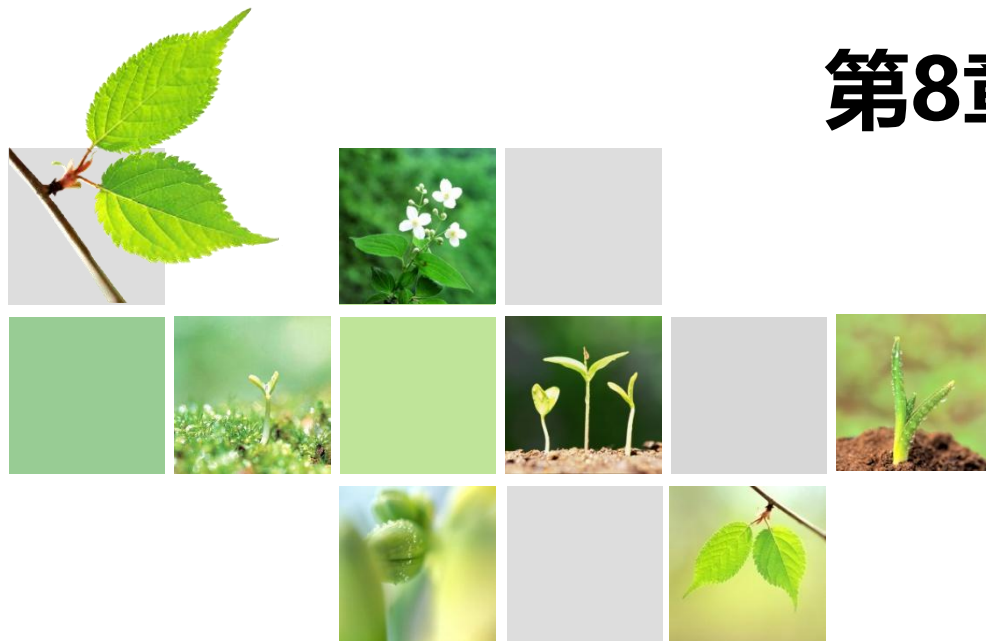




第8章 jQuery中的事件



回顾



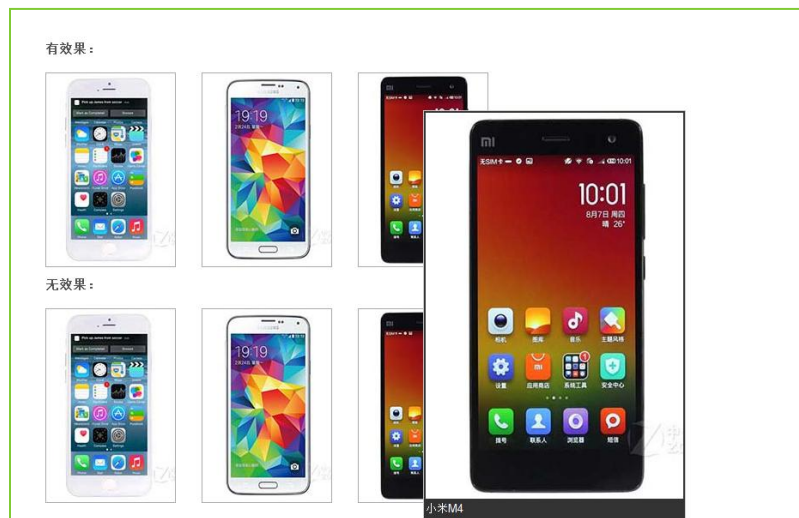
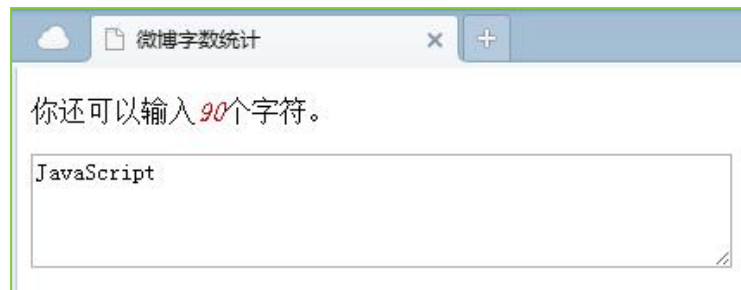
- 在jQuery中使用什么方法获取和设置属性？
- `$("div"). append("<a>DOM操作")`表示什么含义？
- `$("ul li:first"). detach()`与`$("ul li:first"). empty()`有何区别？



本章任务



- 任务1 微博字数统计
- 任务2 制作图片提示效果
- 任务3 改变图片透明度



学习目标



1. 学会绑定事件
2. 会使用简写事件
3. 会使用复合事件
4. 掌握事件对象
5. 了解冒泡和默认行为

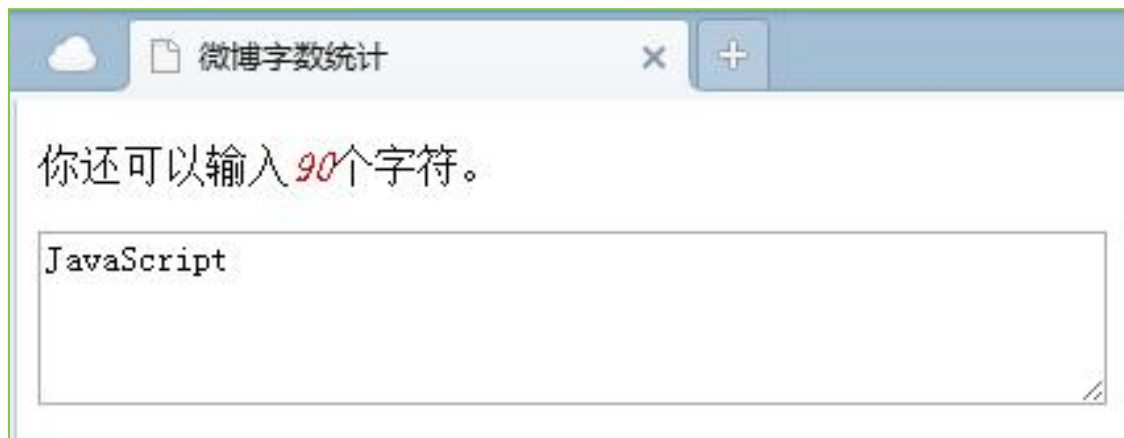


任务1 微博字数统计



- 任务描述：

- 在页面中，允许用户在文本域输入100个字符，当用户每输入一个字符，提示文字则显示还可以输入多少个字符，如图所示，当用户输入的字符长度超过100时，提示用户输入的字符超长了，并使文本域不可用。



任务1 微博字数统计



- 任务分析：

- 1. 完成静态页面设计，标识文本域和提示部分。
- 2. 加载DOM后添加处理函数。
- 3. 按键弹起事件触发时，统计文本域中字符长度。
- 4. 在提示字符数处显示还可以输入（100-文本域长度）个字符，此处添加判断，若用户输入长度大于100时，提示用户输入的字符超长了，并通过attr方法设置文本域为只读。



jQuery 中的事件--加载 DOM



- 在页面加载完毕后, 浏览器会通过 JavaScript 为 DOM 元素添加事件。在常规的 JavaScript 代码中, 通常使用 `window.onload` 方法, 在 jQuery 中使用 `$(document).ready()` 方法。

方法	<code>window.onload</code>	<code>\$(document).ready()</code>
执行时机	必须等待网页中的所有内容加载完毕后(包括图片)才能执行	网页中的所有 DOM 结构绘制完毕后就执行, 可能 DOM 元素关联的东西并没有加载完
编写个数	不能同时编写多个	能同时编写多个
简化写法	无	<code>\$()</code>



事件绑定



事件绑定的语法:

`$(selector).bind(type, [data],function)`

- **type**为事件类型

包括:

blur、focus、load、resize、scroll、unload、click、dblclick、mousedown、mouseup、mousemove、mouseover、mouseout、mouseenter、mouseleave、change、select、submit、keydown、keypress、keyup、error等。

- **data**为方法传递的参数,可以忽略.

- **function**是用来绑定的处理函数.

例如: 为id为hot 的div的元素绑定单击事件。

```
$("#hot").bind("click",function()
```

```
{
```

```
    alert("层被点击");
```

```
}
```



事件移除



有时候事件执行完了，想取消事件的效果可以通过一定的办法来处理。

事件移除的语法：

`$(selector).unbind(type, [data],function)`

例如：移除id为mydiv对象上的mouseout事件

`$("#mydiv").unbind("mouseout");`



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。称它为简写事件。

简写事件绑定方法

方法名	触发条件	描述
click(fn)	鼠标	触发每一个匹配元素的 click(单击)事件
dblclick(fn)	鼠标	触发每一个匹配元素的 dblclick(双击)事件
mousedown(fn)	鼠标	触发每一个匹配元素的 mousedown(点击后)事件
mouseup(fn)	鼠标	触发每一个匹配元素的 mouseup(点击弹起)事件
mouseover(fn)	鼠标	触发每一个匹配元素的 mouseover(鼠标移入)事件
mouseout(fn)	鼠标	触发每一个匹配元素的 mouseout(鼠标移出)事件





- `.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？
- `.mouseenter()`和`.mouseleave()`这组穿过子元素不会触发，而`.mouseover()`和`.mouseout()`则会触发。



示例



- 当鼠标滑过“标题”时，参考书信息显示，如图1所示。
当鼠标滑出“标题”时，参考书信息隐藏，如图2所示。

用户交互设计学习参考书

- JavaScript DOM编程
- 锋利的JQuery
- JQuery入门与提高
- JavaScript高级编程
- JQuery权威指南
- JQuery实战

用户交互设计学习参考书





```
$(document).ready(function(e) {  
    $("h2").mouseover(function(){  
        $(this).next().hide();  
    });  
    $("h2").mouseout(function(){  
        $(this).next().show();  
    });  
});
```



任务实现



- 关键代码：

```
$(document).ready(function(){
    var text=$("#edit_myfrom textarea").val(); //获取文本域内容
    var counter=text.length; //获取文本域内容长度
    $("#numtj var").text(100-counter); //提示还有多少
    $(document).keyup(function() { //按键弹起时触发
        var text=$("#edit_myfrom textarea").val(); //重新获取文本域内容
        var counter=text.length; //重新获取文本域内容长度
        if(counter>=100) {
            $("#numtj var").text("对不起，输入的字符超长了！");//文本域
            //长度大于100时提示
            $("#edit_myfrom textarea").attr("readonly",true);//设置文本域为
            //只读 }
        }
        else
            $("#numtj var").text(100-counter);
    });
});
```

任务2 制作图片提示效果



- 任务描述：

- 当鼠标移动到小图时，显示该图片的大图片及图片的标题，如图所示，当鼠标在小图范围内移动时，提示图片跟随鼠标移动；当鼠标移出小图片时，提示的大图片不显示。

有效果：



无效果：



任务2 制作图片提示效果



- 任务分析：

- 1. 当鼠标滑入超链接时创建一个<div>元素，<div>元素的内容为title属性的值。
- 2. 将创建的<div>元素追加到文档中。
- 3. 设置新建的<div>元素的x坐标和y坐标，使它显示在鼠标位置的旁边。
- 4. 当鼠标滑出超链接时，移除<div>元素。



事件对象



JavaScript 在事件处理函数中默认传递了 **event** 对象，也就是事件对象。但由于浏览器的兼容性，开发者总是会做兼容方面的处理。jQuery 在封装的时候，解决了这些问题，并且还创建了一些非常好用的属性和方法。



事件对象的属性



- **事件对象**: 当触发事件时, 事件对象就被创建了. 在程序中使用事件只需要为函数添加一个参数. 该事件对象只有事件处理函数才能访问到. 事件处理函数执行完毕后, 事件对象就被销毁了.
- `event.pageX`, `event.pageY`: 获取到光标相对于页面的 `x`, `y` 坐标.



事件对象的属性



event 对象的属性

属性名	描述
type	获取这个事件的事件类型，例如：click
target	获取绑定事件的 DOM 元素
data	获取事件调用时的额外数据
relatedTarget	获取移入移出目标点离开或进入的那个 DOM 元素
currentTarget	获取冒泡前触发的 DOM 元素，等同与 this
pageX/pageY	获取相对于页面原点的水平/垂直坐标
screenX/screenY	获取显示器屏幕位置的水平/垂直坐标(非 jQuery 封装)
clientX/clientY	获取相对于页面视口的水平/垂直坐标(非 jQuery 封装)
result	获取上一个相同事件的返回值
timeStamp	获取事件触发的时间戳



事件冒泡



如果在页面中重叠了多个元素，并且重叠的这些元素都绑定了同一个事件，那么就会出现冒泡问题。



例：



假设网页上有两个元素按钮和div，其中按钮嵌套在div里，并且都被绑定了click事件，同时页面文档上也绑定了click事件。

```
$('#input').click(function () {  
    $("#msg").html($("#msg").html()+"按钮被触发了! <br/>"); });  
$('#div').click(function () {  
    $("#msg").html($("#msg").html()+"div 层被触发了! <br/>"); });  
$(document).click(function () {  
    $("#msg").html($("#msg").html()+"文档页面被触发了! <br/>");});
```

当单击按钮时，会输出**三条记录**，这就是由**事件冒泡**引起的。

阻止事件冒泡



事件冒泡可能会引起预料之外的效果，所以需要阻止事件冒泡。

阻止事件冒泡：`event.stopPropagation();`

阻止默认行为：`e.preventDefault();`

可以使用`return false`上述两个方法实现相应功能



任务实现



- 1.HTML结构

```
<ul>
```

```
    <li><a href="images/iphone5s-big.jpg" class="tooltip"
title="iphone5s"></a></li>
```

```
    <li><a href="images/samsung-big.jpg" class="tooltip" title="三星
Note"></a></li>
```

```
    <li><a href="images/xiaomiM4-big.jpg" class="tooltip" title="小米
M4"></a></li>
```

```
</ul>
```

任务实现



• 2.提示效果实现

```
$(function(){
    var x = 10;      var y = 20;
    $("a.tooltip").mouseover(function(e){
        this.myTitle = this.title;
        this.title = "";
        var imgTitle = this.myTitle? "<br/>" + this.myTitle : "";
        var tooltip = "<div id='tooltip'><img src='"+ this.href + "'
alt='产品预览图
'/>" + imgTitle + "</div>"; //创建 div 元素
        $("body").append(tooltip);           //把它追加到文档中
        $("#tooltip")
            .css({
                "top": (e.pageY+y) + "px",
                "left": (e.pageX+x) + "px"
            }).show("fast"); //设置x坐标和y坐标, 并且显示
```





- 3.提示效果实现

```
.mouseout(function(){
    this.title = this.myTitle;
    $("#tooltip").remove(); //移除
})
.mousemove(function(e){
    $("#tooltip")
        .css({
            "top": (e.pageY+y) + "px",
            "left": (e.pageX+x) + "px"
        });
});});
```

任务3 改变图片透明度



- 任务描述
- 如图所示，页面图片初始显示时是半透明效果，当鼠标移入图片上，图片完全显示出来，当鼠标移开图片时，图片恢复半透明效果。



任务3 改变图片透明度



- 任务分析

- 1. 页面初始状态时设置图片样式为半透明显示。
- 2. 当鼠标在图片上悬停时，设置图片样式为完全显示，当鼠标移开，重新设置图片为半透明显示。
- 鼠标悬停和鼠标移开会触发 `.mouseenter` 和 `.mouseleave` 事件，在 JQuery 中为了方便使用，封装了合成事件 `hover()` 方法。



合成事件



- `hover()`方法用于模拟光标悬停事件，语法结构如下：
- **`hover(enter,leave)`**
 - 用于模拟**光标悬停**事件.当光标移动到元素上时,会触发指定的第一个函数,当光标移出这个元素时,触发指定的第二个函数.



改善示例



- 使用hover合成事件改善下例

用户交互设计学习参考书

- JavaScript DOM编程
- 锋利的JQuery
- JQuery入门与提高
- JavaScript高级编程
- JQuery权威指南
- JQuery实战

用户交互设计学习参考书



特殊事件



- `one()`事件只执行一次则被删除，语法结构如下：
- **`one(event,[data],function)`**

例：单击“人在珠海” div时，弹出一次“欢迎来到珠海！”

```
<div id="city">人在珠海</div>
```

```
$(document).ready(function(e) {
```

```
    $("#city").one("click",function(){
```

```
        alert("欢迎来到珠海！ ");
```

```
    }
```



任务实现



```
<script text="text/javascript">
    const OUT_OPACITY = 0.4;           //初始状态
    const OVER_OPACITY = 1.0;        //完全显示
    $(function() {
        $("#miniGallery img").css("opacity", OUT_OPACITY) //设置透明度
        .hover(
            //使用复合事件
            function () {
                $(this).css({opacity:OVER_OPACITY});
            },
            function () {
                $(this).css({opacity:OUT_OPACITY});
            }
        );
    });
</script>
```

拓展



- 在之前的例子中，我们所用接触到一些事件，但这些事件都是通过**用户执行**相应的操作后触发的。有时我们更希望让事件**自动执行**，而不用手动触发，或者在一个事件中调用另一个事件的处理函数。这时我们可以考虑使用**模拟操作**。
- 模拟操作可以使用**trigger()**方法来完成。

例如：

```
$("#pos").click(function(){  
    $("#btn").trigger("click");//模拟操作触发的#btn的click事件。  
})
```



触发自定义事件



- 自定义事件必须靠 代码 来触发。依靠用户操作时无法直接触发的。
- **trigger()**方法来完成。

例如：

```
$("#pos").bind("myclick",function(){  
    alert("hello!");  
}) //绑定事件
```

```
$("#pos").trigger("myclick"); //触发事件
```



小结



1. 使用bind方法可以为元素绑定事件，使用unbind方法可以删除事件。
2. JQuery中的event对象封装了很多非常好用的属性和方法。
3. 如果在页面中重叠了多个元素，并且重叠的这些元素都绑定了同一个事件，就会出现冒泡问题，可以使用stopPropagation方法或return false阻止冒泡。
4. hover实际是mouseenter和mouseleave的合成事件；特殊事件one只会触发一次就删除。
5. 模拟操作trigger()方法可以模拟用户的操作，也可以触发用户自定义的事件。



教育改变生活

