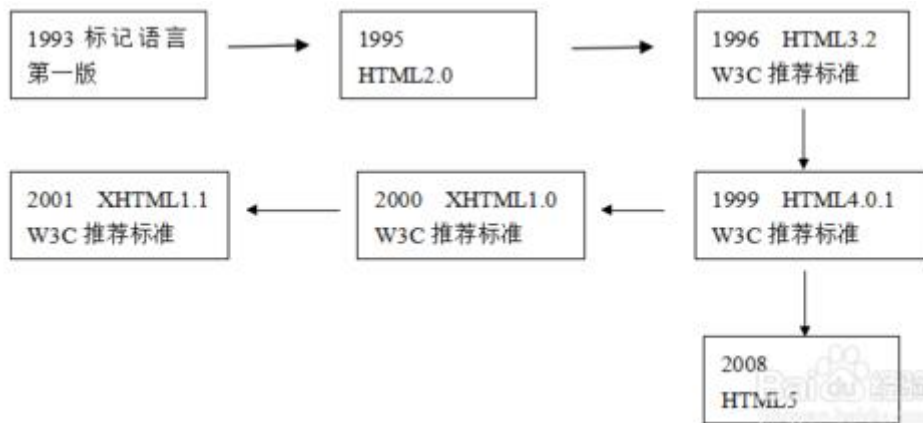


# HTML5新标签与特性



## 1.字符设定

`<meta http-equiv="charset" content="utf-8">`: HTML与XHTML中建议这样去写  
`<meta charset="utf-8">`: HTML5的标签中建议这样去写

## 2.常用新标签

w3c 手册中文官网 : <http://w3school.com.cn/>

- header: 定义文档的页眉 头部
- nav: 定义导航链接的部分
- footer: 定义文档或节的页脚 底部
- article: 定义文章。
- section: 定义文档中的节 (section、区段)
- aside: 定义其所处内容之外的内容 侧边

```
<header> 语义 :定义页面的头部 页眉</header>
<nav> 语义 :定义导航栏 </nav>
<footer> 语义: 定义 页面底部 页脚</footer>
<article> 语义: 定义文章</article>
<section> 语义: 定义区域</section>
<aside> 语义: 定义其所处内容之外的内容 侧边</aside>
```

- datalist 标签定义选项列表。请与 input 元素配合使用该元素

```
<input type="text" value="输入明星" list="star"/> <!-- input里面用 list -->
<datalist id="star"> <!-- datalist 里面用 id 来实现和 input 链接 -->
    <option>刘德华</option>
    <option>刘若英</option>
    <option>刘晓庆</option>
    <option>郭富城</option>
    <option>张学友</option>
    <option>郭郭</option>
</datalist>
```

- fieldset 元素可将表单内的相关元素分组    legend 搭配使用

```
<fieldset>
    <legend>用户登录</legend>    标题
    用户名: <input type="text"><br /><br />
    密 码: <input type="password">
</fieldset>
```

### 3.新增的input type属性值:

类型****	使用示例****	含义****
email****	<input type="email"/>	输入邮箱格式
tel****	<input type="tel"/>	输入手机号码格式
url****	<input type="url"/>	输入url格式
number****	<input type="number"/>	输入数字格式
search****	<input type="search"/>	搜索框（体现语义化）
range****	<input type="range"/>	自由拖动滑块
time****	<input type="time"/>	小时分钟
date****	<input type="date"/>	年月日
datetime****	<input type="datetime"/>	时间
month****	<input type="month"/>	月年
week****	<input type="week"/>	星期 年
color	<input type="color"/>	颜色

##

### 4.常用新属性

属性****	用法****	含义****
placeholder****	<input type="text" value="请输入用户名"/>	占位符 当用户输入的时候 里面的文字消失 删除所有文字，自动返回
autofocus****	<input type="text"/>	规定当页面加载时 input 元素应该自动获得焦点
multiple****	<input type="button" value="选择文件"/> 未选择任何文件	多文件上传
autocomplete****	<input type="text"/>	规定表单是否应该启用自动完成功能 有2个值，一个是on 一个是off on 代表记录已经输入的值 1.autocomplete 首先需要提交按钮 2.这个表单您必须给他名字
accesskey****	<input type="text"/>	规定激活（使元素获得焦点）元素的快捷键 采用 alt + s的形式

## 5.综合案例

```

<form action="">
  <fieldset>
    <legend>学生档案</legend>
    <label for="userName">姓名:</label>
    <input type="text" name="userName" id="userName" placeholder="请输入用户名">
  <br>
    <label for="userPhone">手机号码:</label>
    <input type="tel" name="userPhone" id="userPhone" pattern="^\d{10}$"><br>
    <label for="email">邮箱地址:</label>
    <input type="email" required name="email" id="email"><br>
    <label for="collage">所属学院:</label>
    <input type="text" name="collage" id="collage" list="cList" placeholder="请选择"><br>
    <datalist id="cList">
      <option value="前端与移动开发学院"></option>
      <option value="java学院"></option>
      <option value="c++学院"></option>
    </datalist><br>
    <label for="score">入学成绩:</label>
    <input type="number" max="100" min="0" value="0" id="score"><br>
  <form action="">
    <fieldset>
      <legend>学生档案思密达</legend>
      <label>姓名: <input type="text" placeholder="请输入学生名字"/></label> <br />
      <label>手机号: <input type="tel" /></label> <br /><br />

```

```

<label>邮箱: <input type="email" /></label> <br /><br />
<label>所属学院: <input type="text" placeholder="请选择学院"
list="xueyuan"/>
  <datalist id="xueyuan">
    <option>java学院</option>
    <option>前端学院</option>
    <option>php学院</option>
    <option>设计学院</option>
  </datalist>

<br /><br />

<label>出生日期: <input type="date" /></label> <br /><br />
<label>成绩: <input type="number" /></label> <br /><br />
<label>毕业时间: <input type="date" /></label> <br /><br />
<input type="submit" /> <input type="reset" />
</fieldset>
</form>
<label for="inTime">入学日期:</label>
<input type="date" id="inTime" name="inTime"><br>
<label for="leaveTime">毕业日期:</label>
<input type="date" id="leaveTime" name="leaveTime"><br>
<input type="submit">
</fieldset>
</form>

```

## 6.多媒体标签

- audio: 播放音频
- video: 播放视频

### 6.1多媒体 audio

HTML5通过标签来解决音频播放的问题。

使用相当简单，如下图所示

```

<!-- 通过src指定音频文件路径即可 -->
<audio src="./music/See You Again.mp3"></audio>

```

并且可以通过附加属性可以更友好控制音频的播放，如：

autoplay 自动播放

controls 是否显不默认播放控件

loop 循环播放 不设置就不会循环播放

由于版权等原因，不同的浏览器可支持播放的格式是不一样的，如下图供参考

当前，audio 元素支持三种音频格式：

	IE 9	Firefox 3.5	Opera 10.5	Chrome 3.0	Safari 3.0
Ogg Vorbis		√	√	√	
MP3	√			√	√
Wav		√	√		√

多浏览器支持的方案，如下图

```
<audio controls>
  <!-- 通过source标签指定多格式音频文件 -->
  <source src="./music/See You Again.mp3">
  <source src="./music/See You Again.wav">
  <source src="./music/See You Again.ogg">
  您的浏览器不支持HTML音频播放功能
</audio>
```

## 6.2多媒体 video

HTML5通过标签来解决音频播放的问题。

同音频播放一样，

使用也相当简单，如下图

```
<!-- 通过src属性指定视频文件的路径 -->
<video src="./video/movie.mp4" controls="controls"></video>
```

同样，通过附加属性可以更友好的控制视频的播放

autoplay 自动播放

controls 是否显示默认播放控件

loop 循环播放

width 设置播放窗口宽度

height 设置播放窗口的高度

由于版权等原因，不同的浏览器可支持播放的格式是不一样的，如下图供参考

当前，video 元素支持三种视频格式：

格式	IE	Firefox	Opera	Chrome	Safari
Ogg	No	3.5+	10.5+	5.0+	No
MPEG 4	9.0+	No	No	5.0+	3.0+
WebM	No	4.0+	10.6+	6.0+	No

## H5新增API

### 1 网络状态

我们可以通过window.navigator.onLine来检测，用户当前的网络状况，返回一个布尔值，但是不同浏览器会存在差异

所以HTML5 给我们提供了2个事件 online 和 offline

online用户网络连接时被调用

offline用户网络断开时被调用

他们监听的对象都是window

### 2 全屏

HTML5规范允许用户自定义网页上任一元素全屏显示。

1、Node.requestFullScreen() 开启全屏显示

2、Node.cancelFullScreen() 关闭全屏显示

由于其兼容性原因，不同浏览器需要添加前缀如：

webkit内核浏览器：webkitRequestFullScreen、webkitCancelFullScreen，如chrome浏览器。

Gecko内核浏览器：mozRequestFullScreen、mozCancelFullScreen，如火狐浏览器。

ms 微软 msRequestFullscreen

3、document.fullScreen检测当前是否处于全屏

不同浏览器需要添加前缀

document.webkitIsFullScreen、document.mozFullScreen

全屏伪类选择器 也有兼容性问题 需要添加前缀思密达

:full-screen 、:-webkit-full-screen {}、:moz-full-screen {}

### 3 文件读取

我们想： 可以吧上传的文件，内容显示到页面？ 或者 上传完毕图片显示缩略图到页面上。。。

通过FileReader对象我们可以读取本地存储的文件，使用 [File](#) 对象来指定所要读取的文件或数据。其中 File对象可以是来自用户在一个 元素上选择文件后返回的[FileList](#) 对象，也可以来自自由拖放操作生成的 [DataTransfer](#)

## 3.1 Files对象

由于HTML5中我们可以通过为表单元素添加multiple属性，因此我们通过上传文件后得到的是一个Files对象（伪数组形式）。

## 3.2 FileReader对象

HTML5新增内建对象，可以读取本地文件内容。

var reader = new FileReader; 可以实例化一个对象

实例方法

1、[readAsDataURL\(\)](#) 以DataURL形式读取文件

事件监听

onload 当文读取完成时调用

属性

result 文件读取结果

[参考资料](#)

<https://developer.mozilla.org/zh-CN/docs/Web/API/FileReader>

## 4 地理定位

在HTML规范中，增加了获取用户地理信息的API，这样使得我们可以基于用户位置开发互联网应用，即基于位置服务 (Location Base Service)

### 4.1 API详解

1、获取当前地理信息

navigator.geolocation.getCurrentPosition(successCallback, errorCallback)

2、重复获取当前地理信息

navigator.geolocation.watchPosition(successCallback, errorCallback)

当成功获取地理信息后，会调用succssCallback，并返回一个包含位置信息的对象position。

position.coords.latitude 纬度

position.coords.longitude经度

position.coords.accuracy精度

position.coords.altitude海拔高度

当获取地理信息失败后，会调用errorCallback，并返回错误信息error

[http://www.w3school.com.cn/html5/html\\_5\\_geolocation.asp](http://www.w3school.com.cn/html5/html_5_geolocation.asp)

## 4.2 应用

在现实开发中，通过调用第三方API（如百度地图）来实现地理定位信息，这些API都是基于用户当前位置的，并将用位置位置（经/纬度）当做参数传递，就可以实现相应的功能。

<http://lbsyun.baidu.com/> 百度地图api

## 5 拖拽

---

在 HTML5 中，拖放是标准的一部分，任何元素都能够拖放。

### 5.1 拖拽和释放

拖拽：Drag

释放：Drop

拖拽指的是鼠标点击源对象后一直移动对象不松手，一但松手即释放了

### 5.2 设置元素为可拖放

draggable 属性：就是标签元素要设置draggable=true，否则不会有效果

注意： 链接和图片默认是可拖动的，不需要 draggable 属性。

### 5.2 拖拽API的相关事件

**被拖动的源对象可以触发的事件：**

- (1)ondragstart：源对象开始被拖动
- (2)ondrag：源对象被拖动过程中(鼠标可能在移动也可能未移动)
- (3)ondragend：源对象被拖动结束

**拖动源对象可以进入到上方的目标对象可以触发的事件：**

- (1)ondragenter：目标对象被源对象拖动进入
- (2)ondragover：目标对象被源对象拖动悬停在上方
- (3)ondragleave：源对象拖动离开了目标对象
- (4)ondrop：源对象拖动在目标对象上方释放/松手

拖拽API总共就是7个函数！！

### 5.3 DataTransfer

在进行拖放操作时，`DataTransfer` 对象用来保存被拖动的数据。它可以保存一项或多项数据、一种或者多种数据类型

## 6 Web存储

---



随着互联网的快速发展，基于网页的应用越来越普遍，同时也变的越来越复杂，为了满足各种各样的需求，会经常性在本地存储大量的数据，HTML5规范提出了相关解决方案。

## 6.1 特性

- 1、设置、读取方便、页面刷新不丢失数据
- 2、容量较大，sessionStorage约5M、localStorage约20M
- 4、只能存储字符串，可以将对象JSON.stringify() 编码后存储

## 6.2 window.sessionStorage

- 1、生命周期为关闭浏览器窗口
- 2、在同一个窗口(页面)下数据可以共享

## 6.3 window.localStorage

- 1、永久生效，除非手动删除 关闭页面也会存在
- 2、可以多窗口（页面）共享（同一浏览器可以共享）

## 6.4 方法详解

setItem(key, value) 设置存储内容

getItem(key) 读取存储内容

removeItem(key) 删除键值为key的存储内容

clear() 清空所有存储内容

## 6.5 其它

WebSQL、IndexedDB

# 7 应用缓存

---

HTML5中我们可以轻松的构建一个离线（无网络状态）应用，只需要创建一个cache manifest文件。

## 7.1 优势

- 1、可配置需要缓存的资源
- 2、网络无连接应用仍可用
- 3、本地读取缓存资源，提升访问速度，增强用户体验
- 4、减少请求，缓解服务器负担

## 7.2 缓存清单

一个普通文本文件，其中列出了浏览器应缓存以供离线访问的资源，推荐使用.appcache为后缀名

例如我们创建了一个名为demo.appcache的文件，然后在需要应用缓存在页面的根元素(html)添加属性manifest="demo.appcache"，路径要保证正确。

## 7.3 manifest文件格式\*

- 1、顶行写CACHE MANIFEST
- 2、CACHE: 换行 指定我们需要缓存的静态资源，如.css、image、js等
- 3、NETWORK: 换行 指定需要在线访问的资源，可使用通配符
- 4、FALLBACK: 换行 当被缓存的文件找不到时的备用资源

## 7.4 其它

- 1、CACHE: 可以省略，这种情况下将需要缓存的资源写在CACHE MANIFEST
- 2、可以指定多个CACHE: NETWORK: FALLBACK:，无顺序限制
- 3、#表示注释，只有当demo.appcache文件内容发生改变时或者手动清除缓存后，才会重新缓存。
- 4、chrome 可以通过chrome://appcache-internals/工具和离线（offline）模式来调试管理应用缓存