**JavaScript 文件拖拽上传插件 dropzone.js 介绍**

February 19, 2014 / [编程指南](http://www.renfei.org/blog/category/programming/)

[dropzone.js](http://www.dropzonejs.com/) 是一个开源的 JavaScript 库，提供 AJAX 异步上传功能。

**安装**

下载dropzone.js文件并添加到页面中即可。Dropzone 不依赖 jQuery 框架。

**启用**

可以新建一个div元素，然后通过如下 JavaScript 代码启用 dropzone（如果你使用 jQuery）：

<div id="dropz"></div>

<script>

$("#dropz").dropzone({

url: "handle-upload.php",

maxFiles: 10,

maxFilesize: 512,

acceptedFiles: ".js,.obj,.dae"

});

</script>

如果你没有使用 jQuery 框架，也可以这样来初始化：

<div id="dropz"></div>

<script>

var dropz = new Dropzone("#dropz", {

url: "handle-upload.php",

maxFiles: 10,

maxFilesize: 512,

acceptedFiles: ".js,.obj,.dae"

});

</script>

其中url是必须的值，指明文件上传提交到哪个页面。其他的值都是可选的，如果使用默认值的话可以省略。

**接收文件**

Dropzone 并不含任何服务器端的支持和实现，利用 Dropzone 上传文件和利用下面基本的 HTML 表单对于服务器来说是一样的：

<form action="handle-upload.php" method="post" enctype="multipart/form-data">

<input type="file" name="file">

</form>

**配置 Dropzone**

此插件的特色就在于非常灵活，提供了许多可选项、事件等。下面分类介绍常用的配置项。

**功能选项**

* url：最重要的参数，指明了文件提交到哪个页面。
* method：默认为post，如果需要，可以改为put。
* paramName：相当于<input>元素的name属性，默认为file。
* maxFilesize：最大文件大小，单位是 MB。
* maxFiles：默认为null，可以指定为一个数值，限制最多文件数量。
* addRemoveLinks：默认false。如果设为true，则会给文件添加一个删除链接。
* acceptedFiles：指明允许上传的文件类型，格式是逗号分隔的 MIME type 或者扩展名。例如：image/\*,application/pdf,.psd,.obj
* uploadMultiple：指明是否允许 Dropzone 一次提交多个文件。默认为false。如果设为true，则相当于 HTML 表单添加multiple属性。
* headers：如果设定，则会作为额外的 header 信息发送到服务器。例如：{"custom-header": "value"}
* init：一个函数，在 Dropzone 初始化的时候调用，可以用来添加自己的事件监听器。
* forceFallback：Fallback 是一种机制，当浏览器不支持此插件时，提供一个备选方案。默认为false。如果设为true，则强制 fallback。
* fallback：一个函数，如果浏览器不支持此插件则调用。

**翻译选项**

* dictDefaultMessage：没有任何文件被添加的时候的提示文本。
* dictFallbackMessage：Fallback 情况下的提示文本。
* dictInvalidInputType：文件类型被拒绝时的提示文本。
* dictFileTooBig：文件大小过大时的提示文本。
* dictCancelUpload：取消上传链接的文本。
* dictCancelUploadConfirmation：取消上传确认信息的文本。
* dictRemoveFile：移除文件链接的文本。
* dictMaxFilesExceeded：超过最大文件数量的提示文本。

**添加事件监听**

如果你希望在一个事件发生时采取一些额外的操作，而不干扰 Dropzone 的默认行为，那么你应该通过添加事件监听器的办法对事件做出响应，**而非重写默认事件函数**。

重写默认事件函数的例子如下：

$("#dropz").dropzone({

addedfile: function() {

// actions...

}

});

如果你重写默认事件函数，该事件发生时插件默认采取的动作将被覆盖。大多数情况下你仅仅想在事件发生时添加自己的行为，那么应该使用on方法。

jQuery 版本：

$("#dropz").dropzone({

init: function() {

this.on("addedfile", function(file) {

// actions...

});

}

});

非 jQuery 版本：

dropz.on("addedfile", function(file) {

// actions...

});

**常用事件**

**以下事件接收 file 为第一个参数**

* addedfile：添加了一个文件时发生。
* removedfile：一个文件被移除时发生。你可以监听这个事件并手动从服务器删除这个文件。
* uploadprogress：上传时按一定间隔发生这个事件。第二个参数为一个整数，表示进度，从 0 到 100。第三个参数是一个整数，表示发送到服务器的字节数。当一个上传结束时，Dropzone 保证会把进度设为 100。注意：这个函数可能被以同一个进度调用多次。
* success：文件成功上传之后发生，第二个参数为服务器响应。
* complete：当文件上传成功或失败之后发生。
* canceled：当文件在上传时被取消的时候发生。
* maxfilesreached：当文件数量达到最大时发生。
* maxfilesexceeded：当文件数量超过限制时发生。

**以下事件接收一个 file list 作为第一个参数（仅当uploadMultiple被设为true时才会发生）**

* successmultiple
* completemultiple
* cancelmultiple

**特殊事件**

* totaluploadprogress：第一个参数为总上传进度，第二个参数为总字节数，第三个参数为总上传字节数。

**例子**

这里我使用上面的选项、事件等写了一个例子，供参考：

<div class="dropz"></div>

<script>

$(".dropz").dropzone({

url: "handle-upload.php",

addRemoveLinks: true,

dictRemoveLinks: "x",

dictCancelUpload: "x",

maxFiles: 10,

maxFilesize: 5,

acceptedFiles: ".js",

init: function() {

this.on("success", function(file) {

console.log("File " + file.name + "uploaded");

});

this.on("removedfile", function(file) {

console.log("File " + file.name + "removed");

});

}

});

</script>

**外观**

Dropzone 下载之后没有自带任何 CSS 样式（人家只有一个 js 文件嘛）。我觉得[官网](http://www.dropzonejs.com/)提供的 Demo 的[外观设计](https://github.com/enyo/dropzone/tree/master/downloads/css)就非常不错，可以供大家参考。