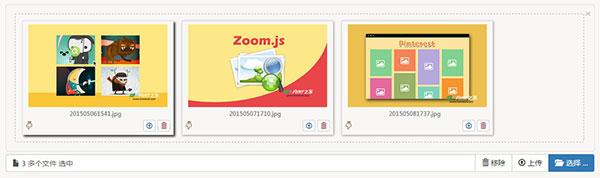
简要教程

bootstrap-fileinput是一款基于Bootstrap 3.x的[HTML5](http://www.htmleaf.com/html5/)文件上传插件。该文件上传插件带有预览图效果，可同时选择多个文件。该插件使用bootstrap CSS3样式来制作文件上传界面，美观大方。并且它提供了多国语言，你可以选择使用中文。

该文件上传插件比普通的文件上传插件功能更强大，它可以对图片、文本文件、HTML文件、视频文件、音频文件、flash文件生成预览图。另外，它还可以基于AJAX上传文件，拖拽上传文件，查看上传进度，可以选择性的预览，添加或删除文件。



该文件上传插件的特点有：

* AJAX上传功能基于HTML5 FormData（绝大多数现代浏览器都支持该属性）。在不支持该属性的浏览器中会回退为普通的文件上传组件。使用AJAX必须设置uploadUrl属性。
* 允许你添加、移除和追加文件。添加的文件可以生成预览图。
* 可以将文件拖拽到指定区域来上传文件。
* 可以一个个的删除或更新文件，也可以同时完成这些操作。
* 如果showPreview属性设置为false，或者浏览器不支持uploadUrl属性，将会回退为普通的文件上传组件。
* 可以配置文件上传等待指示，文件上传成功信息，文件上传出错信息。
* 在使用ajax上传文件时可以添加额外的表单信息。
* 可以显示文件当前上传的进度。
* 可以取消和终止当前正在上传的任务。
* 文件上传完毕会自动刷新预览区域的内容。

 插件外部依赖需求

* [Bootstrap 3.x](http://getbootstrap.com/)
* 最新版本的[jQuery](http://www.htmleaf.com/jQuery/)
* 大多数现代浏览器都支持HTML5（inputs和FileReader AP），[CSS3](http://www.htmleaf.com/css3/)和jQuery。对于IE浏览器，必须是IE10以上的版本。IE9及以下的版本会回退为普通的文件上传组件，并且不支持选择多个文件和HTML 5 FileReader API。
* AJAX上传需要浏览器支持HTML5 FormData和XHR2 (XMLHttpRequest 2)。大多数现代浏览器都支持FormData和XHR2。在不支持这些特性的浏览器中将会回退为普通的文件上传组件。

 安装

可以使用bower来安装该文件上传插件。

bower install bootstrap-fileinput

 使用方法

使用该文件上传插件首先要引入需要的外部依赖文件和fileinput.min.js及fileinput.min.css文件。

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css" rel="stylesheet">

<link href="path/to/css/fileinput.min.css" media="all" rel="stylesheet" type="text/css" />

<script src="jquery/2.1.1/jquery.min.js"></script>

<script src="path/to/js/fileinput.min.js"></script>

<!-- bootstrap.js below is only needed if you wish to the feature of viewing details

of text file preview via modal dialog -->

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js" type="text/javascript"></script>

<!-- optionally if you need translation for your language then include

locale file as mentioned below -->

<script src="path/to/js/fileinput\_locale\_<lang>.js"></script>

如果你需要使用语言国际化功能，可以添加上fileinput\_locale\_lang.js文件，中文对应的是fileinput\_locale\_zh.js文件。

**初始化插件**

// initialize with defaults

$("#input-id").fileinput();

// with plugin options

$("#input-id").fileinput({'showUpload':false, 'previewFileType':'any'});

#input-id是文件上传input元素的ID标识符（例如：type = file）。

你也可以直接通过HTML5 data属性来初始化插件。

<input id="input-id" type="file" class="file" data-preview-file-type="text" >

 配置参数

* showCaption：是否显示文件的标题。默认值true。
* showPreview：是否显示文件的预览图。默认值true。
* showRemove：是否显示删除/清空按钮。默认值true。
* showUpload：是否显示文件上传按钮。默认是submit按钮，除非指定了uploadUrl属性。默认值true。
* showCancel：是否显示取消文件上传按钮。只有在AJAX上传线程中该属性才可见可用。默认值true。
* captionClass：在标题容器上额外的class。类型string。
* previewClass：在预览区域容器上的额外的class。类型string。
* mainClass：添加在文件上传主容器。类型string。
* initialDelimiter：在initialPreview属性中用于上传多个文件时的分隔符。默认值：'\*$$\*'。
* initialPreview：类型string或array。显示的初始化预览内容。你可以传入一个简单的HTML标签用于显示图片、文本或文件。如果设置一个字符串，会在初始化预览图中显示一个文件。你可以在initialDelimiter属性中设置分隔符用于显示多个预览图。如果设置为数组，初始化预览图中会显示数组中所有的文件。

下面的CSS样式用于显示各种不同类型的文件预览图：

* + image files：CSS样式为file-preview-image
  + text files：CSS样式为file-preview-text
  + other files：CSS样式为file-preview-other

下面的例子是如何设置不同类型文件的初始化预览图：

// for image files

initialPreview: [

"<img src='/images/desert.jpg' class='file-preview-image' alt='Desert' title='Desert'>",

"<img src='/images/jellyfish.jpg' class='file-preview-image' alt='Jelly Fish' title='Jelly Fish'>",

],

// for text files

initialPreview: "<div class='file-preview-text' title='NOTES.txt'>" +

"This is the sample text file content upto wrapTextLength of 250 characters" +

"<span class='wrap-indicator' onclick='$(\"#show-detailed-text\").modal(\"show\")' title='NOTES.txt'>[…]</span>" +

"</div>"

// for other files

initialPreview: "<div class='file-preview-text'>" +

"<h2><i class='glyphicon glyphicon-file'></i></h2>" +

"Filename.xlsx" + "</div>"

* initialPreviewCount：类型int。被添加到选择的文件的初始化预览图的数量。当overwriteInitial属性设置为false时，该参数可以显示正确的文件标题。
* initialPreviewDelimiter：类型string。用于分割初始化预览图内容的分割符。只有在initialPreview属性传入的参数是字符串而不是数组时可用。默认值是：\*$$\*。
* initialPreviewConfig：类型：array。每一个initialPreview项的配置属性。数组中的每一个元素都应该是下面关键字组成的对象或关联数组：
* - `caption`: \_string\_, the caption or filename to display for each initial preview item content.
* - `width`: \_string\_, the CSS width of the image/content displayed.
* - `url`: \_string\_, the URL for deleting the image/content in the initial preview via AJAX post response. This will default to `deleteUrl` if not set.
* - `key`: \_string|object\_, the key that will be passed as data to the `url` via AJAX POST.
* - `extra`: \_object|function\_, the extra data that will be passed as data to the initial preview delete url/AJAX server call via POST. This will default to `deleteExtraData` if not set.

下面是配置initialPreviewConfig属性的一个例子。

// setup initial preview with data keys

initialPreview: [

"<img src='/images/desert.jpg' class='file-preview-image' alt='Desert' title='Desert'>",

"<img src='/images/jellyfish.jpg' class='file-preview-image' alt='Jelly Fish' title='Jelly Fish'>",

],

// initial preview configuration

initialPreviewConfig: [

{

caption: 'desert.jpg',

width: '120px',

url: '/localhost/avatar/delete',

key: 100,

extra: {id: 100}

},

{

caption: 'jellyfish.jpg',

width: '120px',

url: '/localhost/avatar/delete',

key: 101,

extra: function() {

return {id: $("#id").val()};

},

}

]

注意：ajax delete操作会通过POST向服务器发送下面的数据：

* + key：在initialPreviewConfig['key']中设置的key值。
  + 通过initialPreviewConfig['extra']或deleteExtraData格式传入的其它作为key: value键值对数据。
* initialPreviewShowDelete：是否为每一个由initialPreview插件的预览图创建一个删除按钮。
* previewThumbTags：类型array。预览图的一组标签。它会在预览图生成之后动态的替换掉缩略图标签。例如：
* // change thumbnail footer template

layoutTemplates.footer = '

\n' +

'

{caption}

\n' +

' {CUSTOM\_TAG\_NEW}\n' +

' {CUSTOM\_TAG\_INIT}\n' +

' {actions}\n' +

'

';

// set preview template tags

previewThumbTags = {

'{CUSTOM\_TAG\_NEW}': 'CUSTOM MARKUP',

'{CUSTOM\_TAG\_INIT}': ' '

};

上面只是该文件上传插件一部分的属性，要查看完整的属性和可用方法，可以查看：<https://github.com/kartik-v/bootstrap-fileinput/>。