

## 15天学会jQuery

### 15 Days of jQuery(Day 11) --- 使用不苛刻的javascript代码实现多文件上传

好几个月以前，当我在追逐互联网上AJAX热潮的时候，我在 FiftyFourEleven网站上发现了一篇使用创新的javascript代码实现当时正在困扰我的“单文件元素实现多文件上传”的文章。

所以当我想写作《15天漫游jQuery》的时候，我第一个想到的就是用jQuery实现这个功能。

#### 接触易用性狂热爱好者

几天前当我检查网站记录的时候，发现了一条遗漏的文章trackback。跟过去看的时候我发现我的两篇jQuery文章被作者引用来证明他为什么讨厌javascript。

根据这个人的说法，任何工具或技术如果没有将易用性放在第一位都将成为垃圾。

尽管我很不同意这位仁兄一杆子打死的态度，但他还是让我对这篇详细教程有所留意。当我在编写一个简单网页效果的时候，我会尽量小心谨慎的处理。这样如果网站访客们决定关闭javascript代码执行功能的时候，他们仍然可以正常使用网站的功能。

#### 关于第一价值的两个教程

- 使用一个文件输入元素实现多文件上传，并让整个交互过程流畅舒适。
- 让多文件上传更加人性化，但要避免以牺牲可用性为代价。关键在于使用不苛刻的javascript代码制作多文件输入区域。

#### 演示

- 只有一个文件输入元素，但添加了jQuery和其他代码实现较为亲近用户的多文件上传功能。

演示一地址

- 在页面(x)html代码中使用了多个文件输入元素，但通过jQuery调整为与第一个演示类似的显示页面效果。优点是代码是不苛刻的。。。即使关闭了javascript执行，用户也能上传多个文件。

演示二地址(这个演示有bug导致无法使用，作者修正了bug，使用这个后面链接查看演示演示二地址(no bug))

#### 解释

##### 单文件输入框

jQuery的\$(document).ready() 函数的工作有两个：

创建一个用来显示最大允许上传文件数的div元素。 查找文件上传框（假设这里只有一个），然后给它附上一个onChange事件。

```
$("#input[@type=file]").change(function(){
    doIt(this, fileMax);
});
```

doit()函数（简单又好记，呵呵~）检查是否达到了最大文件数量限制，如果不是，它会隐藏当前文件输入框，在父div里添加一个新的文件输入框，将输入框内的文件名使用id “files\_list” 作为标记，在最后添加一个

“删除”按钮。

在DOM树中导航，我使用jQuery的parent()函数，然后用remove()函数移除元素。我还使用了append()和prepend()函数分别添加文件名和新的输入框。

两个关键点

- 最大文件上传数量设定:

```
var fileMax = 3;
```

- 输入框必须有适当的定位措施:

```
<input type="file" class="upload" name="fileX[]" />
```

这样弄以后输入框可疑由访问者决定添加还是删除，没有任何关于id或名称的操作。当这个窗体代码发送给服务器端脚本的时候，相关信息就已经被存放在了一个数组中了。

### 多文件输入框

首先，文件允许上传的数量由页面中的文件输入框的数量决定。其次，你仍然需要通过某种方法为每个输入框接收到的内容用一个数组存放。

```
<input type="file" class="upload" name="fileX[]" />
```

第二个演示跟前面的比起来最大的不同在于，我遍历了每个文件输入框并在其内容有改动时执行doit()函数。通过遍历每一个输入框，我可以为我的代码添加有用的额外信息：输入框内容在“堆栈”中的顺序。

换句话说，当这段代码执行时，它会特别指定第一个输入框，或者第二个，抑或第三个。

代码见下:

```
$("#input[@type=file]:nthoftype("+  
n+")")
```

jQuery的灵活性允许我们使用CSS和XPath描述语句定位指定的元素位置。

你会发现当一个文件被选中时，文件输入框都会被文件名称覆盖。点击文件名就可以选择其他不同的文件。

## 15 Days of jQuery(Day 12) --- jQuery Lightbox (插件)

Cody Lindley 移植的第一版“Thickbox”让我第一次感受到了jQuery的魅力。后来他又做了一些 代码升级以修复若干跨浏览器的兼容性问题。

### 一些需要注意的地方

\$(document).ready 取代了TB\_init() 函数，作用是在每个包含对象名“thickbox”的链接上附加一个onClickListener。

```
function TB_init(){  
    $(".thickbox").click(function(){  
        var t = this.title || this.innerHTML || this.href;  
        TB_show(t,this.href);  
    });  
}
```

```
this.blur();
return false;
});
```

当这些链接被点击时，TB\_show()函数就将执行。

```
$("#body")
.append("<div id='TB_overlay'></div><div id='TB_window'></div>");
$("#TB_overlay").click(TB_remove);
$(window).resize(TB_position);
$(window).scroll(TB_position);
$("#TB_overlay").show();
$("#body").append("<div id='TB_load'><div id='TB_loadContent'><img
src='images/circle_animation.gif' /></div></div>");
```

如你所见，在文档body元素前添加了两个div元素。换句话说，这两个div元素将被添加在页面html代码的body关闭元素前。

覆盖的div将使用一个特定的包含不透明外表的CSS文件指定表现。TB\_window的代码用来通过AJAX在页面中放置一张图片或者加入另一个页面。\$(window).resize 和\$(window).scroll 告诉jQuery在用户重新调整窗口大小或者拖动页面翻页的时候执行TB\_position函数。这是保证Thickbox始终保持在窗口中心部位的手段。

接下来，Cody查询url的后缀。

```
var urlString = /\.jpg|.jpeg|.png|.gif|.html|.htm|.php|.cfm|.asp|.aspx|.jsp|.jst|.rb|.txt/g;
var urlType = url.match(urlString);
if(urlType == '.jpg' || urlType == '.jpeg' || urlType == '.png' || urlType == '.gif'){//code to show
```

如果这是一个图片文件，则jQuery的append函数会添加html代码到适当位置。

```
$("#TB_window").append("<a href=' ' id='TB_ImageOff' title='Close'><img
id='TB_Image' src='"+url+"' width='"+imageWidth+"' height='"+imageHeight+"'
alt='"+caption+"' /></a>"+
"<div id='TB_caption'>"+caption+"</div><div
id='TB_closeWindow'><a href='#' id='TB_closeWindowButton'>close</a></div>");
$("#TB_closeWindowButton").click(TB_remove);
```

另外，远程文件将使用jQuery的load()函数导入。

```
$("#TB_ajaxContent").load(url, function() {
```

## 15 Days of jQuery(Day 13) --- jQuery 表格

一位叫Klaus的朋友编写了一个小插件，用jQuery实现可用性极佳的javascript表格。

设置好正确的(x)HTML 和CSS后，你可以像下面那样创建表格：

`$.tabs(" container" ); first tab on by default` 如果你像在默认位置“上方”再添加一个表格：`$.tabs( " container" , 2); second tab on`

Klaus这里 示例，你可以看看最终效果。

### 我的改版

我稍微修改了Klaus的代码，添加了一个简单的表单用来生成表格的表头。

## 用法:

非常简单。只需要输入每个表格的表头（最多5个），然后点击表单下方的按钮。下一个页面将生成结果HTML代码，你可以复制然后粘贴到文件中。

你还需要 下载Klaus网站的CSS文件，做些你自己的修改，当然还要上传jQuery框架库到你的服务器上。

这里是表格生成器的地址。

## 15 Days of jQuery(Day 14) --- Javascript 工具提示

Cody Lindley，Thickbox的作者，日前发布了 jTip - jQuery 工具提示。

我对其中很多想法和思路拍案叫绝。我知道你已经看过很多类似的工具提示代码了。但是，Cody 的方法已经在我的工作中显露出了闪光点。

当我检查HTML代码时，我发现了一个大问题，可访问性。链接在javascript关闭的时候无法工作。我并不是倾向于一定要实现全面的可访问性，只是在这里我认为可以有其他更具亲和力的方式实现相同的功能。

尤其是，我个人不喜欢那种为了可访问性而去牺牲可用性来实现在提示框上链接另一个页面链接的方法。我喜欢这个提示框 - 不是对Cody不尊重，只是在我这里我“需要”它能够在各种情况下工作。

今天我要提供给大家的是Cody的工具提示代码的小小修改。如果你不是Cody工具提示的爱好者的话，我的改版对你来说也许不是很在意。但如果你喜欢他的作品同时希望它可以在javascript关闭的时候照常工作，这个也许是你需要的。

### 我的改动

让我产生修改想法的，是他的代码在Yahoo上的应用。我不喜欢他使用的代码：

```
<a href="yahoo.htm?width=175&link=http://www.yahoo.com"
name="Before You Click..."
id="yahooCopy"
class="jTip">Go To Yahoo</a>
```

所以我重写了他的部分代码，成了现在这个样子：

```
<a href="http://www.yahoo.com"
rel="yahoo.htm?width=175&link=yahoo&name=Before%20
%20You%20Click..."
id="yahooCopy"
class="jTip">
Go To Yahoo</a>
```

我的示例

### 改进：HTML标准校验

我的代码可以通过w3.org的测试

### 改进：命名

在我修改Cody的代码的时候我发现他使用了一个用来存储链接名称的叫做“title”的变量名，这会导致一些混淆。

我标出了这个命名问题，即使我认为这不过是个小小的失误。

### **改进：可用性**

使用我的代码，你可以让每个提示框都含有真实链接地址到另一个文档，不管内部的还是外部的。或者你只是想要那个提示框，不想关心可用性，你同样可以让链接部分留空。

选择权在你。

### **感谢**

Cody提供了伟大的代码，帮助我节省了大量的时间和精力。我的修改只是对原有代码的轻微“调整”，希望朋友们喜欢。

---

## **15 Days of jQuery(Day 15) --- 拖拽效果和选择器**

jquery中文指导/15天学会jquery\_3.txt • 最后更改: 2008/03/13 01:18 由 风之翼