# <input type="file">

使用 type="file"的 [<input>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \o "This is a link to an unwritten page) 元素使得用户可以选择一个或多个元素以[提交表单](https://developer.mozilla.org/zh-CN/docs/Learn/HTML/Forms" \o "This is a link to an unwritten page)的方式上传到服务器上，或者通过 Javascript 的 [File API](https://developer.mozilla.org/zh-CN/docs/Web/API/File/Using_files_from_web_applications) 对文件进行操作。

|  |  |
| --- | --- |
| [Value](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "value) | A [DOMString](https://developer.mozilla.org/zh-CN/docs/Web/API/DOMString) 表示已选择文件的路径 |
| **事件** | [change](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLElement/change_event" \o "change) 和 [input](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLElement/input_event" \o "input) |
| **支持的公共属性** | [required](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input" \l "attr-required) |
| **附加属性** | [accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-accept" \o "This is a link to an unwritten page), [capture](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-capture" \o "This is a link to an unwritten page), [files](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-files" \o "This is a link to an unwritten page), [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-multiple" \o "This is a link to an unwritten page) |
| **IDL 属性** | files 和 value |
| **DOM 接口** | [HTMLInputElement](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLInputElement) |
| **属性** | [仅适用于类型为 file 元素的属性](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLInputElement) |
| **方法** | [select()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLInputElement/select" \o "select()) |

## [值](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E5%80%BC" \o "Permalink to 值)

文件 input 的 [value](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input" \l "attr-value) 属性包括了一个 [DOMString](https://developer.mozilla.org/zh-CN/docs/Web/API/DOMString)，表示已选择文件的路径。如果用户选择了多个文件，则 value 表示他们选择的文件列表中的第一个文件。 可以使用 input 的 HTMLInputElement.files 属性标识其他文件。

**Note:**

1. 如果选择了多个文件, 这个值表示第一个已选择文件的路径。JavaScript 可以[通过 input 的 files 属性访问其他文件](https://developer.mozilla.org/zh-CN/docs/Web/API/File/Using_files_from_web_applications" \l "Getting_information_about_selected_files" \o "This is a link to an unwritten page)。
2. 如果没有选择文件，该值为空字符串 ""。
3. 为了阻止恶意软件猜测文件路径，字符串[以 C:\fakepath\ 为前缀](https://html.spec.whatwg.org/multipage/input.html" \l "fakepath-srsly)。

## [附加属性](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E9%99%84%E5%8A%A0%E5%B1%9E%E6%80%A7" \o "Permalink to 附加属性)

除了被所有 [<input>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \o "This is a link to an unwritten page) 元素共享的公共属性，file 类型的 input 还支持下列属性：

| **属性** | **说明** |
| --- | --- |
| [accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "accept) | 一个或多个 [unique file type specifiers](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "unique_file_type_specifiers) 描述允许的文件类型 |
| [capture](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "capture) | 捕获图像或视频数据的源 |
| [files](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "files) | [FileList](https://developer.mozilla.org/zh-CN/docs/Web/API/FileList) 列出了已选择的文件 |
| [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "multiple) | 布尔值，如果出现，则表示用户可以选择多个文件 |

### **[accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-accept)**

[accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Attributes/accept" \o "This is a link to an unwritten page) 属性是一个字符串，它定义了文件 input 应该接受的文件类型。这个字符串是一个以逗号为分隔的 [唯一文件类型说明符](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "unique_file_type_specifiers)列表。由于给定的文件类型可以用多种方式指定，因此当你需要给定格式的文件时，提供一组完整的类型指定符是非常有用的。

例如，有许多方法可以识别 Microsoft Word 文件，所以接受 Word 文件的站点可以使用一个<input>：

<input type="file" id="docpicker"

accept=".doc,.docx,application/msword,application/vnd.openxmlformats-officedocument.wordprocessingml.document">

### **[capture](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-capture)**

[capture](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Attributes/capture" \o "This is a link to an unwritten page) 属性是一个字符串，如果 [accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Attributes/accept" \o "This is a link to an unwritten page) 属性指出了 input 是图片或者视频类型，则它指定了使用哪个摄像头去这些数据。值 user 表示应该使用前置摄像头和/或麦克风。值 environment 表示应该使用后置摄像头和/或麦克风。如果缺少此属性，则 [user agent](https://developer.mozilla.org/zh-CN/docs/Glossary/User_agent) 可以自由决定做什么。如果请求的前置模式不可用，则用户代理可能退回到其首选的默认模式。

**Note:** capture 以前是一个布尔类型的属性，如果存在，则请求使用设备的媒体捕获设备（如：摄像机），而不是请求一个文件输入。

### **[files](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-files)**

[FileList](https://developer.mozilla.org/zh-CN/docs/Web/API/FileList) 对象每个已选择的文件。如果 [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-multiple" \o "This is a link to an unwritten page) 属性没有指定，则这个列表只有一个成员。

### **[multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-multiple)**

当指定布尔类型属性 [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Attributes/multiple" \o "This is a link to an unwritten page), 文件 input 允许用户选择多个文件。

## [非标准属性](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E9%9D%9E%E6%A0%87%E5%87%86%E5%B1%9E%E6%80%A7" \o "Permalink to 非标准属性)

除了上面列出来的属性，下面的非标准属性在某些浏览器中可用。你应该尽量避免使用它们，因为这样做将限制代码在没有实现它们的浏览器中的运行能力。

| **属性** | **描述** |
| --- | --- |
| [webkitdirectory](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "webkitdirectory) | 一个布尔值，表示是否仅允许用户选择一个目录（或多个目录，如果 [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "multiple) 也出现的话） |

### **[webkitdirectory](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-webkitdirectory)**

如果出现布尔属性 webkitdirectory，表示在文件选择器界面中用户只能选择目录。更多细节和示例见  [HTMLInputElement.webkitdirectory](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLInputElement/webkitdirectory)。

**Note:** 尽管 webkitdirectory 最初仅为基于webkit的浏览器实现，它还在Microsoft Edge和Firefox 50及其后版本中可用。然而，尽管它有相对广泛的支持，它仍然是非标准的。除非你别无选择，否则不要使用它。

## [唯一文件类型说明符](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E5%94%AF%E4%B8%80%E6%96%87%E4%BB%B6%E7%B1%BB%E5%9E%8B%E8%AF%B4%E6%98%8E%E7%AC%A6" \o "Permalink to 唯一文件类型说明符)

**唯一文件类型说明符**是一个字符串，表示在 file 类型的 [<input>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \o "This is a link to an unwritten page) 元素中用户可以选择的文件类型。每个唯一文件类型说明符可以采用下列形式之一：

* 一个以英文句号（"."）开头的合法的不区分大小写的文件名扩展名。例如： .jpg，.pdf 或 .doc。
* 一个不带扩展名的 MIME 类型字符串。
* 字符串 audio/\*， 表示“任何音频文件”。
* 字符串 video/\*，表示 “任何视频文件”。
* 字符串 image/\*，表示 “任何图片文件”。

accept 属性的值是一个包含一个或多个（用逗号分隔）这种唯一文件类型说明符的字符串。 例如，一个文件选择器需要能被表示成一张图片的内容，包括标准的图片格式和 PDF 文件，大概是这样的：

<input type="file" accept="image/\*,.pdf">

## [使用文件输入](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E4%BD%BF%E7%94%A8%E6%96%87%E4%BB%B6%E8%BE%93%E5%85%A5" \o "Permalink to 使用文件输入)

### [基本示例](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E5%9F%BA%E6%9C%AC%E7%A4%BA%E4%BE%8B" \o "Permalink to 基本示例)

<form method="post" enctype="multipart/form-data">

<div>

<label for="file">Choose file to upload</label>

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

</div>

<div>

<button>Submit</button>

</div></form>

产生如下结果：

**Note**: 你也可以在 GitHub 中找到这个示例 — [源代码](https://github.com/mdn/learning-area/blob/master/html/forms/file-examples/simple-file.html)和[在线运行](https://mdn.github.io/learning-area/html/forms/file-examples/simple-file.html)。

无论用户的设备或操作系统是什么，文件输入都提供一个按钮，打开一个允许用户选择文件的文件选择对话框。

包含上述的 [multiple](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-multiple" \o "This is a link to an unwritten page) 属性，指定可以一次选择多个文件。 用户可以用他们选择的平台允许的任何方式从文件选择器中选择多个文件（如，按住 **Shift** or **Control**，然后单击）。如果您只想让用户为每个 <input> 选择一个文件，那么省略 multiple 属性。

### [获取已选择文件的信息](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E8%8E%B7%E5%8F%96%E5%B7%B2%E9%80%89%E6%8B%A9%E6%96%87%E4%BB%B6%E7%9A%84%E4%BF%A1%E6%81%AF" \o "Permalink to 获取已选择文件的信息)

被选择的文件以 HTMLInputElement.files 属性返回，它是一个包含一列 [File](https://developer.mozilla.org/zh-CN/docs/Web/API/File) 对象的 [FileList](https://developer.mozilla.org/zh-CN/docs/Web/API/FileList) 对象。FileList 的行为像一个数组，所以你可以检查 length 属性来获得已选择文件的数量。

每个 File 包含下列信息：

**name**

文件名。

**lastModified**

一个数字，指定文件最后一次修改的日期和时间，以 UNIX 新纪元（1970年1月1日午夜）以来的毫秒数表示。

**size**

以字节数为单位的文件大小。

**type**

文件的 [MIME 类型](https://developer.mozilla.org/zh-CN/docs/Web/HTTP/Basics_of_HTTP/MIME_types)。

**webkitRelativePath**

一个字符串，指定了相对于在目录选择器中选择的基本目录的文件路径（即，一个设置了 [webkitdirectory](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-webkitdirectory" \o "This is a link to an unwritten page) 属性的 file 选择器）。 这是非标准的，应该谨慎使用。

**Note**: 你可以在所有现代浏览器中读写 HTMLInputElement.files 的值；最近这被添加到 Firefox 57 中（见 [bug 1384030](https://bugzilla.mozilla.org/show_bug.cgi?id=1384030)）。

### [限制可接受的文件类型](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E9%99%90%E5%88%B6%E5%8F%AF%E6%8E%A5%E5%8F%97%E7%9A%84%E6%96%87%E4%BB%B6%E7%B1%BB%E5%9E%8B" \o "Permalink to 限制可接受的文件类型)

通常，你不希望用户能够选择任意类型的文件；相反，你通常希望它们选择特定类型的文件。例如，如果你的文件输入让用户上传个人资料图片，您可能希望他们选择 Web 兼容的图像格式，如 [JPEG](https://developer.mozilla.org/zh-CN/docs/Glossary/jpeg) 或 [PNG](https://developer.mozilla.org/zh-CN/docs/Glossary/PNG)。

可以用 [accept](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "attr-accept" \o "This is a link to an unwritten page) 属性指定可接受的文件类型，它是一个以逗号间隔的文件扩展名和 MIME 类型列表。一些例子：

* accept="image/png" 或 accept=".png" — 接受 PNG 文件。
* accept="image/png, image/jpeg" 或 accept=".png, .jpg, .jpeg" — 接受 PNG 或 JPEG 文件。
* accept="image/\*" — 接受带有一个 image/\* MIME 类型的任何文件。（许多移动设备也允许用户在使用它时用摄像头拍照。）
* accept=".doc,.docx,.xml,application/msword,application/vnd.openxmlformats-officedocument.wordprocessingml.document" — 接受 MS Word 文档之类的任何文件。

让我们来看一个更完整的例子：

<form method="post" enctype="multipart/form-data">

<div>

<label for="profile\_pic">Choose file to upload</label>

<input type="file" id="profile\_pic" name="profile\_pic"

accept=".jpg, .jpeg, .png">

</div>

<div>

<button>Submit</button>

</div></form>

上述代码产生和之前示例相似的结果：

**Note**: 你也可以在 GitHub 中找到这个示例 — [源代码](https://github.com/mdn/learning-area/blob/master/html/forms/file-examples/file-with-accept.html)和[在线运行](https://mdn.github.io/learning-area/html/forms/file-examples/file-with-accept.html)。

这可能看起来很相似，但是如果你尝试用该输入选择一个文件，你将看到文件选择器只允许你选择 accept 值指定的文件类型（实际特性会按不同的浏览器和操作系统有所不同）。

accept 属性不验证所选文件的类型；它只是为浏览器提供提示来引导用户选择正确的文件类型。用户仍然可以（在大多数情况下）在文件选择器中切换一个选项，使其能够覆盖此选项并选择他们希望的任何文件，然后选择不正确的文件类型。

因此，您应该确保 accept 属性得到适当的服务器端验证的支持。

### [注意事项](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E6%B3%A8%E6%84%8F%E4%BA%8B%E9%A1%B9" \o "Permalink to 注意事项)

你不能从脚本中设置文件选取器的值 — You cannot set the value of a file picker from a script — 做下面这样的事情是没有效果的：

const input = document.querySelector("input[type=file]");

input.value = "foo";

当使用一个 <input type="file"> 选择一个文件，When a file is chosen using an <input type="file">，出于明显的安全原因，源文件的实际路径没有显示在 input 的 value 属性中。相反，显示了文件名，并用 C:\fakepath\ 附加在路径的开头。这种怪癖有一些历史原因，但它在所有现代浏览器中都受到支持，而且实际上[在规范中也有定义](https://html.spec.whatwg.org/multipage/forms.html" \l "fakepath-srsly)。

## [示例](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Input/file" \l "%E7%A4%BA%E4%BE%8B" \o "Permalink to 示例)

在本例中，我们将展示一个稍微高级一点的文件选择器，它利用了在 HTMLInputElement.files 属性中可用的文件信息，并且展示了一些聪明的技巧。

**Note**: 你可以在 GitHub 中查看本示例的完整源代码 — [file-example.html](https://github.com/mdn/learning-area/blob/master/html/forms/file-examples/file-example.html) （[在线运行](https://mdn.github.io/learning-area/html/forms/file-examples/file-example.html)）。我们不会解释 CSS；JavaScript是主要的关注点。

首先，让我们看看 HTML：

<form method="post" enctype="multipart/form-data">

<div>

<label for="image\_uploads">Choose images to upload (PNG, JPG)</label>

<input type="file" id="image\_uploads" name="image\_uploads" accept=".jpg, .jpeg, .png" multiple>

</div>

<div class="preview">

<p>No files currently selected for upload</p>

</div>

<div>

<button>Submit</button>

</div></form>

这和我们之前看到的类似 — 没有什么要说明的。

接下来，让我们看一下JavaScript。

在脚本的第一行，我们获得了对表单 input 本身和拥有 .preview 类的 [<div>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/div) 元素的引用。然后，我们隐藏了 [<input>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/input" \o "This is a link to an unwritten page) 元素 — 我们这样做是因为文件输入往往很难看，难于设计样式，而且在跨浏览器中对它们的设计不一致。你可以通过单击 input 元素来激活它，因此，最好在视觉上隐藏 input 并将 label 设计成按钮的样式，这样用户如果想上传文件就会知道如何与之交互。

const input = document.querySelector('input');const preview = document.querySelector('.preview');

input.style.opacity = 0;

**Note:** 使用[opacity](https://developer.mozilla.org/zh-CN/docs/Web/CSS/opacity) 来隐藏文件输入，而不是使用 [visibility: hidden](https://developer.mozilla.org/zh-CN/docs/Web/CSS/visibility) 或者 [display: none](https://developer.mozilla.org/zh-CN/docs/Web/CSS/display)，因为辅助技术将后两种样式解释为文件 input 是不可交互的。

接下来，我们将[事件监听器](https://developer.mozilla.org/zh-CN/docs/Web/API/EventTarget/addEventListener)添加到 input 中，以监听选择的值的更改（在本例中，当选择文件时）。事件监听器调用我们定制的 updateImageDisplay() 函数。

input.addEventListener('change', updateImageDisplay);

每当 updateImageDisplay() 函数被调用时，我们：

* 使用一个 [while](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Statements/while) 循环来清空preview <div> 留下的内容。
* 获取包含所有已选择文件信息的 [FileList](https://developer.mozilla.org/zh-CN/docs/Web/API/FileList) 对象，并将其用一个变量 curFiles 保存。
* 通过检查 curFiles.length 是否等于0来检查是否没有文件被选择。如果是，则用 preview <div> 输出一条消息，来表示没有选择文件。
* 如果选择了文件，我们将循环遍历每个文件，并将关于它的信息输出到 preview <div>。注意事项：
* 我们使用定制的 validFileType() 函数来检查文件的类型是否正确。（例如，用 accept 属性指定的图片类型）。
* 如果是，我们：
  + 将其名称和文件大小输出到到 previous <div> 的一个列表项中（从 file.name 和 file.size获取）。定制的 returnFileSize() 函数返回一个用 bytes/KB/MB 表示的格式良好的大小（默认情况下，浏览器以绝对字节数报告大小）。
  + 通过调用 [URL.createObjectURL(curFiles[i])](https://developer.mozilla.org/zh-CN/docs/Web/API/URL/createObjectURL" \o "URL.createObjectURL(curFiles[i])) 来生成图片的一张缩略预览图。然后，通过创建一个新的 [<img>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/img) 来将这张图片也插入到列表项，并且将它的 [src](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/img" \l "attr-src) 设置为缩略图。
* 如果文件类型无效，则在列表项中显示一条消息，告诉用户需要选择一个其它的文件类型。

function updateImageDisplay() {

while(preview.firstChild) {

preview.removeChild(preview.firstChild);

}

const curFiles = input.files;

if(curFiles.length === 0) {

const para = document.createElement('p');

para.textContent = 'No files currently selected for upload';

preview.appendChild(para);

} else {

const list = document.createElement('ol');

preview.appendChild(list);

for(const file of curFiles) {

const listItem = document.createElement('li');

const para = document.createElement('p');

if(validFileType(file)) {

para.textContent = `File name ${file.name}, file size ${returnFileSize(file.size)}.`;

const image = document.createElement('img');

image.src = URL.createObjectURL(file);

listItem.appendChild(image);

listItem.appendChild(para);

} else {

para.textContent = `File name ${file.name}: Not a valid file type. Update your selection.`;

listItem.appendChild(para);

}

list.appendChild(listItem);

}

}}

定制的 validFileType() 函数接受一个 [File](https://developer.mozilla.org/zh-CN/docs/Web/API/File) 对象作为参数，然后使用 [Array.prototype.includes()](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Array/includes) 检查 fileTypes 中是否有值和文件的 type 属性匹配。如果找到匹配项，函数返回 true。如果没找到，返回 false。

const fileTypes = [

'image/jpeg',

'image/pjpeg',

'image/png'];

function validFileType(file) {

return fileTypes.includes(file.type);}

returnFileSize() 函数接受一个数字（字节数，取自当前文件的 size 属性）作为参数，并且将其转化为用 bytes/KB/MB 表示的格式良好的大小。

function returnFileSize(number) {

if(number < 1024) {

return number + 'bytes';

} else if(number >= 1024 && number < 1048576) {

return (number/1024).toFixed(1) + 'KB';

} else if(number >= 1048576) {

return (number/1048576).toFixed(1) + 'MB';

}}

这个例子是这样的；演示一下：

