# 第一章html5标准

## HTML概述

HTML是一种规范、一种标准，它通过标签符号来标记要显示的网页各部分，王爷本身是一种文本文件，通过在文本文件中添加标签，可以告诉浏览器如何显示其中的内容（如显示文本信息、处理图像、播放动画，以及网页中显示的样式等）

## 什么是网站：

网站直观的讲就是位于服务器（一台提供服务器的电脑）上的一个文件夹，文件夹中放置着网站中所有资源，包括所有网页、图片、视频、声音等。

网站三要素：服务器 站点 域名/IP地址

http：超文本传输协议：用来传输超文本的，

从

1.在地址栏输入域名之后，服务器会接受域名,在接受这个域名之前会对当前你输入的这个域名进行解析（浏览器和服务器之间是不能直接通信的）DNS服务

2.解析完成之后因为服务器有很多个，然后需要确定实在哪一个服务器上

3.然后通过tcp协议进行连接的建立，在建立链接的过程当中需要3次握手，第一次握手：现需要确定的是请求连接，第二次握手：服务器端确认请求，第三次握手：客户端确认请求

4.建立完连接之后，就可以进行传输了，客户端会把域名传输给服务器端，服务器端会返回html内容给客户端

5.客户端接收和解析

## URL统一资源定位符

URL统一资源定位符：使用同一的格式定位网络上的资源（就是网址）

格式

协议：//IP或域名：端口号/URI

协议—默认—http:// ftp:// <mailto://>

IP或域名：用来定位服务器

端口号：用户来确定通信端口，例如web服务默认端口80：sql server服务端口1433；mysql服务端口3306

URI统一资源标识符：例如：mytest/index.html

例如：

<http://news.163.com/12/1203/07/8HPK2BV800014JB6.html>

静态网站：一般指网站内容”固定不变”，谁访问内容都一样。网站扩展名一般是:.html .htm .shtml .xml

动态网站：网站内容一般来源于数据库，根据不同用户操作，产生网页内容有所不同，常用动态网站，扩展名：php .aspx .asp .jsp等

## 常用浏览器

1. Microsoft Internet Explorer是微软公司开发的一种web浏览器，仅支持windows操作系统，目前流行的版本：IE6 8 9 10
2. Mozilla Firefox六来暖气是由Mozilla基金会于开源社区共同开发的一种免费开源的web浏览器，是在全球范围内仅次于IE的浏览器。
3. Apple Safari浏览器是美国苹果公司开发的一种基于开源内核的web浏览器
4. Chrome是美国Google公司开发的开放源码浏览器。
5. Opera浏览器是挪威Opera Software ASA 开发的一款浏览器。
6. 360安全浏览器(360Safety Browser)是360安全中心推车的一款基于IE内核的浏览器，是世界之窗开发者凤凰工作室和360安全中心合作的产品。

## 浏览器内核



## Html语言

--超文本标记语言(第一版)---在1993年6月作为互联网工程工作小组(IETF)工作室草案发布(并非标准)

--HTML2.0---1995年11月作为RFC 1866发布，在RFC 2854于2000年6月发布之后被宣布已经超时

--HTML3.2---1996年1月14日，W3C推荐标准 html

--HTML4.0---1997年12月18日，W3C推荐标准HTML4.01(微小改进)---1999年12月24日W3C推荐标准

## Xml语言

--Xml即可扩展标记语言。

--可扩展标记语言，标准通用标记语言的子集，一种用于标记电子文件使其具有结构性的标记语言。

--它可以用来标记数据、定义数据类型，是一种允许用户自己的标记语言进行定义的源语言。它非常适合万维网传输，提供统一的方法来描述和交换独立于应用程序或供应商的节后化数据。

## XHTML

可扩展超文本标记语言，是一种置标语言，表现方式与超文本标记语言(HTML)类似，不过语法上更加严格。

## Html5介绍

### Html历史背景

--1993年发布了第一版html

--1995年发布html2.0

--1996年发布html3.0

--1997年发布html4.0

--1999年发布了html4.01

--2006年W3C组织，监理工作组，研究html5新标准，2008年发布了html5的工作草案，在2014年10月29日正式发布了html5新标准。

### Html5现状

--HTML5有两大特点：首先，钱花了web网页的表现性能。其次，追加了本地数据库等web应用的功能。

--广义论及HTML5时，实际指的是包括HTML、CSS和javaScript在内的一套技术组合。他希望能够减少浏览器对于需要的丰富性网络应用服务(plug-in-based rich internet application,RIA),如Adobe Flash、Mircrosoft Sliverlight，与Oracle JavaFX的需求，并且提供更多能有效增强网络应用的标准集。

### Html5使用方向

--网站(pc端/移动端)，微信公众号，html5场景，混合式app，html5游戏等等

### Html5兼容性

--移动端完全支持

--Pc端一部分支持(最新版本的chrome、firfox、opear全部支持，IE9以上部分支持)

### Html5优势

化繁为简

向下兼容

语义化标签

实用性

用户优先

通用访问

### 新增html5原生功能

Canvas绘图

跨文本消息传递

地理位置

本地存储

本地数据库

等等

## doctype

1.doctype的作用是什么？什么是严格模式和混杂模式？以及如何触发严格模式和混杂模式？

Html中有doctype：用来定义文档类型，框架型，严谨型，过渡型，

严格模式和混杂模式的区别，以及如何触发严格模式和混杂模式

严格模式就是在定义文档类型的时候将文档类型直接定义出来了

混杂模式：没有doctype

触发使用doctype来触发的

废弃的标记

<Center></center>

<font></font>

<big></big>

<bgsound></bgsound>

<marquee></marquee>

<frame></frame>

H5优势是什么？新增了哪些功能？新增了那些标记？废弃了哪些标记？

# 第二章html新增元素及属性

## html5保留的常用元素

### 基本元素

<!-- -->:定义html注释

<html></html>:html5的根元素，可以完全省略

<head>:用于定义html5的头部信息，可以完全省略

<title>:用于定义页面标题

<body>:用于定义html5文档主体部分

<style>:该元素用于引用css样式

<h1>到<h6>:定义标题文字

<p>:定义段落，该标签可以指定id，class，style等核心属性，还可以指定onclick等各种时间属性

<br>:插入一个换行指定属性可以与<p>相同

<hr>:定义水平线

<div>:定义文档中的节

<span>:同div相同，不同的是默认不会换行。

### 文本格式化元素

<b>:定义粗体文本，可以指定id，class，style等核心，onclick等各种属性

<i>:定义斜体，同上

<em>:定义前调文本，实际效果和斜体文本一样

<strong>:定义粗体文本，和<b>标记的作用和用法基本相同

<small>:定义小号字体文本

Html5删除原有的<big>字体，保留了SMALL，主要用于版权声明，注意事项，法律规定等声明性文字

<sup>:定义上标文本

<sub>:定义下标文本

<bdo>:定义文本显示方向，其他同上，此外也可以指定dir属性，该属性只能是ltr（左到右）或rtl（右到左）

### 语义相关元素

<abbr>:用于表示一个缩写，使用该元素是可以指定如下属性

title:该属性用于指定该所写所代表的全称

<address>:用于表示一个地址，浏览器常会用斜体显示

<blockquote>:用于指定一段长的引用文本，浏览器常用缩进的方式显示，可以定义该属性：

cite:指定该引用文本所引用的网址

<q>:用于定义一段短的引用文本，浏览器会对这段文本加引号

Blackquote和q作用相似，blockquote引用带换行的文本，q引用不带换行的本文

<cite>:用于表示作品的标题，浏览器会用斜体字表示

<code>:用于表示一段计算机代码

<dfn>:用于定义一个专业术语，浏览器用粗体或斜体来表示

<del>:定义删除的文档

<ins>:定义已经被插入到文档中的文本

<del>和<ins>一起使用，描述文档中的更新修正，浏览器通常会在已删除文本上添加一条删除线，在新插入文本下添加一条下划线

cite:规定一个url，该文档解释了文本被插入的原因

detetime：规定文本被插入的日期和时间。

<pre>:预格式化。

### 超链接和锚点

Html5保留了<a>，可以指定id，class，style和事件属性外，还有

href：指定超链接所关联的另一个资源

target：指定使用框架集中的那个框架来装载另一个资源。也保留\_blank,\_self,\_top,\_parent四个属性值

medio：指定目标url所引用的媒体类型，默认为all，只有当制定了href属性时，才生效(此属性时h5新增属性)

锚点与以前一样没有变化

### 列表相关元素

<ul>:定义无序列表，制定了id，class，style等属性，还有事件属性

<ol>:定义有序列表，除了指定和ul相同的属性外，还有2个属性

strat：指定列表项的起始数字

type：指定哪有类项的编号(html5不推荐使用，用css来指定)

<li>:用于定义列表项

<dl>:用于定义列表项，该元素只能包含<dt>和<dd>两种子元素，属性与li相似

<dt>:定义标题列表项，属性同上

<dd>:定义普通列表项，属性和li相似

### 图像相关元素

<img>:单标记元素，可以指定id,class,style的属性，也可以指定时间属性

src:指定图片的路径

alt：指定图片的提示信息

title：指定图片的提示信息

height,width：指定图片的高和宽

usemap：#mapname;

<map name=”mapname”>:用于定义图片映射，该元素包含一个或多个<area>（单标记）子元素

<area>:用于定义图片映射的内部区域，单标记

shape：指定该内部区域的形状：默认是rect矩形，还可以是circle圆形和poly多边形

cords: 定义点击区域的坐标，圆形coords=”x,y,r” 矩形coords=”x1,y1,x2,y2”,左上角和右下角坐标

多边形cords=“x1,y1,x2,y2,x3,y3,…”,坐标表示多边形每一个顶点的坐标

href：资源链接

alt：图片提示信息

### 表格相关元素

<table>:对于caption，thead，tbody，tfoot只能定义0个或1个

保留了cellspacing，cellpadding，width，height

删除了align，bgcolor，border属性

建议cellpadding，cellspacing，width，height在css中设置

<caption>:用于定义表格标题，只能包含文本，图片，超链接和表单控件

<tr>:定义表格的行，有id，style，class属性

<td>:定义单元格，有以下属性

colspan：跨列合并

rowspan：跨行合并

Height和width

Html5删除了align，bgcolor，valign等属性

Height和width建议使用css控制

<col>:该元素用于为某一表格中的一列或多列指定属性值，该元素只能出现在<colgroup>元素内，可以指定id,class,style,onclick等属性，还可以有下面的属性：

span:指定该列可以横跨多少列，值为一个数字

说明：<col>元素是一个单标记，自己本省不会产生列，如果创建列必须依靠<tr>,<col>元素只是为表格中指定列整体指定属性值，因此一旦在table中使用col为表格列指定属性，col定义的列数就应该和表格中实际包含的列数相等。

<colgropu>:该元素用于为表格定义一个或多个列指定的属性值，只能出现在<table>标记中。

<colgroup>:的作用只用于组织多个col元素，当使用<colgroup>组织多个col时，此元素上指定的属性将对齐所包含的所有col元素有效。

## Html5新增的通用属性

contenteditable

当值为true，浏览器允许用户直接编辑html元素内容并且具有可继承性，除非将值改为false(不是所有浏览器都支持)

html5提供了isContentEditable属性，如果可以编辑返回值为true，否则返回值为false

designmode

相当于一个全局contenteditable属性，如果把整个页面的designmode属性设置为on，该页面上所有可支持contenteditable的元素都可以编程编辑状态

默认off

一般用document.designMode=”on”在body开启(用onload事件加载)

hidden

标记当中带有hidden属性的组件都会被隐藏并且不会保留所占用空间同css样式例display：none一样

Spellcheck

Html5为input，textarea增加了此属性

为true是，浏览器会负责对用户输入的文本内容执行输入检查，如果检查不通过，浏览器会对拼写错的单词进行提示

大部分浏览器不支持此属性

## Html5新增常用元素

### 文档结构元素

<article>:用于指定页面上独立、完整的一片文章或区域

内部可以包含以下几个元

<section>:对页面元素进行分块

section和article的区别在于section侧重于文章分块，article侧重于整篇文章

<nav>:用于定义页面上的导航

<aside>:用于定义当前页面或者当前文章的附属信息，推荐aside元素使用css渲染成侧边栏

<header>:主要用于article元素定义文档头部信息，可以包块级或行内元素

<footer>:主要用于<article>元素定义脚注部分，包含该文章的版本信息，作者授权等

<figure>:该元素主要表示一块独立的图片区域，可以包含一个或多个img，包含figcaption元素定义图片标题

<figcaption>:用在figure内部，定义标题。

## Html5新增语义化相关元素

<mark>:用于显示html页面需要重点关注的内容

该元素可以指定id，style，class，hedden等通用属性

浏览器常用背景黄色显示

<time>:用来显示被标注内容，可以是任意内容，包含属性同上，同时还有以下属性

datetime：高属性主要用于向机器提供时间

如果time包含内容本身就满足标准的日期时间格式，可以不用指定datetime属性

<mark>html5新增属性特别多</mark>

<time datetime=”09:00”>早上九点</time>

<time>201609-09</time>

<meter>:表示一个阈值最大值或最小值的技术仪表 (血条)

value:指定技术仪表当前值。默认为0

min:最小值，默认为0

max:最大值，默认为1

low:指定范围的最小值，该值必须大于min

high:指定范围的最大值，该值必须小于max

optimum:指定有效的仪表范围的最佳值

<progress>:表示一个进度条

max:指定进度条完成时的值

value:指定进度条当前完成的进度值

# 第三章 视频音频

## 音频的发展史

早期：<embed>+<object>+文件

问题：不是所有浏览器都支持，而且embed不是标准。

现状：Realplay、window media、Quick Time、Flash

问题：每个厂商每个标准，网站编码和格式也都不相同，flash的出现解决了面的问题，但是apple在07年决定任何设备将不再支持flash。

HTML5认为浏览器应该原生支持音频视频，因为现在也是web中的一等公民了！

## 视频格式的简单介绍

1. 常见的视频格式

视频的组成部分：画面、音频、编码格式

视频编码：H.264、Theora、VP8(google开源)

常见的音频格式：

视频编码:ACC、MP3、Vorbis

## HTML5支持的格式

HTML5能在完全脱离插件的情况下播放音视频

但是不是所有格式都支持

HTML5支持的视频格式：

Ogg = 带有Theora视频编码+Viorbis音频编码的Ogg文件

支持的浏览器:F、C、0

MEPG4=带有H.264视频编码+AAC音频编码的MPEG4文件

支持的浏览器:S、C

WebM=带有VP8视频编码+Vorbis音频编码的WebM格式

支持的浏览器I、F、C、O

<Video>的使用

<video src=”文件地址” controls=”controls”></video>

<video src=”文件地址” controls=”controls”>

您的浏览器暂不支持video标签。播放视频

</video>

<video controls=”controls” width=”300”>

<source src=”move.ogg” type=”video/ogg”>

<source src=”move.mp4” type=”video/mp4”>

您的浏览器暂不支持video标签。播放视频

</video>

## Video的常见属性

|  |  |  |
| --- | --- | --- |
| 属性 | 值 | 描述 |
| Autoplay | Autoplay////不好使了 | 视频就绪自动播放 |
| Controls | Controls | 向用户显示播放控件 |
| Width | Pixels（像素） | 设置播放器宽度 |
| Height | Pixels(像素) | 设置播放器高度 |
| Loop | Loop | 播放完是否继续播放该视频，循环播放 |
| Preload | Proload | 是否等加载完在播放 |
| Src | url | 视频url地址 |
| Poster | Imgurl | 加载等待的画面图片 |
| Autobuffer | Autobuffer | 设为浏览器缓冲方式，不设置autoply |

## Video的API方法

|  |  |  |
| --- | --- | --- |
| 方法 | 属性 | 事件 |
| play() | currentSrc | play |
| pause() | currentTime | pause |
| load() | videoWidth | progress |
| canPlayType | videoHeight | error |

|  |  |  |
| --- | --- | --- |
|  | 全屏 | 退出全屏 |
| Webkit  (Safari5.1/Chrome 15) | element.webkitRequestF  ullScreen() | Document.webkitCancel  FullScreen(); |
| Firfox  (works in nightly) | Element.mozRequestFull  Screen(); | Document.mozCancelFul  lScreen(); |
| W3C建议 | element.requestFullscre  en(); | Document.exitFullscreen |

## Video的API属性

|  |  |
| --- | --- |
| 属性 | 说明 |
| audioTracks | 返回可用的音轨列表(MultipleTrackList对象) |
| autoplay | 媒体加载后自动播放 |
| buffered | 返回缓冲部件的时间范围(TimeRanges) |
| controller | 返回当前的媒体控制器(MediaController) |
| controls | 显示播放控件 |
| crossOrigin | CORS设置 |
| currentSrc | 返回当前媒体的URL |
| currentTime | 当前播放的时间，单位秒(快进快退10秒) |
| defaultMuted | 缺省是否静音 |
| defaultPlayback  Rate | 播放的缺省倍速 |
| duration | 返回媒体的播放总时长，单位秒 |
| ended | 返回当前播放是否结束标志 |
| error | 返回当前播放的错误状态 |
| initialTime | 返回初始播放的位置 |
| loop | 是否循环播放 |
| mediaGroup | 当前音视频所属媒体组(用来连接多个音频标签) |
| muted | 是否静音 |
| networkState | 返回当前网络状态 |
| paused | 是否暂停 |
| playbackRate | 播放的倍数（加速、减速播放） |
| played | 当前播放部件已经播放的时间范围(TimeRanges对象) |
| preload | 页面加载时是否同时加载音视频 |
| readyState | 返回当前的准备状态 |
| seekable | 返回当前的准备状态 |
| seeking | 返回用户是否做了跳转操作 |
| src | 当音视频源的 |
| startOffsetTime | 返回当前的事件便宜(Date对象) |
| textTracks | 返回可用的文本轨迹(TextTrackList对象) |
| videoTracks | 返回可用的视频轨迹(VideoTrackList对象) |
| volume | 音量值 |

## Video的常用事件

|  |  |
| --- | --- |
| 事件 | 描述 |
| abort | 当音频加载被异常终止时产生该事件 |
| canpaly | 当浏览器可以开始播放该音频时产生该事件 |
| canplaythrough | 当浏览器可以播放该音视频到结束而无需因缓冲而停止时产生该事件 |
| durationchange | 当媒体的总时长改变时产生该事件 |
| emptied | 当前播放列表为空时产生该事件 |
| ended | 当前播放列表结束时产生该事件 |
| error | 当加载媒体发生错误时产生该事件 |
| loadeddata | 当加载媒体数据时产生该事件 |
| loadedmetadata | 当收到总时长，分辨率和字轨等metadata时产生该事件 |
| loadstart | 当开始查找媒体数据时产生该事件 |

## HTML5支持的音频格式

HTML5在不使用插件的情况下也可以原生的支持音频格式文件的播放，当然支持格式是有限的

HTML5支持的音频格式：

Ogg 免费支持的浏览器：C、F、O

MP3 收费支持的浏览器：I、C、S

Wav 收费支持的浏览器：F、O、S

## <audio>的使用

<audio src=”文件地址” controls=”controls”></audio>

<audio src=”文件地址” controls=”controls”>

您的浏览器暂不支持audio标签播放视频

</audio>

<audio controls=”controls”>

<source src=”happy.MP3” type=”video/mpeg”>

<source src=”happy.ogg” type=”video/ogg”>

您的浏览器暂不支持audio标签播放视频

</audio>

## audio的常见属性

|  |  |  |
| --- | --- | --- |
| 属性 | 值 | 描述 |
| autoplay | autoplay | 如果出现该属性，则音频在就绪后马上播放。 |
| controls | controls | 如果出现该属性，则向用户显示控件，比如播放按钮 |
| loop | loop | 如果出现该属性，则每当音频结束时重新开始播放。 |
| preload | preload | 如果出现该属性，则音频在页面加载时进行加载，并预备播放。如果使用”autoplay,则忽略该属性。” |
| src | url | 要播放的音频的URL。 |

# 第四章 html5表单

## HTML5表单元素

<label>标签为input元素定义标注

Label元素不会向用户呈现任何特殊效果，不过，他为鼠标用户改进了可用性，如果您在label元素内点击文本，就会触发此控件。就是说，当前用户选择该标签时，浏览器就会自动将焦点转到标签相关的表单控件上，

<label>标签的for属性应当与相关元素的id属性相同。

<button>标签定义一个按钮

在<button>元素内部。您可以防止内容，比如文本或图像，这是该元素与使用<input>元素创建的按钮之间的不同之处。

始终为<button>元素规定type属性，不同的浏览器对<button>元素的type属性使用不同的默认值。

如果在HTML表单中使用<button>元素，不同的浏览器可能会提交不同的按钮值，请使用<input>在HTML表单中创建按钮。

<optgroup>标签把相关的选项组和在一起，来分组<select>标记中的<opction>选项

属性label为选项组规定描述(组名)

<datalist>标签规定了<input>元素可能的选项列表。

<datalist>标签被用来在位<input>元素提供“自动完成”的特性。

选项时预先定义好的。将作为用户的输入数据。

使用<input>元素的list属性来绑定<datalist>元素。

<input list=”browsers”>

<datalist id=”browsers”>

<opction value=”Internet Explorer”>

<opction value=”Firefox”>

<opction value=”Chrome”>

<opction value=”Opera”>

<opction value=”Safan”>

</datalist>

## 表单属性

placeholder属性提供可描述输入字段预期值的提示信息。

该提示会在输入字段为空时显示，并会在字段获得焦点时消失

Placeholder属性适用于一下的<input>类型：text,serch,url,telephone,email以及password.

### THML5 Input类型

|  |  |
| --- | --- |
| color | 用于选取颜色 |
| date | 允许用户从一个日期选择器中选择-一个日期，使用date类型，会自动出现一个日期选择器让用户选择 |
| datetime | 允许用户选择-一个日期和时间(utc时间) |
| datetime-local | 允许用户选择-一个时间和日期(不包含时区) |
| month | 允许用户选择-一个年份和月份 |
| time | 允许选择--个时间 |
| week | 选择年和第几周 |
| email | 用于输入e-mail地址，并且提交表单是会自动验证e-mail地址是否合法 |
| tel | 定义电话号码的输入框，与--般文本框一-样，没有自动验证 |
| number | 控制用户输入数字的区间，可以设置最大和和最小值,属性min和max |
| range | 显示一个滑块用于，包含一定分范围内的数字，进行选择,-般用于不需要很精确的选择属性min,max,value,step(数字间隔) |
| search | 搜索域，定义一个搜索字段，显示效果与text类型-样 |
| url | url类型用于输入url地址，提交表单时会进行验证，是否符合url规则(必须是完整的url才可以)。 |

### form标记的新增属性:

autocomplete=”on|off”（不好使）

作用：设置是否让浏览器自动记录之前输入过的值，这个属性属于form标记，同时也是属于input标记的

novalidate属性

作用:就是在表单提交时不在自动进行验证。

### Input新增的属性

|  |  |
| --- | --- |
| autofocus | 自动获得焦点 |
| placeholder | 提示信息 |
| required | 该属性用来验证，表单项中是否为空，有空的就提示并且不允许提交 |
| form | 使用form="id"的方式将表单项指向相同id的form表单 |
| multiple | 使用该属性允许用户输入选择多个值，使用的类型为type="file|email"同时上传多个文件，或写多个邮箱地址 |
| pattern | 使用方法pattern="正则表达式”，如果不符合规则，会自动提示，并且不会提交表单 |

## FileReader API

■API (Application Programming Interface,应用程序编程接口)是一些预先定义的函数，目的是提供应用程序与. 开发人员基于某软件或硬件得以访问--组例程的能力，而又无需访问源码，或理解内部工作机制的细节。

■HTML .5定义了FileReader对象作为文件API的重要成员用于读取文件，根据W3C的定义，FileReader接口提供了读取文件的方法和包含读取结果的事件模型。FileReader的使用方式非常简单，创建FileReader对象并调用其 方法即可

■FileReader的实例拥有4个方法，其中3个用以读取文件，另一个用来中断读取。下 面的表格列出 了这些方法以及他们的参数和功能，需要注意的是，无论读取成功或失败，方法并不会返回读取结果，这一结果存储在result属性中

|  |  |  |
| --- | --- | --- |
| 方法名 | 参数 | 描述 |
| abort | none | 中断读取 |
| readAsBinaryString | file | 将文件读取为二进制码 |
| readAsDataURL | file | 将文件读取为DataURL |
| readAsText | file | 将文件读取为文本 |

filse获取的是两个数组

处理事件

FileReader包含了一套完整的事件模型，用于捕获读取文件时的状态，下面这个表格归纳了这些事件。

文件一旦开始读取，无论成功或失败，实例的result属性都会被填充。如果读取失败，则result的值位null，否则即是读取的结果，绝大多数的程序都会在成功读取文件的时候，抓取这个值。

|  |  |
| --- | --- |
| 事件 | 描述 |
| onabort | 中断时触发 |
| onerror | 出错时触发 |
| onload | 文件读取成功完成时触发 |
| onloadend | 读取完成触发，无论成功或失败 |
| onloadstart | 读取开始时触发 |
| onprogress | 读取中 |

**上传图片文件**

**document . getElementById("file" ) . onchange=function(){**

**var source=this. files[0];**

**if(window. FileReader){**

**var fr=new FileReader( );**

**fr . readAsDataURL (source);**

**fr . onload=function(e){**

**e = ell window . event ;**

**document getE lementById("im" ).src=e . target . result;**

**console. log("文件显示成功" );**

}