logo.png

jquery

H5 web前端

[www.bufanui.com](http://www.bufanui.com)

jQuery其他必备知识

# 一、jQuery事件机制

## 1.1 jQuery 事件的绑定

简单事件绑定🡺bind事件绑定🡺 delegate事件绑定🡺 on

* **简单事件绑定：**

click(handler) 单击事件

blur(handler) 失去焦点事件

mouseenter(handler) 鼠标进入事件

mouseleave(handler) 鼠标离开事件

dbclick(handler) 双击事件

change(handler) 改变事件，如：文本框值改变，下来列表值改变等

focus(handler) 获得焦点事件

keydown(handler) 键盘按下事件

更多：<http://www.w3school.com.cn/jquery/jquery_ref_events.asp>

* **bind方式**（不推荐，1.7以后的jQuery版本被on取代）

作用：给匹配到的元素直接绑定事件

$("p").bind("click mouseenter", function(e){

});

* **delegate方式**（特点：性能高，支持动态创建的元素）

作用：给匹配到的元素绑定事件，对支持动态创建的元素有效

$(".parentBox").delegate("p", "click", function(){

});

与前两种方式最大的优势：减少事件绑定次数提高效率，支持动态创建出来的元素绑定事件！

* **on方式**（兼容zepto(移动端类似jQuery的一个库)，建议使用的方式）

**jQuery1.7版本后，jQuery用on统一了所有的事件处理的方法**

作用：给匹配的元素绑定事件，包括了上面所有绑定事件方式的优点

语法：

* + - **$(selector).on(events[,selector][,data],handler);**
    - **$(selector).on(events[,selector], handler);**
    - **$(selector).on(events,handler);**

## 1.2 事件解绑

* **unbind() 方式**

作用：解绑 bind方式绑定的事件

$(selector).unbind(); // 解绑所有事件

$(selector).unbind(“click”,fn); // 解绑固定事件

$(selector).unbind(“click”); // 解除一类事件

* **undelegate() 方式**

作用：解绑delegate方式绑定的事件

$( selector ).undelegate();

$( selector).undelegate( “click” );

$( selector).undelegate( “xx”,“click”);

$( selector).undelegate( “xx”,“click”,fn );

* **off解绑on方式绑定的事件**

$(selector).off();

$(selector).off(“click”);

// 解绑所有代理的click事件，元素本身的事件不会被解绑

$(selector).off( “click”, “xx” );

$(selector).off( “click”, “xx” ,fn);

## 1.3 事件触发

简单事件触发

**$(selector).click();** //触发 click事件

trigger方法触发指定事件

**$(selector).trigger(“click”)**;

triggerHandler() 方法触发被选元素的指定事件类型。

**$(selector).triggerHandler(“click”);**

### triggerHandler与 trigger() 方法相比的不同之处

* 它不会引起事件（比如表单提交）的默认行为（细节，开发并不在意）
* .trigger() 会操作 jQuery 对象匹配的所有元素，而 .triggerHandler() 只影响第一个匹配元素。
* 由 .triggerHandler() 创建的事件不会在 DOM 树中冒泡；如果目标元素不直接处理它们，则不会发生任何事情。
* 该方法的返回的是事件处理函数的返回值，而不是具有可链性的 jQuery 对象。此外，如果没有处理程序被触发，则这个方法返回 undefined。

什么时候用？ 一般都不用。如果在特殊时候存在这些细微的差别的时候，需要考虑他们之间的不同的

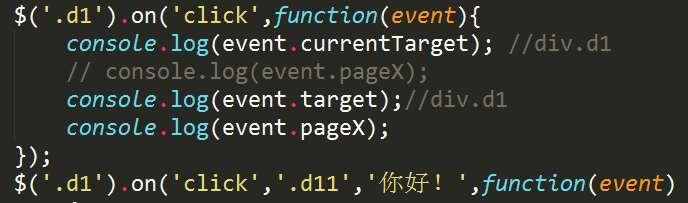
# 二、jQuery事件对象介绍

event.data 传递给事件处理程序的额外数据

event.currentTarget 等同于this，当前DOM对象

event.pageX  鼠标相对于文档左部边缘的位置

**event.target** 触发事件源，不一定===this





**event.stopPropagation()；** 阻止事件冒泡

**event.preventDefault();**  阻止默认行为

event.type 事件类型：click，dbclick…

event.which 鼠标的按键类型：左1 中2 右3

event.keyCode 键盘按键代码

# 三、jQuery补充

## 3.1 链式编程

链式编程原理：return this; 调用”任何”一个方法都是返回了对象本身

通常情况下，只有设置操作才能把链式编程延续下去。因为获取操作的时候，会返回获取到的相应的值，无法返回 this。

~~end();~~ // 结束当前链最近的一次过滤操作，并且返回匹配元素之前的状态。

## 3.2 隐式迭代

隐式迭代的意思是：在方法的内部会为匹配到的所有元素进行循环遍历，执行相应的方法；而不用我们再进行循环，简化我们的操作，方便我们调用。

如果获取的是多元素的值，大部分情况下返回的是第一个元素的值。

案例【五角星评分控件】

## 3.3 each方法

大部分情况下是不需要使用each方法的，因为jQuery的隐式迭代特性。

如果要对每个元素做不同的处理，这时候就用到了each方法

作用：遍历jQuery对象集合，为每个匹配的元素执行一个函数

**$(selector).each(function(index,element){});**

Element是一个 js对象，需要转换成jquery对象

【案例】 什么都看不见

## 3.4 多库共存

此处多库共存指的是：jQuery占用了$ 和jQuery这两个变量。当在同一个页面中引用了jQuery这个js库，并且引用的其他库（或者其他版本的jQuery库）中也用到了$或者jQuery这两个变量，那么，要保证每个库都能正常使用，这时候就有了多库共存的问题。

其他一些 JavaScript 框架包括：MooTools、Backbone、Sammy、Cappuccino、Knockout、JavaScript MVC、Google Web Toolkit、Google Closure、Ember、Batman 以及 Ext JS。

其中某些框架也使用 $ 符号作为简写（就像 jQuery），如果您在用的两种不同的框架正在使用相同的简写符号，有可能导致脚本停止运行。

jQuery 的团队考虑到了这个问题，并实现了 noConflict() 方法。

解决方式：

$.noConflict();让jQuery释放对$的控制权，让其他库能够使用$符号，达到多库共存的目的。此后，只能使用jQuery来调用jQuery提供的方法

# 四、jQuery插件机制

jQuery这个js库，虽然功能强大，但也不是面面俱到包含所有的功能。

jQuery是通过插件的方式，来扩展它的功能：

当你需要某个插件的时候，你可以“安装”到jQuery上面，然后，使用。

当你不再需要这个插件，那你就可以从jQuery上“卸载”它。

（联想：手机软件，安装的app就好比插件，用的时候安装上，不用的时候卸载掉）

### 第三方插件

jQuery.color.js

animate()自定义动画：不支持背景的动画效果

[animate动画支持的属性列表](http://www.w3school.com.cn/jquery/effect_animate.asp)

### 制作插件

如何创建jQuery插件：

<http://learn.jquery.com/plugins/basic-plugin-creation/>

全局jQuery函数扩展方法

$.pluginName = function() {};

jQuery对象扩展方法 fn=prototype

$.fn. pluginName = function() {};

//考虑 封装一个自己的scrller库

### jQuery插件

日历

轮播

灯箱

goto

视差

瀑布流

任何一种jq插件的使用过程

1. 下载插件库
2. 在页面引入插件的css或者字体图片等（如果有的话）
3. 在页面引入jQuery.js
4. 在页面引入插件的js文件（core）
5. 在页面通过插件的api初始化插件 即可使用（通过查看相应的API）

学会查看api 学会swiper插件的使用

了解更多插件的作用 在以后的工作中 随时可能遇到需求任何插件

常用的插件有 \*\*日历插件 **\*\*滚动插件** \*\*表格插件 **\*\*报表插件echar**  \*\*树行插件 \*\*富文本插件