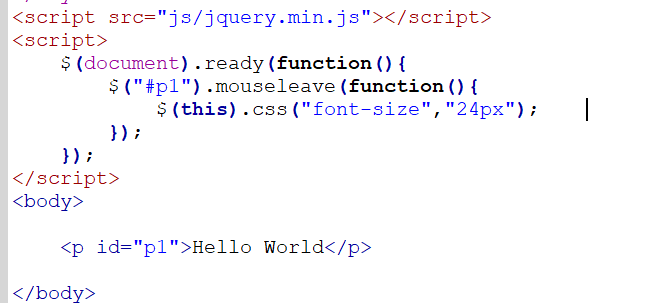


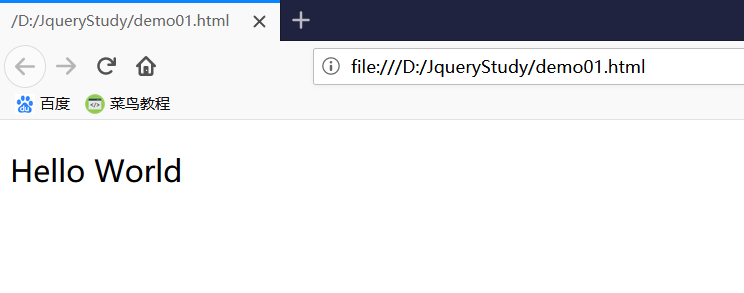
将鼠标指针移动到该元素上时，执行结果如下：



### 鼠标离开事件 mouseleave( )

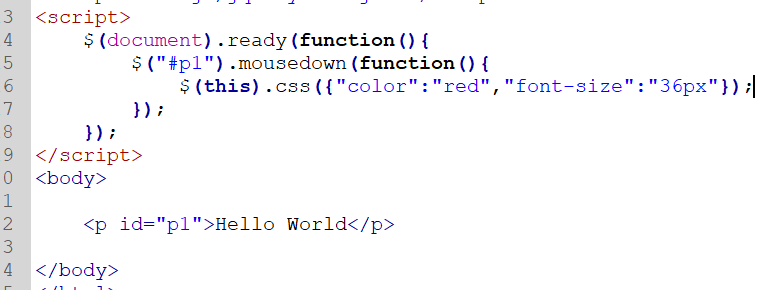
元素绑定鼠标离开事件mouseleave( )后，当鼠标移动到该元素上，将鼠标离开后，执行该事件的处理程序。如下：



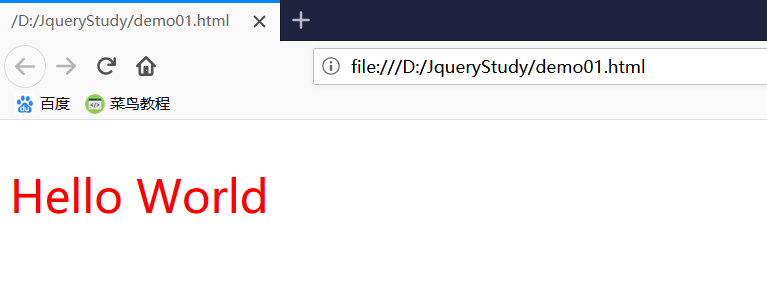
 当鼠标进行该元素上并离开后，运行结果如下：

### 鼠标按下事件 mousedown( )

元素绑定鼠标按下事件mousedown( )后，当鼠标移动到该元素上并按下鼠标任意键时执行该事件的处理程序。如下：

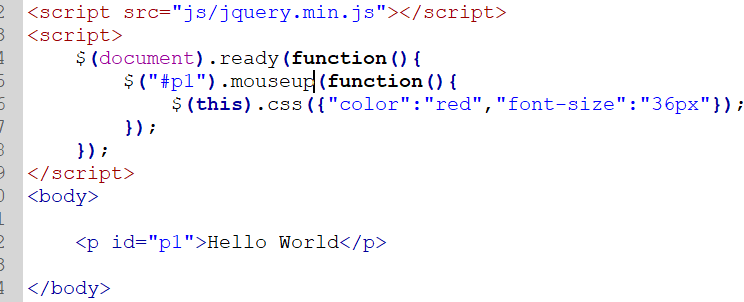


当鼠标移动到该元素上并按下了鼠标的任意键时，执行结果如下：

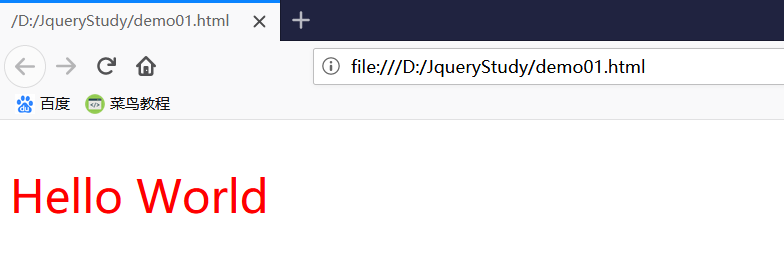


### 鼠标松开事件 mouseup( )

元素绑定鼠标松开事件mouseup( )后，当鼠标移动到该元素并按下鼠标任意键并松开后，执行该事件绑定的处理程序。



当鼠标移动到该元素并按下鼠标任意键并松开后，执行结果如下：



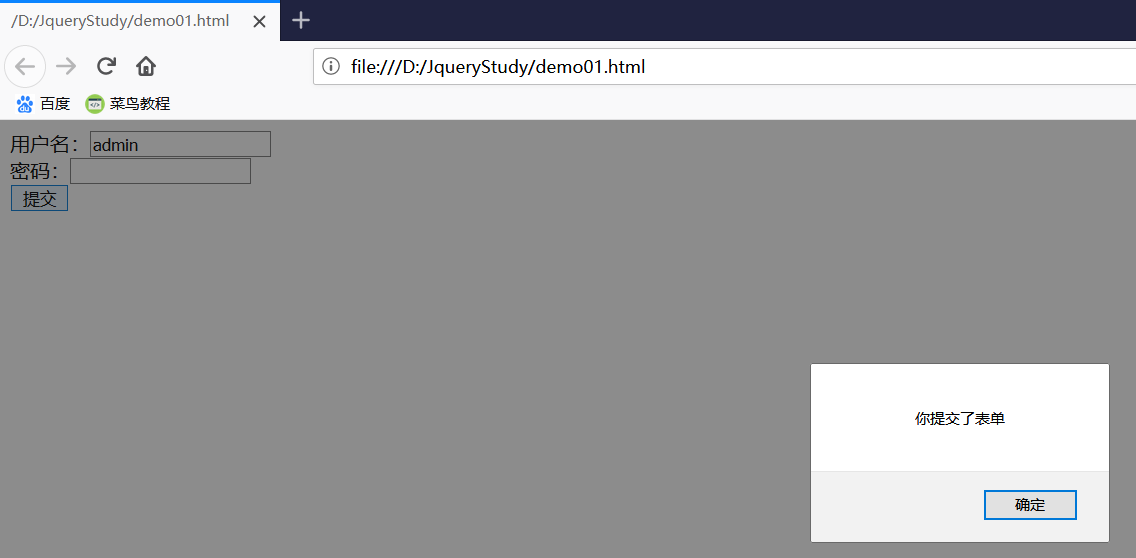
## 表单事件

### 表单提交事件 submit( )

为表单元素（一般情况下是<form>标签）绑定表单提交事件后，当点击该表单的提交 按钮时，执行该事件的处理程序。如下：



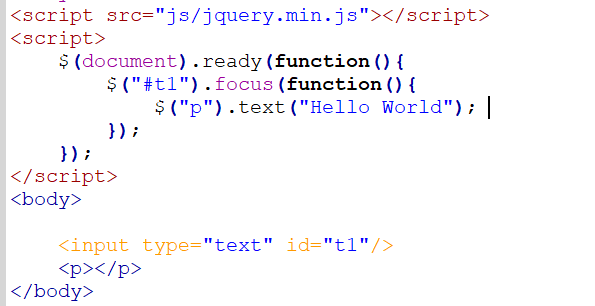
当提交表单后，执行结果如下：



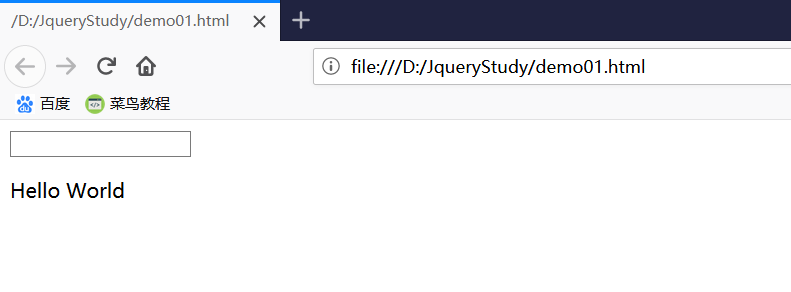
通过表单提交事件，你可以在事件处理函数中对表单中的数据进行验证，验证是否是合法的数据。

### 获取焦点事件 focus( )

表单元素（通常是单行文本框和文本域）绑定focus( )事件后，当该元素得到焦点后，执行该事件绑定的事件处理程序。如下：

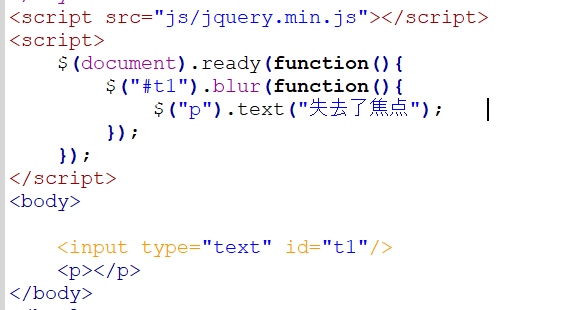


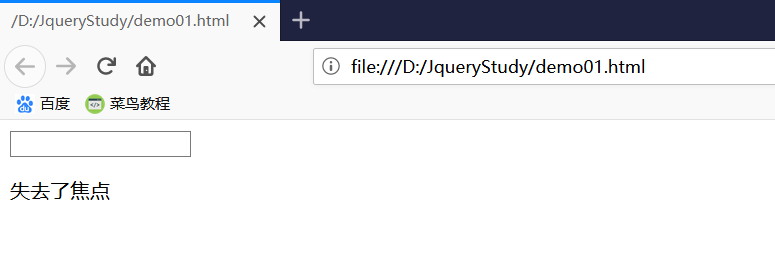
当该元素获得焦点后，执行结果如下：



### 失去焦点事件 blur( )

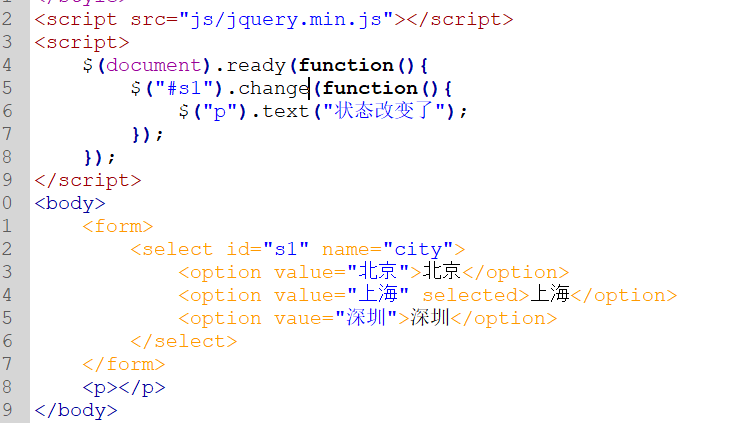
表单元素（通常是单行文本框或文本域）绑定失去焦点事件blur( )后，当该表单元素在获取焦点并失去焦点后，执行该事件绑定的事件处理程序。如下：



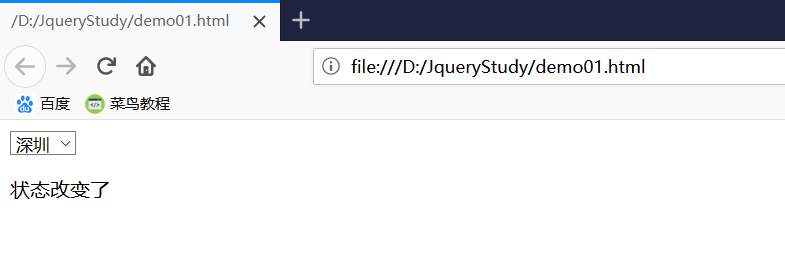
当该表单元素在获取焦点并失去焦点后，执行结果如下：

### 状态改变事件 change( )

表单元素（通常是选择性表单元素）绑定状态改变事件change( )后，当选择的选项改变后，执行该事件绑定的事件处理程序。如下：



当你在选择选项时，触发了该元素的状态改变事件。运行结果如下：

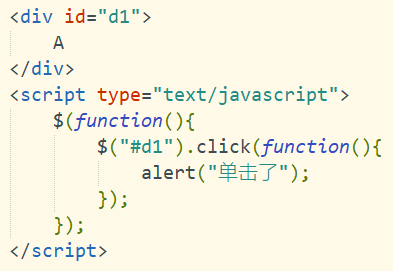


# 事件的绑定方式

为元素绑定事件的方式有两种：一是直接使用事件名函数绑定，二是使用bind( )函数定。

## 使用事件名函数绑定

使用事件名函数为元素绑定事件只能一次只能绑定单个事件，如下，为元素绑定单击事件：



## 使用bind( )函数绑定

使用bind( )函数为元素绑定事件分为两种情况，一种是为元素绑定单个事件（作用与使用事件名函数绑定相同），另外一种是为元素绑定多个事件。

**绑定单个事件**

使用bind为元素绑定单个事件的语法如下：



它的作用与使用事件名为元素绑定事件相同。在实际开发中，如果为元素只绑定一个事件，我们一般使用事件名函数进行绑定，而不是使用bind( )函数。

**绑定多个事件**

bind( )函数最大的作用就是可以同时为元素绑定多个事件。如下：



## 两者的区别

使用事件名或bind( )函数为元素绑定事件的最大区别在于：bind( )函数可以为元素绑定多个事件，而事件名的方式只能一次给元素绑定一个事件。其语法差异如下：



# 解除绑定事件

