

# 移动端\_Day01---HTML5

作者：李振超

## 一、了解HTML5

是对HTML标准的第五次修订。其主要的目标是将互联网语义化，以便更好地被人类和机器阅读，并同时提供更好地支持各种媒体的嵌入。HTML5的语法是向后兼容的。

HTML5是从Web Applications 1.0项目发展而来的，由网页超文本技术工作小组（WHATWG）发起，后被W3C采纳。HTML5规范的大部分内容都偏重于处理Web应用程序。

目前最新的HTML5.1标准在2016发布，并且成为w3c的推荐标准。HTML5.2标准亦在制定中。

W3C » Participate » Mail, News, Blogs, Podcasts, and... » W3C News

### HTML 5.1 IS A W3C RECOMMENDATION

1 November 2016



The [Web Platform Working Group](#) has published a W3C Recommendation of [HTML 5.1](#). This specification defines the 5th major version, first minor revision of the core language of the World Wide Web: the Hypertext Markup Language (HTML). In this version, new features continue to be introduced to help Web application authors, new elements continue to be introduced based on research into prevailing authoring practices, and special attention continues to be given to defining clear conformance criteria for user agents in an effort to improve interoperability.

HTML5不仅仅是对HTML的升级改进，更加体现了**Web应用程序**这个概念。HTML5目前是包括CSS3和JavaScript技术在内的技术和标准的统称。

HTML5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。

**广义概念：**HTML5代表浏览器端前端技术的一个新发展阶段。在这个阶段，浏览器呈现技术得到了一个飞跃发展和广泛支持，HTML5是包括：HTML5，CSS3，Javascript API在内的一套技术组合

## 二、HTML5语法规范

**特点：**HTML5的语法更简洁和更宽松

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。

1. 简洁的文档声明。

```
1 //HTML5的文档声明
2 <!DOCTYPE html>
3
4 //下面是以前的4.01过渡版本的文档声明
5 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
6     "http://www.w3.org/TR/html4/loose.dtd">
```

2. 标签支持大写和小写。

**建议：**开发过程中所有标签名小写！

3. 空标签可以关闭也可以不关闭。

**建议：**所有的标签都有关闭，空标签也建议关闭。 对自己来说至少要做到统一。比如：空标签要么都关闭要么都不关闭。

```
1 <!--空标签可以不关闭-->
2 <img src="" >
3 <!--空标签也可以关闭-->
4 <img src="" />
```

4. 双标签可以省略结束标签。但是建议永远不要省略，因为浏览器在自动给你补齐的时候，可能并不是你想要的。

```
1 <!--在html5中，这是完全合法的。 只有开始标签没有结束标签-->
2 <div id=box>
```

5. 属性值双引号(")或单引号(')均可以省略。

*建议：不要省略双引号和单引号*

- 1 <!--属性值的引号可以省略。 但是有一点例外：属性值如果有空格的时候，则引号不能省略，比如多个类名-->
- 2 <div id=box></div>

6. script、style、link标签的type属性可以省略。

- 1 <!--属性 type="text/css" 可以省略-->
- 2 <style></style>
- 3 <!-- type="text/javascript" 可以省略-->
- 4 <script></script>
- 5 <!-- type="text/css" 可以省略-->
- 6 <link rel="stylesheet" href="" />

7. 更简单方法指定字符编码

- 1 <!--HTML5指定页面字符编码的方式-->
- 2 <meta charset="UTF-8">
- 3
- 4 <!--以前的版本指定页面字符编码的方式-->
- 5 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

7. html、head、body可以完全省略。

*建议永远不要省略！*

*而且省略后有些浏览器中可能会崩溃*

## 三、HTML5新增语义标签

### 3.1 常用新增结构化语义元素

在以前的html中，我们通常会用很多的div来给网页布局，然后给div定义有一个意义的类。这种情况一般情况下程序员是能读的懂的，但是搜索引擎很难理解每个div到底代表什么意思。

看下面的代码：

```
1 <body>
2 <div id="wrapper">
3
4     <!--the header -->
5     <div id="header">
6
7         <div id="logo"></div>
8         <div id="navigation">
9             <ul>
10                 <li><a href="#">Why?</a></li>
11             </ul>
12         </div>
13     </div>
14
15     <!-- the content -->
16     <div id="content">
17     </div>
18
19     <!-- the sidebar -->
20     <div id="sidebar">
21     </div>
22
23     <!-- the footer -->
24     <div id="footer">
25     </div>
26 </div>
27 </body>
```

HTML5 旨在使用新增的全新语义化元素来解决问题。

### 3.1.1 标签:<section>

<section>元素用来定义文档或应用程序中的区域（或节）。例如，可以用它组织你的个人信息，一个<section>用于联系信息，另一个用于新闻动态。需要重点理解的是用它的目的不是为了美化样式。如果你只想将某个元素包裹起来以便设置样式，那应该和以前一样继续使用<div>。

section 元素代表文档或应用的一个一般的区块。section 是具有相似主题的一组内容，通常包含一个标题。

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="utf-8">
5     <title>section</title>
6 </head>
7 <body>
8
9 <section>
10     <h1>苹果</h1>
11     <p>
12         一种树上结的果子，很甜，有的也酸
13     </p>
14 </section>
15
16 <section>
17     <h1>香蕉</h1>
18     <p>
19         一种书上结果的水平，很长，很黄，女生比较喜欢吃
20     </p>
21 </section>
22
23 </body>
24 </html>
```

### 3.1.2 标签:<nav>

<nav>元素用来定义文档的主导航区域，其中的链接指向其他页面或当前页面的某些区域。因为<nav>用于主导航区域，所以严格来讲它不是为页脚或其他经常会包含一组链接的区块而设计的（**虽然将它用在这些区块里包含链接也没问题**）。

**将一组链接指定为重要导航!**

比如一个网站的最上方的主导航。

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <nav>
6   <a href="/html/">HTML</a> |
7   <a href="/css/">CSS</a> |
8   <a href="/js/">JavaScript</a> |
9   <a href="/jquery/">jQuery</a>
10 </nav>
11
12 </body>
13 </html>
```

### 3.1.3 标签:

元素表示文档、页面、应用或网站中一个独立的容器，原则上是可独立分配或可再用的，就像聚合内容中的各部分。它可以是一篇论坛帖子、一篇杂志或报纸文章、一篇博客条目、一则用户提交的评论、一个交互式的小部件或小工具，或者任何其他独立的内容项。

元素用来包裹独立的内容片段。当搭建一个页面时，想想你准备放入标签的内容能否作为一个整块而被复制粘贴到另外一个完全不同的网站且能保持完整的意义？另一种办法是，想想包裹在其中的内容能否在 RSS 订阅源中独立成为一篇文章？**应该使用标签包裹的内容，最明显的例子就是博客正文。**

注意：

元素与元素很容易混淆。

在考虑何时使用 section 的时候，记住定义中“具有相似主题的一组内容”这一条是很有帮助的。这也是 section 区别于 div 的另一个原因。

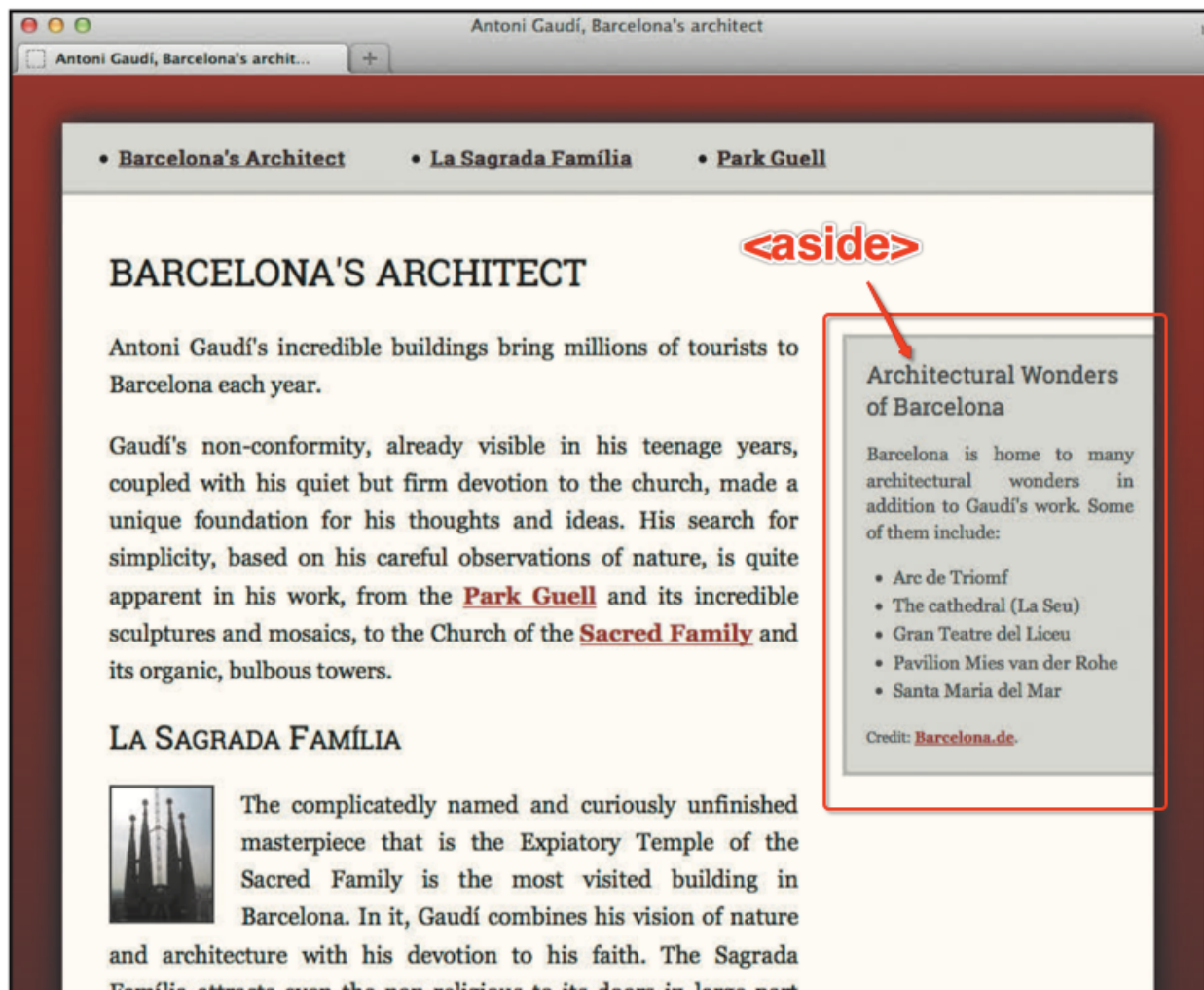
section 和 article 的区别在于，section 在本质上组织性和结构性更强，而 article 代表的是自包含的容器。

对内容进行标记时，并非总能分出对和错，只是大多数时候有正确的选择。而其他时候，就只能依靠个人对最适合描述内容的 HTML 元素的感觉了。

在考虑是否使用 section 的时候，一定要仔细思考，不过也不必每次都对是否用对感到担心。有时，些许主观并不会影响页面正常工作。

### 3.1.4 标签:

元素用来表示与页面主内容松散相关的内容。在实践中，我经常将其用作侧边栏（当它包含合适的内容时）。另外，引文、广告以及导航元素（如友情链接等）也可以使用它。



### 3.1.5 标签:

如果页面中有一块包含一组介绍性或导航性内容的区域，应该用 `header` 元素对其进行标记。

一个页面可以有任意数量的 `header` 元素，它们的含义可以根据其上下文而有所不同。例如，处于页面顶端或接近这个位置的 `header` 可能代表整个页面的页眉（有时称为页头）

通常，页眉包括网站标志、主导航和其他全站链接，甚至搜索框。这无疑是 `header` 元素最常见的使用形式。



### 3.1.6 标签:

当你想到页脚的时候，你大概想的是页面底部的页脚（通常包括版权声明，可能还包括指向隐私政策页面的链接以及其他类似的内容）。HTML5 的 footer 元素可以用在这样的地方，但它同 header 一样，还可以用在其他地方。

footer 元素代表嵌套它的最近的 article、aside、blockquote、body、details、fieldset、figure、nav、section 或 td 元素的页脚。只有当它最近的祖先是 body 时，它才是整个页面的页脚。



## 3.2 常用非结构化语义标签

### ► Details

#### 3.2.1 标签:

和

标签规定独立的流内容（图像、图表、照片、代码等等）。

元素的内容应该与主内容相关，同时元素的位置相对于主内容是独立的。如果被删除，则不应影响文档流。

主要用来标记图片的。

以前添加图片列表经常这样写：



```
1 <li>
2   <p>这个是个美女图片</P><img src="" />
3 </li>
```

上面这种写法明显没有语义。使用figure可以增加语义。使用

替换掉

```
●1 <figure>
2   <p>这个是美女图片</p>
3   
4 </figure>
```

是这个图片的标题，则可以使用  
来替换

```
1 <figure>
2   <figcaption>这个是美女图片</figcaption>
3   
4 </figure>
```

### 3.2.2 标签:

标签定义带有记号的文本。

出于引用的目的，对与另一个上下文相关的文本进行突出显示

```
1 <h1>美女</h1>
2 <p>四大<mark>美女</mark></p>
3 <p>杨玉环是<mark>美女</mark></p>
4 <p>凤姐也是<mark>美女</mark></p>
```

效果:

# 美女

四大美女

杨玉环是美女

凤姐也是美女

### 3.2.3 标签:

它指示某项任务的完成进度。可以用它表示一个进度条

```
1 <progress></progress>
```

`progress`主要支持三个可选的属性：`max`，`value`和`form`

- `max`属性指定任务的总工作量。其值必须大于0.其默认值是1
- `value`是任务已经完成量。
- `form`的含义，下章讲表单的时候再讲。

可以通过js来动态的更改`value`的值，从而达到显示任务动态的完成情况。

```
1 <progress max="100"></progress>
2 <script type="text/javascript">
3     var pro =
document.getElementsByTagName("progress")[0];
4     var i = 0;
5     setTimeout(function step() {
6         //更改progress的value的值
7         pro.value = i++;
8         if(i == 100) return;
9         setTimeout(step, 100);
10    }, 100)
11 </script>
```

### 3.2.4 标签:

在以前的网页中没有标准的语义标签来表示时间，`time`标签填补了这个空白。其目的是让时间有语义，让搜索引擎等其它程序可以更容易的提取这些信息。

**`time`标签在任何浏览器中都不会呈现出任何特殊的显示效果。**

## 四、增强的表单元素

## 五、文件API