# html5 基础

[1. html5 基础](#_Toc30375)

[1.1. HTML5 简单介绍](#_Toc6119)

[1.1.1. 认识html5](#_Toc23328)

[1.1.2. 语法规范](#_Toc10547)

[1.2. HTML5语义标签的介绍](#_Toc4530)

[1.2.1. 语义化标签的概念](#_Toc21058)

[1.2.2. 常见语义化标签](#_Toc3038)

[1.2.3. 新语义标签的的区别](#_Toc28881)

[1.2.4. 兼容处理](#_Toc16783)

[1.3. HTML5 表单新增的类型](#_Toc29122)

[1.3.1. 新增的类型](#_Toc22099)

[1.4. HTML5 新增的属性介绍](#_Toc29649)

[1.5. 多媒体标记](#_Toc31159)

[1.5.1. 音频](#_Toc19726)

[1.5.2. 视频](#_Toc25438)

[1.6. HTML5 自定义属性](#_Toc28960)

[1.7. HTML5 新增dom 操作API](#_Toc4278)

[1.7.1. 获取元素](#_Toc26262)

[1.7.2. 类名操作](#_Toc22910)

[1.7.3. Tab 选项卡案例](#_Toc11224)

[1.8. 定位的API 以及调用百度地图的使用](#_Toc22921)

[1.8.1. 获取地理信息方式](#_Toc12396)

[1.8.2. 隐私](#_Toc24909)

[1.8.3. API详解](#_Toc28086)

[1.9. HTML5 本地存储SessionStorage与localStorage 的使用](#_Toc27722)

[1.9.1. 存储特性](#_Toc16982)

[1.9.2. sessionStroage 的使用](#_Toc21323)

[1.9.3. localStorage 的使用](#_Toc3534)

[1.9.4. 方法详细介绍](#_Toc25103)

[1.9.5. 换肤案例](#_Toc13857)

## HTML5 简单介绍

### 认识html5

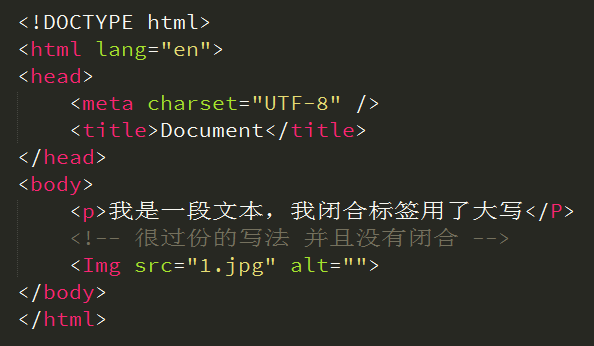
HTML5并不仅仅只是做为HTML标记语言的一个最新版本，更重要的是它制定了Web应用开发的一系列标准，成为第一个将Web做为应用开发平台的HTML语言。

HTML5定义了一系列新元素，如新语义标签、智能表单、多媒体标签等，可以帮助开发者创建富互联网应用，还提供了一些Javascript API，如地理定位、重力感应、硬件访问等，可以在浏览器内实现类原生应用，甚至结合Canvas我们可开发网页版游戏。

H5范称=HTML + CSS3 + JS

### 语法规范

HTML5在语法规范上也做了比较大的调整，去除了许多冗余的内容，书写规则更加简洁、清晰。见代码实例



**特点：**

1、更简洁

2、更宽松

实际开发中应规范书写，不建议太随意 ！

**W3C验证地址**

<https://validator.w3.org/>

## HTML5语义标签的介绍

### 语义化标签的概念

传统的做法我们或许通过增加类名如class="header"、class="footer"，使HTML页面具有语义性，但是不具有通用性。

HTML5则是通过新增语义标签的形式来解决这个问题，例如<header></header>、<footer></footer>等，这样就可以使其具有通用性。

在此基础上HTML5增加了大量更有意义的语义标签，更有利于搜索引擎或辅助设备理解HTML页面内容。

通过查看全球各大网站的标签使用情况 从而进一步理解新增标签的由来。

此章节学习目的为了解增加语义标签的目的，以及各语义标签所表达的意义，在网页布局中能够合理使用标签。

传统网页布局：



Html5 语义化标签布局



### 常见语义化标签

<nav> 表示导航 <div class=”nav”></div>

<header> 表示页眉 <div class=”header”></header>

<footer> 表示页脚

<section> 表示区块

<article> 表示 文章 如文章、评论、帖子、博客

<aside> 表示侧边栏 如文章的侧栏

<figure> 表示媒介内容分组 与 ul > li 做个比较

figcaption

img

<progress> 表示进度 (带用“UI”，不怎么用) max min value

<time> 表示日期

其他：

details open属性 summary汇总 p

hgroup h1 h2 address mark

### 新语义标签的的区别

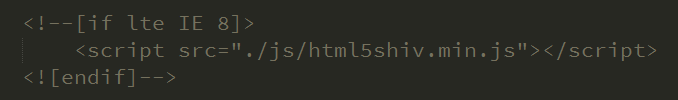
本质上新语义标签与<div>、<span>没有区别，只是其具有表意性，使用时除了在HTML结构上需要注意外，其它和普通标签的使用无任何差别，可以理解成<div class="nav"> 相当于 <nav>。不要好奇，它只是一个标签！

尽量避免全局使用header、footer、aside等语义标签。

### 兼容处理

我们在测试ie 的兼容的时候，有一个叫做ietest 的软件，这个软件可以模拟ie6-ie11

在不支持HTML5新标签的浏览器里，会将这些新的标签解析成行内元素(inline)对待，所以我们只需要将其转换成块元素(block)即可使用，但是在IE9版本以下，并不能正常解析这些新标签，但是却可以识别通过document.createElement('tagName')创建的自定义标签，于是我们的解决方案就是将HTML5的新标签全部通过document.createElement('tagName')来创建一遍，这样IE低版本也能正常解析HTML5新标签了，在实际开发中我们更多采用的是通过检测IE浏览器的版本来加载三方的一个JS库来解决兼容问题



## HTML5 表单新增的类型

伴随着互联网富应用以及移动开发的兴起，传统的Web表单已经越来越不能满足开发的需求，所以HTML5在Web表单方向也做了很大的改进，如拾色器、日期/时间组件等，使表单处理更加高效。

此章节学习目的，了解HTML5表单的新增的特性，以及PC和移动设备间的差异，其兼容性较差。

### 新增的类型（在移动端支持会更好）

email 输入email格式

tel 手机号码(在移动端，定位焦点，会弹出数字键盘)

url 只能输入url格式

number 只能输入数字 (在移动端，定位焦点，会弹出数字键盘)

search 搜索框 (在移动端 enter 键里面的文字 会变成搜索)

range 范围

color 拾色器

time 时间

date 日期 不是绝对的 （会出现日期选择器）

datetime-local 时间日期

month 月份

week 星期

部分类型是针对移动设备生效的，且具有一定的兼容性，在实际应用当中可选择性的使用。

## HTML5 新增的属性介绍

placeholder 占位符

autofocus 获取焦点

required 验证条件，必填项

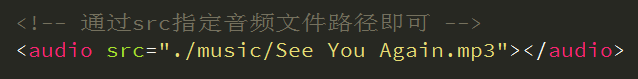
Pattern : 正则验证 pattern="\d{1,5}“

## 多媒体标记

### 音频

HTML5通过<audio>标签来解决音频播放的问题。

使用相当简单，如下图所示

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

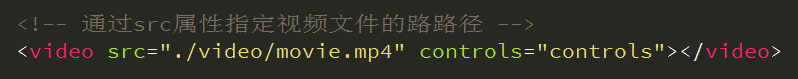
loop 循环播放

source 音视频源

### 视频

HTML5通过<video>标签来解决音频播放的问题。

同音频播放一样，<video>使用也相当简单，如下图



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

## HTML5 自定义属性

在HTML5中我们可以自定义属性，其格式如下data-\*=""，例如

data-info="我是自定义属性"，通过Node.dataset['info'] 我们便可以获取到自定义的属性值。

Node.dataset是以类对象形式存在的

当我们如下格式设置时，则需要以驼峰格式才能正确获取

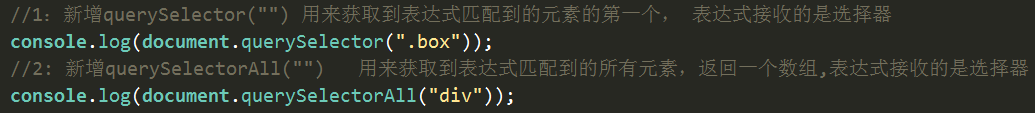


## HTML5 新增dom 操作API

### 获取元素

1、document.querySelector('selector') 通过CSS选择器获取元素，符合匹配条件的第1个元素。

2、document.querySelectorAll('selector') 通过CSS选择器获取元素，以类数组形式存在。



### 类名操作

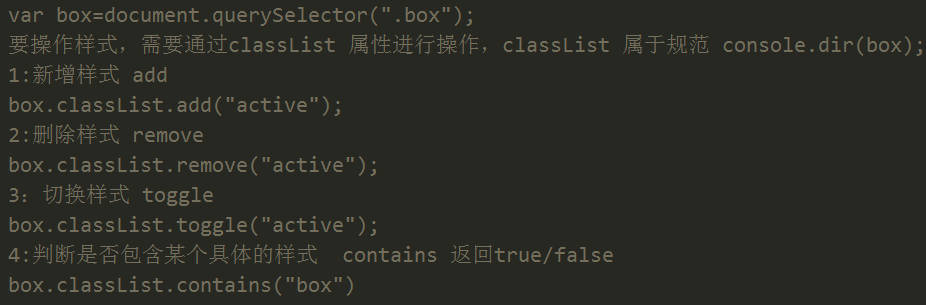
1、Node.classList.add('class') 添加class

2、Node.classList.remove('class') 移除class

3、Node.classList.toggle('class') 切换class，有则移除，无则添加

4、Node.classList.contains('class') 检测是否存在class

Node指一个有效的DOM节点，是一个通称。



### Tab 选项卡案例

(function(key){  
 var navs=document.querySelectorAll("nav>a");  
 //我要遍历 navs 集合  
 for(var i=0;i<navs.length;i++){  
 if(i==key){  
 //默认选中那个导航..  
 navs[i].classList.add("active");  
 var cont= navs[i].dataset['cont'];  
 //根据导航找到它对应的tabs 选项. 样式选择器去找  
 document.getElementById(cont).style.display="block";  
 }  
 //我需要给每个导航添加点击事件..  
 navs[i].onclick=function(){  
 /\* 第一步，我要过滤掉无效的点击this 指向当前点击的a 标签.\*/  
 // 判断当前元素是否具备某个样式. this.classList.contains("")  
 if(this.classList.contains("active")){  
 return;  
 }  
 /\*我点击的超链接. //获取到当前选中的a 标签 \*/  
 var active=document.querySelector(".active");  
 active.classList.remove("active")  
 //找到对应的tabs 项..  
 var ccont=active.dataset['cont'];  
 document.querySelector("#"+ccont).style.display="none";  
 //给当前点击的添加一个背景..  
 this.classList.add("active")  
 var currenta=this.dataset['cont'];  
 document.getElementById(currenta).style.display="block";  
 }  
 }  
})(2);