

简介:

标签

语义标签

表单标签新增内容

进度条标签

度量器标签

<audio>标签

<video>标签

DOM操作

新增获取元素的方法

样式操作方法

自定义属性

网络接口

全屏接口

拖拽接口

地理定位接口

Web存储

应用程序缓存

history对象

**简介:**

- 1.HTML5是下一代的HTML
- 2.2014年正式公布
- 3.为移动端而生

4.新特性应该基于HTML、CSS、DOM及Javascript

5.减少外部插件的需求

6.更多取代脚本的标记

分析：

1.支持所有主流浏览器，IE9及以上支持（有选择的支持，但不是全部支持），但不支持IE8及以下。

2.改变了用户与文档的交互方式：多媒体：video audio canvas

3.增加了其他新的特性：语义特性，本地存储特性，网页多媒体，二维三维，特效（过渡，动画）

4.相对于h4：

i.进步：抛弃了一些不合理不常用的标记和属性

ii.新增了一些编辑和属性--表单

iii.从代码角度而言，h5的网页结构代码更加简洁。

## 标签

### 语义标签

<header>

<nav>

<main>

<article>

<aside>

<footer>

兼容问题：

1.兼容主流浏览器；

2.IE9认识这些语义标签，但默认这些元素为行内块，需要display: block

3.IE8及以下根本不认识这些标签，需要通过JS createElement手动创建这些元素，但JS创建的元素默认为行内元素，性质同span，需要转换成块级元素。兼容html5插件的使用。

### 表单标签新增内容

#### 新增type属性

email：提供了默认的电子邮箱验证，要求必须包含@符号，同时必须包含服务器名称，如果不能满足验证，则会组织当前的数据提交

tel：不是来实现验证。它的本质目的是为了能够在移动端打开数字键盘。意味着数字键盘限制了用户只能输入数字。

url：验证只能输入合法的网址：必须包含http://

search：提供更人性化的体验，输入商品后提供小叉叉删除文本

number: 只能输入数字（含小数点），不能输入其他的字符。可以搭配max/min/value属性。value指默认值。

range: 范围。与number一样，可以搭配max/min/value属性

color: 颜色。可以打开选色块

time: 时间：时分秒

date: 日期：年月日

datetime: 大多数浏览器不支持datetime，只能苹果下面的safari支持

datetime-local: 日期和时间

month: 月份

week: 星期

### **新增其他属性**

placeholder: 提示文本，提示占位；

autofocus: 自动获取焦点

autocomplete: 自动完成: on: 打开, off: 关闭。成功使用前提: 1.必须成功提交过, 2.当前体检autocomplete的元素必须由name属性。

required: 必须输入，如果没有输入则会停止当前数据提交

pattern: 正则表达式验证

multiple: 可以选择多个文件，在email中允许输入多个邮箱，用逗号分隔。

form: 指定表单id，那么在将来指定的id的表单进行数据提交的时候，也会将当前表单数据一起提交。ps: 这个是form属性。当我们的页面中包含几个form时，第一个form表单提交信息不会获取到第二个表单中信息。form= "myID"

### **新增的表单元素**

#### **datalist**

实现表单不仅可以选下拉菜单，还可以输入（既有select元素功能，又有功能）。

datalist: 创建选项列表。子元素仍然通过option标签创建。

option的属性中value指具体的值，label: 提示信息，辅助作用。

input标签和datalist创建关联。datalist中创建id属性，input中创建list属性等于datalist属性名

#### **Keygen**

密钥生成器。当提交表单时，会生成两个键，一个是私钥，一个公钥。私钥存储于客户端，公钥则被发送到服务器。公钥可用于之后验证用户的客户端证书，即实现非对称加密。目前浏览器支持率极低。

#### **output**

新的语义标签：用来放置输出内容的，但是不能自动的计算结果，依旧需要通过js的方式来动态修改结果，只是相比于其他的标签，语义性更强。

### 新增表单事件

oninput：监听当前指定元素内容的改变：只要内容改变（添加内容、删除内容），就会触发这个事件

onkeyup：键盘弹起的时候触发：每个键只要弹起依次就会触发一次。

oninvalid：当验证不通过时出发。设置默认的提示信息this.setCustomValidity()

### 进度条标签

`<progress></progress>`

进度条属性：value 当前按进度值； max：最大值； min：最小值

### 度量器标签

`<meter></meter>`

衡量当前的进度值

度量器属性：

high：规定的较高的值

low：规定的较低的值

max：最大值

min：最小值

value：当前度量值

### `<audio>` 标签

属性：

1.src：播放文件的路径

2.controls：必须，音频播放器的控制面板

3.autoplay：自动播放

4.loop：循环

### `<video>` 标签

支持格式：Ogg、MPEG4、WebM。Safari 浏览器只识别WebM格式

属性：

1.单独设置了width、height属性；注意：视频始终会保持原始的宽高比。意味着如果你同时设置宽高，并不是真正地将视频的画面大小设置为指定的大小，而只是将视频的占据区域设置为指定大小，除非你设置的狂傲比例正好是原始视频的宽高比例。所以建议：在设置受聘的宽高时，一般只会设置宽度或者高度。

2.src：要播放的视频的 URL。source元素类似功能。video 元素允许多个 source 元素。

source 元素可以链接不同的视频文件。浏览器将使用第一个可识别的格式

3.autoplay：视频在就绪后马上播放。

4.controls: 向用户显示控件, 比如播放按钮。

5.loop: 循环播放。

6.perload: 视频在页面加载时进行加载, 并预备播放。

7.poster: 指定视频还没有完全下载完毕, 或者用户没有点击播放前显示的封面。默认显示当前视频文件的第一帧图像。

8.source元素: 因为不同的浏览器支持的视频格式不一样, 为了保证用户能够看到视频, 我们可以提供多个视频未见让浏览器自当选择

### DOM控制: 方法、属性、事件

[http://www.w3school.com.cn/html5/html5\\_ref\\_audio\\_video\\_dom.asp](http://www.w3school.com.cn/html5/html5_ref_audio_video_dom.asp)

方法	属性	事件
play()	currentSrc	play
pause()	currentTime	pause
load()	videoWidth	progress
canPlayType	videoHeight	error
	duration	timeupdate
	ended	ended
	error	abort
	paused	empty
	muted	emptied
	seeking	waiting
	volume	loadedmetadata
	height	
	width	

## DOM操作

索引: 通过索引获取元素, 不直观。以后的数据都是从后台动态获取, 前端动态生成添加。

### 新增获取元素的方法

querySelector

querySelectorAll

详情在js笔记中有

### 样式操作方法

classList.add添加类。只能添加一个类。与className方法比较, 更加方便。

classList.remove删除类

classList.toggle切换类

classList.contains: 判断类, 返回false/true

classList.item(1): 获取样式, 类名。

### 自定义属性

标签中添加自定义属性规范:

1.data-开头

2.data-后必须至少有一个字符，多个单词使用-链接

3.eg:data-school-name="itcast "

建议：

1.名称应该使用小写--不要包含任何大写字符；

2.名称中不要有任何特殊符号；

3.名称不要使用纯数字

JS中获取自定义属性值

1.在js中

2.使用dataset["schoolName"]

3.将data-后面的单词使用camel命名法链接，否则有可能无法取到值

## 网络接口

ononline：网络连通的时候出发事件

onoffline：网络断开时触发事件

## 全屏接口

不同浏览器需要添加不同的前缀：chrome-webkit firefox: moz ie: ms opera: o

使用能力测试添加不同浏览器的前缀。

1.requestFullScreen ()：开启全屏显示

2.cancelFullScreen ()：退出全屏显示

3.fullScreenElement：是否是全屏状态

4.退出全屏和是否全屏状态都只能使用document来实现。

## FileReader

作用：读取文件内容

1.readAsText()：读取文本文件（可以使用TXT打开的文件），返回文本字符串，默认编码是UTF-8；

2.readAsBinaryString()：读取任意类型的文件。返回二进制字符串。这个方法不是用来读取文件展示给用户看，而是存储文件。例如：读物文件的内容，获取二进制数据，传递给后台，后台接受数据之后，再将数据存储

3.readAsDataURL()：读取文件获取一段以data开头的字符串，这段字符串的本质就是DataURL。DataURL是一种将文件（这个文件一般就是指图像或者能够嵌入到文档的文件格式）嵌入到文档的方案。DataURL是将资源转换为base64编码的字符串形式，并且将这些内容直接存储在url中，优化网站的加载速度和执行效率。

4.abort()：中断读取。

## 拖拽接口

拖拽属性

在h5中，如果想拖拽元素，就必须为元素添加draggable="true",图片和超链接默认可以拖拽。

### **拖拽事件**

#### **应用于被拖拽元素的事件**

ondragstart: 应用于拖拽元素，当拖拽开始时调用；

ondrag: 应用于拖拽元素，整个拖拽过程都会调用，持续；

ondragend: 应用于拖拽元素，当拖拽结束时调用

ondragleave: 应用于拖拽元素，当鼠标离开拖拽元素时调用；

#### **应用于目标元素的事件**

ongragerter: 应用于目标元素，当拖拽元素进入时调用

ondragover: 应用于目标元素，当停留在目标元素上时调用

ondrop: 应用于目标元素，当在目标元素上松开鼠标时调用

ondragleave: 应用于目标元素，当鼠标离开目标元素时调用

#### **拖拽事件对象DragEvent**

e.target, 事件对象目标

e.preventDefault();\*\*\*浏览器默认会阻止ondrop事件：我们必须在ondragover中阻止浏览器的默认行为

e.dataTransfer:

e.dataTransfer.setData("text/html",e.target.id)

setData(format,data);

format: 数据的类型: text/html text/url-list

data: 数据，一般来说是字符串类型

getData(format);

通过setData存储的数据，只能在ondrop事件中获取。

### **全局变量的缺点:**

(1) 全局变量保存在静态存储区，程序开始运行时为其分配内存，程序结束释放该内存。与局部变量的动态分配、动态释放相比，生存期比较长，因此过多的全局变量会占用较多的内存单元。

(2) 全局变量破坏了函数的封装性能。函数象一个黑匣子，一般是通过函数参数和返回值进行输入输出，函数内部实现相对独立。但函数中如果使用了全局变量，那么函数体内的语句就可以绕过函数参数和返回值进行存取，这种情况破坏了函数的独立性，使函数对全局变量产生依赖。同时，也降低了该函数的可移植性。

(3) 全局变量使函数的代码可读性降低。由于多个函数都可能使用全局变量，函数执行时

全局变量的值可能随时发生变化，对于程序的查错和调试都非常不利。

因此，如果不是万不得已，最好不要使用全局变量。

## 地理定位接口

Darf2LCCGzn6T16zgy8ZPkvYYESCT6fu

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尾椎开发互联网应用，即基于位置服务。

### 获取位置的方式

#### IP地址

优点：任何地方都可用，在服务器端处理

缺点：不精确（经常出错，一般精确到城市级），运算代价大

#### GPS

优点：很精确

缺点：定位时间长，耗电大；室内效果差；需要额外引荐设备支持

#### Wi-Fi

优点：精确；可在室内使用；简单、快捷

缺点：在乡间这些WiFi接入点少的地区无法使用

#### 手机信号

优点：相当准确；可在室内使用；简单、快捷

缺点：需要能够访问手机或其modem设备

#### 用户自定义

优点：可获得比程序定位服务更准确的位置数据；用户自行输入可能比自动检测更快

缺点：可能很不准确，特别是用户位置变更后

安全性：获取地理位置属于私密信息。所以浏览器会弹出提示框，在征得用户允许的情况下再获取地理位置。

## 语法

`navigator.geolocation.getCurrentPosition(successCallback,errorCallback,options)`获取当前地理信息

参数1：获取地理信息成功之后的回调；

参数2：获取地理信息失败之后的回调；

参数3：获取当前地理信息的方式；可以设置获取数据的方式

`enableHighAccuracy`: true/false: 是否使用高精度

`timeout`: 设置超时时间，单位ms

`maximumAge`: 可以设置浏览器重新获取地理信息的事件间隔。

`navigator.watchPosition(successCallback,errorCallback,options)`重复获取当前地理信息



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

```
navigator.geolocation.getCurrentPosition(function(position){  
    position.coords.latitude //纬度  
    position.coords.longitude //经度度  
    position.coords.accuracy //精度  
    position.coords.altitude //海拔高度  
},function(error){  
    error;//错误对象  
},)
```

## Web存储

### sessionStorage

- 1.存储数据到本地。存储的容量5mb左右.
- 2.这个数据本质是存储到当前页面的内存中；查看在F12资源中查看
- 3.它的生命周期为关闭当前页面。关闭页面，数据会自动清除。

setItem(key,value):

以键值对的方式存储数据

键名相同会覆盖；

getItem(key):

获取数据，通过制定名称的key获取对应的value值。

如果找不到对应名称的key，name就会获取null

获取到的是字符串 JSON.parse：字符串转换成JSON。JSON.stringify：JSON转换成字符串

removeItem(key):

删除数据，通过制定名称的key删除对应的值

在删除的时候如果key值错误，不会报错，但是也不会删除数据

clear(): 清空所有存储的内容

### localStorage

- 1.存储的数据更多，20mb左右
- 2.不同浏览器不能共享数据，但是在同一个浏览器的不同窗口中可以共享数据
- 3.永久生效，它的数据是存储在硬盘，并不会随着页面或浏览器的关闭而清除。如果要清除，必须手动清除。

方法用法同sessionStorage相同。

## 应用程序缓存

使用HTML5，通过创建cache manifest文件，可以轻松地创建web应用的离线版本

### 优势

- 1.可配置需要缓存的资源，而浏览器缓存要么不缓存，要么你全部缓存，html5缓存更人性化；
- 2.网络五连接应用认可用
- 3.本地读取缓存资源，提升访问速度，增强用户体验
- 4.减少清酒，缓解服务器负担。

### Cache Manifest基础

- 1.如需启动应用程序缓存，请在文档的<html>标签中包含manifest属性：

```
<html manifest="demo.appcache">
```

manifest=“应用程序缓存清单文件的路径”，建议文件的扩展名是appcache。这个文件的本质就是一个文本文件。

- 2.每个指定了manifest的页面在yoghurt对齐访问时都会被缓存。如果未指定manifest属性，则页面不会被缓存（除非在manifest文件中直接指定了该页面）。

- 3.注意：manifest文件需要配置正确的MIME-type，即“text/cache-manifest”，必须在web服务器上进行配置。

### Manifest文件

- 1.第一句必须是：CACHE MANIFEST

- 2.#后面写注释

- 3.需要缓存的文件清单列表：

CACHE:

../images/1.jpg

\*代表所有文件需要缓存

- 4.配置每一次需要重新从服务器获取的文件清单列表：

NETWORK:

../images/1.jpg

- 5.配置如果文件无法获取则使用指定的文件进行替代。

FALLBACK:

../images/1.jpg ../images/2.jpg

/代表所有无法获取的文件使用指定文件代替

### history对象

回退：window.history.back()

前进：window.history.forward()

刷新：window.history.go()

