

第 1 篇

HTML 5 网页设计

- 第 1 章 初识 HTML 5
- 第 2 章 HTML 5 网页的文档结构
- 第 3 章 HTML 5 网页中的文本和图像
- 第 4 章 用 HTML 5 建立超链接
- 第 5 章 用 HTML 5 创建表格
- 第 6 章 用 HTML 5 创建表单
- 第 7 章 用 HTML 5 绘制图形
- 第 8 章 HTML 5 中的音频和视频

第 1 章

初识 HTML 5

目前，网络已经成为人们娱乐、工作中不可或缺的一部分，网页设计也成为学习计算机知识的重要内容之一。制作网页可采用可视化编辑软件，但是无论采用哪一种网页编辑软件，最后都是将所设计的网页转化为 HTML。

HTML 是网页设计的基础语言，本章就来介绍 HTML 的基本概念和编写方法，以及浏览 HTML 文件的方法，使读者初步了解 HTML，从而为后面的学习打下基础。

1.1 HTML 5 的基本概念

因特网上的信息是以网页形式展示给用户的,网页是网络信息传递的载体。网页文件是用标记语言书写的,这种语言称为超文本标记语言(Hyper Text Markup Language, HTML)。

1.1.1 HTML 的发展历程

HTML 是一种描述语言,而不是一种编程语言,主要用于描述超文本中内容的显示方式。标记语言从诞生至今,经历了 20 多年,发展过程中也有很多曲折,经历的版本及发布日期如表 1-1 所示。

表 1-1 超文本标记语言的发展过程

版本	发布日期	说明
超文本标记语言(第一版)	1993 年 6 月	作为互联网工程工作小组(IETF)工作草案发布(并非标准)
HTML 2.0	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过时
HTML 3.2	1996 年 1 月 14 日	W3C 推荐标准
HTML 4.0	1997 年 12 月 18 日	W3C 推荐标准
HTML 4.01	1999 年 12 月 24 日	微小改进, W3C 推荐标准
ISO HTML	2000 年 5 月 15 日	基于严格的 HTML 4.01 语法,是国际标准化组织和国际电工委员会的标准
XHTML 1.0	2000 年 1 月 26 日	W3C 推荐标准(修订后于 2002 年 8 月 1 日重新发布)
XHTML 1.1	2001 年 5 月 31 日	较 1.0 有微小改进
XHTML 2.0 草案	没有发布	2009 年, W3C 停止了 XHTML 2.0 工作组的工作
HTML 5	2014 年 10 月	W3C 推荐标准

1.1.2 什么是 HTML 5

HTML 5 是一种描述性的标记语言,用于描述超文本中的内容和结构。HTML 最基本的语法是<标记符></标记符>。标记符通常都是成对使用,有一个开头标记和一个结束标记。结束标记只是在开头标记的前面加一个斜杠“/”。当浏览器收到 HTML 文件后,就会解释里面的标记符,然后把标记符相对应的功能表达出来。

例如,在 HTML 5 中用<p></p>标记符来定义一个换行符。当浏览器遇到<p></p>标记符时,会把该标记中的内容自动形成一个段落。当遇到
标记符时,会自动换行,并且该标记符后的内容会从一个新行开始。这里的
标记符是单标记,没有结束标记,标记后的“/”符号可以省略;但为了使代码规范,一般建议加上。

1.1.3 HTML 5 文件的基本结构

完整的 HTML 文件包括标题、段落、列表、表格、绘制的图形以及各种嵌入对象，这些对象统称为 HTML 元素。一个 HTML 5 文件的基本结构如下：

<code><!DOCTYPE html></code>	
<code><html></code>	文件开始的标记
<code><head></code>	文件头部开始的标记
<code>...</code>	文件头部的内容
<code></head></code>	文件头部结束的标记
<code><body></code>	文件主体开始的标记
<code>...</code>	文件主体的内容
<code></body></code>	文件主体结束的标记
<code></html></code>	文件结束的标记

从上面的代码可以看出，在 HTML 文件中，所有的标记都是相对应的，开头标记为`< >`，结束标记为`</>`，在这两个标记中间添加内容。这些基本标记的使用方法及详细解释见第 2 章的内容。

1.2 HTML 5 的优势

从 HTML 4.0、XHTML 到 HTML 5，从某种意义上讲，这是 HTML 描述性标记语言的一种更加规范的过程，因此，HTML 5 并没有给开发者带来多大的冲击。但 HTML 5 也增加了很多非常实用的新功能。下面介绍 HTML 5 的一些优势。

1.2.1 解决了跨浏览器问题

浏览器是网页的运行环境，因此浏览器的类型也是在网页设计时会遇到的一个问题。由于各个软件厂商对 HTML 标准的支持有所不同，导致同样的网页在不同的浏览器下会有不同的表现。并且 HTML 5 新增的功能在各个浏览器中的支持程度也不一致，浏览器的因素变得比以往传统的网页设计更加重要。

为了保证设计出来的网页在不同浏览器上效果一致，HTML 5 会让问题简单化，具备良好的跨浏览器性能。针对不支持新标签的老式 IE 浏览器，用户只要简单地添加 JavaScript 代码，就可以让它们使用新的 HTML 5 元素。

1.2.2 新增了多个新特性

HTML 语言从 1.0 至 5.0 经历了巨大的变化，从单一的文本显示功能到图文并茂的多媒体显示功能，许多特性经过多年的完善，已经成为一种非常重要的标记语言。尤其是 HTML 5，对多媒体的支持功能更强。具体而言，它具备如下功能。

- (1) 新增了语义化标签，使文档结构明确。
- (2) 新的文档对象模型(DOM)。

- (3) 实现了 2D 绘图的 Canvas 对象。
- (4) 可控媒体播放。
- (5) 离线存储。
- (6) 文档编辑。
- (7) 拖放。
- (8) 跨文档消息。
- (9) 浏览器历史管理。
- (10) MIME 类型和协议注册。

对于这些新功能，支持 HTML 5 的浏览器在处理 HTML 5 代码错误的时候必须更灵活，而那些不支持 HTML 5 的浏览器将忽略 HTML 5 代码。

1.2.3 用户优先的原则

HTML 5 标准的制定是以用户优先为原则的，一旦遇到无法解决的冲突时，规范会把用户放到第一位，其次是网页的作者，再次是浏览器，接着是规范制定者(W3C/WHATWG)，最后才考虑理论的纯粹性。所以，总体来看，HTML 5 的绝大部分特性还是实用的，只是有些情况下还不够完美。

举例说明一下，下述 3 行代码虽然有所不同，但在 HTML 5 中都能被正确识别：

```
id="HTML 5"
id=HTML 5
iD="HTML 5"
```

在以上示例中，除了第一个外，另外两个语法都不是很严谨，这种不严谨的语法被广泛使用，遭到一些技术人员的反对。但是无论语法严格与否，对网页查看者来说没有任何影响，他们只需要看到想要的网页效果就可以了。

为了增强 HTML 5 的使用体验，还加强了以下两个方面的设计。

1. 安全机制的设计

为确保 HTML 5 的安全，在设计 HTML 5 时做了很多针对安全的设计。HTML 5 引入了一种新的基于来源的安全模型，该模型不仅易用，而且对各种不同的 API 都通用。使用这个安全模型，可以做一些以前做不到的事情，不需要借助于任何所谓聪明、有创意却不安全的 hack 就能跨域进行安全对话。

2. 表现和内容分离

表现和内容分离是 HTML 5 设计中的另一个重要内容，HTML 5 在所有可能的地方都努力进行了分离，也包括 CSS。实际上，表现和内容的分离早在 HTML 4 中就有设计，但是分离得并不彻底。为了避免可访问性差、代码高复杂度、文件过大等问题，HTML 5 规范中更细致、清晰地分离了表现和内容。但是考虑 HTML 5 的兼容性问题，一些老的表现和内容的代码还是可以兼容使用的。

1.2.4 化繁为简的优势

作为当下流行的通用标记语言，HTML 5 越简单越好。所以在设计 HTML 5 时，严格遵循了“简单至上”的原则，主要体现在以下几个方面。

- (1) 新的简化的字符集声明。
- (2) 新的简化的 DOCTYPE。
- (3) 简单而强大的 HTML 5 API。
- (4) 以浏览器原生能力替代复杂的 JavaScript 代码。

为了实现以上这些简化操作，HTML 5 规范需要比以前更加细致、精确，比以往任何版本的 HTML 规范都要精确。

在 HTML 5 规范细化的过程中，为了避免造成误解，几乎对所有内容都给出了彻底、完全的定义，特别是对 Web 应用。

基于多种改进过的、强大的错误处理方案，HTML 5 具备了良好的错误处理机制。具体来讲，HTML 5 提倡重大错误的平缓恢复，再次把最终用户的利益放在了第一位。举例来说，如果页面中有错误的话，在以前，可能会影响整个页面的显示，而在 HTML 5 中，不会出现这种情况，取而代之的是以标准方式显示 broken 标记，这要归功于 HTML 5 中精确定义的错误恢复机制。

1.3 HTML 5 文件的编写方法

有两种方式可以产生 HTML 文件：一种是自己写 HTML 文件，事实上这并不是很困难，也不需要特别的技巧；另一种是使用 HTML 编辑器，它可以辅助使用者来做编写工作。

1.3.1 使用记事本手工编写 HTML 5

前面介绍过，HTML 5 是一种标记语言，标记语言代码是以文本形式存在的。因此，所有的记事本工具都可以作为它的开发环境。

HTML 文件的扩展名为.html 或.htm，将 HTML 源代码输入到记事本并保存之后，可以在浏览器中打开文档以查看其效果。

使用记事本编写 HTML 文件的具体操作步骤如下。

step 01 单击 Windows 桌面上的“开始”按钮，选择“所有程序”→“附件”→“记事本”命令，打开一个记事本，在记事本中输入 HTML 代码，如图 1-1 所示。

step 02 编辑完 HTML 文件后，选择“文件”→“保存”命令或按 Ctrl+S 组合键，在弹出的“另存为”对话框中，选择“保存类型”为“所有文件”，然后将文件扩展名设为.html 或.htm，如图 1-2 所示。

step 03 单击“保存”按钮，即可保存文件。打开网页文档，在浏览器中预览，如图 1-3 所示。

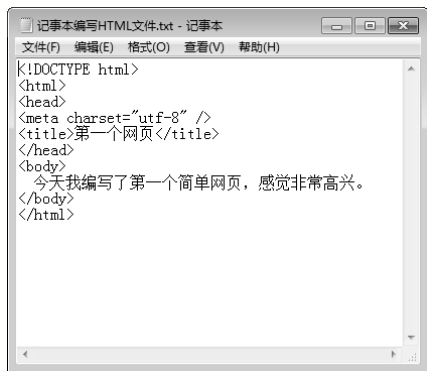


图 1-1 编辑 HTML 代码

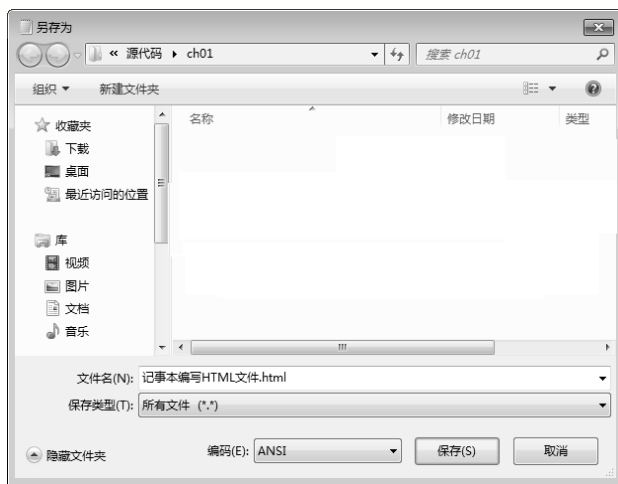


图 1-2 “另存为”对话框

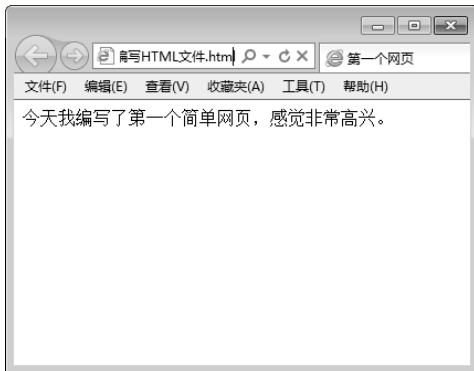


图 1-3 网页的浏览效果

1.3.2 使用 Dreamweaver CC 编写 HTML 文件

“工欲善其事，必先利其器。”虽然使用记事本可以编写 HTML 文件，但是编写效率太低，对于语法错误及格式都没有提示。因此，很多专门编写 HTML 网页的工具弥补了这种缺陷。Adobe 公司的 Dreamweaver CC 用户界面非常友好，是一款非常优秀的网页开发工具，深受广大用户的喜爱。Dreamweaver CC 的主界面如图 1-4 所示。

1. 文档窗口

文档窗口位于界面的中部，它是用来编排网页的区域，与在浏览器中的结果相似。在文档窗口中，可以将文档分为 3 种视图显示模式。

- (1) 代码视图。使用代码视图，可以在文档窗口中显示当前文档的源代码，也可以在该窗口中直接键入 HTML 代码。
- (2) 设计视图。在设计视图下无须编辑任何代码，可以直接使用可视化的操作编辑网页。

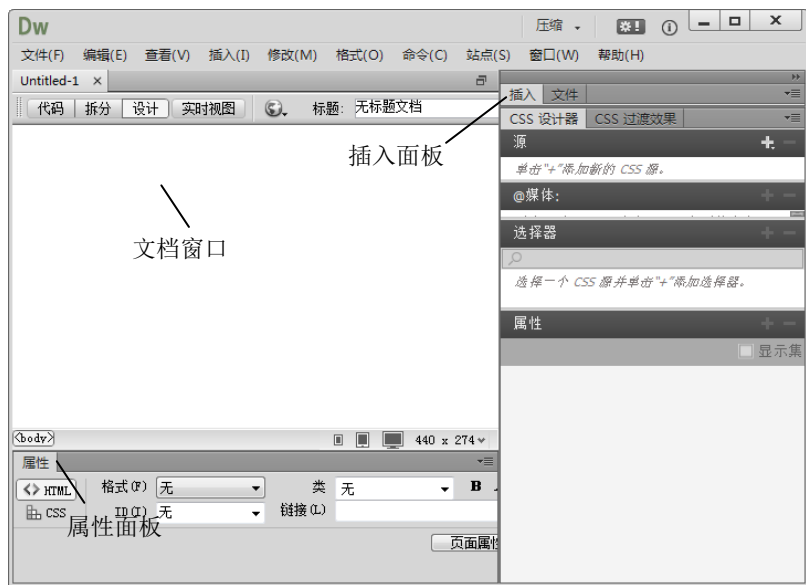


图 1-4 Dreamweaver CC 的主界面

(3) 拆分视图。在拆分视图下，左半部分显示代码视图，右半部分显示设计视图。在这种视图模式下，可以通过键入 HTML 代码直接观看效果，还可以通过设计视图插入对象，直接查看源文件。

在各种视图间进行切换时，只需要在文档工具栏中单击相应的视图按钮即可。文档工具栏如图 1-5 所示。



图 1-5 文档工具栏

2. “插入”面板

“插入”面板是在设计视图下使用频度很高的面板之一。“插入”面板默认打开的是“常用”页，它包括了最常用的一些对象。例如，在文档中的光标位置插入一段文本、图像或表格等。用户可以根据需要切换到其他页，如图 1-6 所示。

3. “属性”面板

“属性”面板中主要包含当前选择的对象相关属性的设置。可以通过菜单栏中的“窗口”→“属性”命令或 Ctrl+F3 组合键打开或关闭“属性”面板。

“属性”面板是常用的一个面板，因为无论要编辑哪个对象的属性，都要用到它。其内容也会随着选择对象的不同而改变。例如，当光标定位在文档主体文字内容部分时，“属性”面板显示文字的相关属性，如图 1-7 所示。



图 1-6 “插入”面板



图 1-7 “属性”面板

Dreamweaver CC 中还有很多面板,在以后使用时,再做详细讲解。打开的面板越多,编辑文档的区域就会越小,为了便于编辑文档,可以通过 F4 功能键快速隐藏和显示所有面板。使用 Dreamweaver CC 编写 HTML 文件的具体操作步骤如下。

step 01 启动 Dreamweaver CC,如图 1-8 所示,在欢迎屏幕的“新建”栏中选择 HTML 选项。或者选择菜单栏中的“文件”→“新建”命令(快捷键为 Ctrl+N)。



图 1-8 包含欢迎屏幕的主界面

step 02 弹出“新建文档”对话框,如图 1-9 所示,在“页面类型”列表中选择 HTML 选项。



图 1-9 “新建文档”对话框

step 03 单击“创建”按钮，创建 HTML 文件，如图 1-10 所示。



图 1-10 在设计视图下显示创建的文档

step 04 在文档工具栏中，单击“代码”按钮，切换到代码视图，如图 1-11 所示。

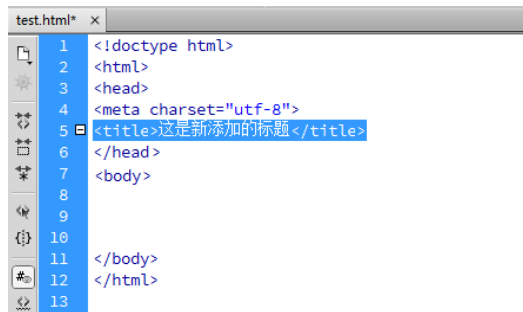


图 1-11 在代码视图下显示创建的文档


step 05 修改 HTML 文档标题，将代码中<title>标记中的网页标题修改成“第一个网页”。然后在<body>标记中键入“今天我使用 Dreamweaver CC 编写了第一个简单网页，感觉非常高兴。”。

完整的 HTML 代码如下：

```

<!doctype html >
<html>
<head>
<meta charset=utf-8" />
<title>第一个网页</title>
</head>
<body>
今天我使用 Dreamweaver CC 编写了第一个简单网页，感觉非常高兴。
</body>
</html>
  
```

step 06 保存文件。从菜单栏中选择“文件”→“保存”命令或按 Ctrl+S 组合键，弹出“另存为”对话框。在该对话框中选择保存位置，并输入文件名，单击“保存”按钮，如图 1-12 所示。

step 07 单击文档工具栏中的  图标，选择查看网页的浏览器，或按 F12 键，使用默认浏览器查看网页，预览效果如图 1-13 所示。

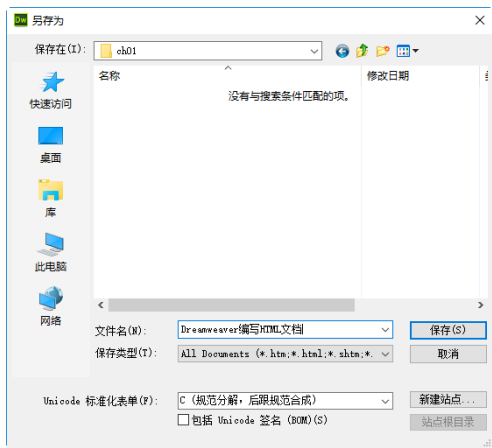


图 1-12 保存文件

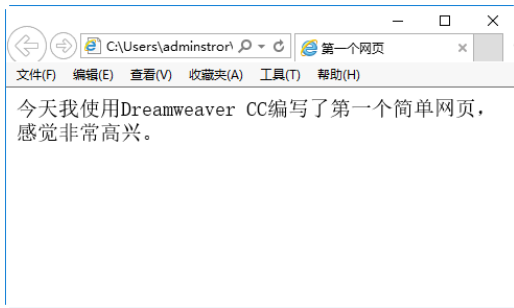


图 1-13 浏览器预览效果

1.4 使用浏览器查看 HTML 5 文件

开发者经常需要查看 HTML 源代码及其效果。使用浏览器可以查看网页的显示效果，也可以在浏览器中直接查看 HTML 源代码。

1.4.1 查看页面效果

前面已经介绍过，为了测试网页的兼容性，可以在不同的浏览器中打开网页。在非默认浏览器中打开网页的方法有很多种，在此介绍两种常用的方法。

方法一：选择浏览器中的“文件”→“打开”命令(有些浏览器的菜单命令为“打开文件”)，选择要打开的网页即可，如图 1-14 所示。

方法二：在 HTML 文件上右击，从弹出的快捷菜单中选择“打开方式”，然后选择需要的浏览器，如图 1-15 所示。如果浏览器没有出现在菜单中，可以选择“选择其他应用(C)”命令，在计算机中查找浏览器程序。

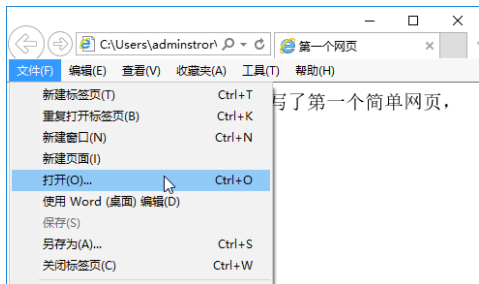


图 1-14 选择“打开”命令

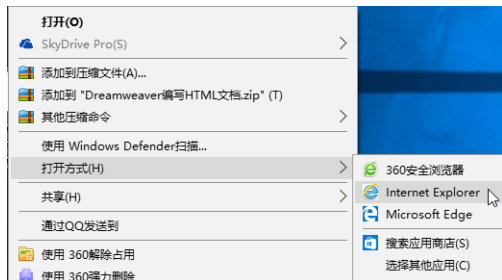


图 1-15 选择不同的浏览器来打开网页

1.4.2 查看源文件

查看网页源代码的常见方法有以下两种。

(1) 在页面空白处右击，从弹出的快捷菜单中选择“查看源”命令，如图 1-16 所示。



图 1-16 选择“查看源”命令

(2) 在浏览器的菜单栏中选择“查看”→“源”命令，如图 1-17 所示。

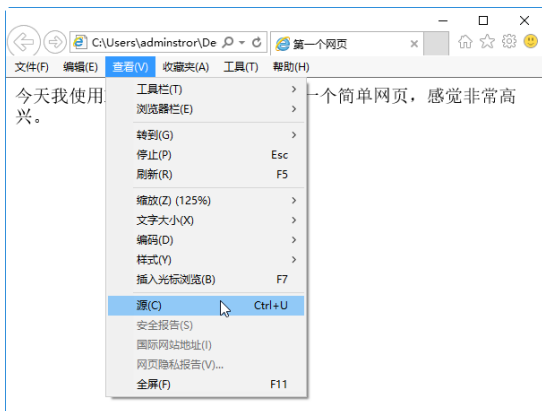


图 1-17 选择“源”命令



由于浏览器的规定各不相同，有些浏览器将“查看源”命名为“查看源代码”，但是，操作方法完全相同。

1.5 疑难解惑

疑问 1: 为何使用记事本编辑的 HTML 文件无法在浏览器中预览，而是直接在记事本中打开？

很多初学者，在保存文件时，没有将 HTML 文件的扩展名.html 或.htm 作为文件的后缀，

导致文件还是以.txt 为扩展名, 因此无法在浏览器中查看。如果读者是通过鼠标右击创建记事本文件的, 在为文件重命名时, 一定要以.html 或.htm 作为文件的后缀。特别要注意的是, 当 Windows 系统的扩展名隐藏时, 更容易出现这样的错误。读者可以在“文件夹选项”对话框中查看是否显示扩展名。

疑问 2: 如何显示或隐藏 Dreamweaver CC 的欢迎屏幕?

Dreamweaver CC 的欢迎屏幕可以帮助使用者快速进行打开文件、新建文件和获取相关帮助的操作。如果读者不希望显示该窗口, 可以按 **Ctrl+U** 组合键, 在弹出的窗口中, 选择左侧的“常规”页, 将右侧“文档选项”部分的“显示欢迎屏幕”取消勾选。

第 2 章

HTML 5 网页的 文档结构

一个完整的 HTML 5 网页文档包括标题、段落、列表、表格、绘制的图形以及各种嵌入对象，这些对象统称为 HTML 5 元素。本章详细介绍 HTML 5 网页文档的基本结构。

2.1 HTML 5 文件的基本结构

在一个 HTML 5 文档中, 必须包含<HTML></HTML>标记, 并且放在一个 HTML 5 文档的开始和结束位置。即每个文档以<HTML>开始, 以</HTML>结束。<HTML></HTML>之间通常包含两个部分, 分别为<HEAD></HEAD>和<BODY></BODY>。HEAD 标记包含 HTML 头部信息, 如文档标题、样式定义等。BODY 包含文档主体部分, 即网页内容。需要注意的是, HTML 标记不区分大小写。

2.1.1 HTML 5 页面的整体结构

为了便于读者从整体上把握 HTML 5 的文档结构, 我们通过一个 HTML 5 页面来介绍 HTML 5 页面的整体结构, 示例代码如下:

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
  <TITLE>网页标题</TITLE>
</HEAD>
<BODY>
  网页内容
</BODY>
</HTML>
```

从上面的代码可以看出, 一个基本的 HTML 5 页面由以下几个部分构成。

(1) <!DOCTYPE HTML>声明。该声明必须位于 HTML 5 文档中的第一行, 也就是位于<html>标记之前。该标记告知浏览器文档所使用的 HTML 规范。<!DOCTYPE HTML>声明不属于 HTML 标记; 它是一条指令, 告诉浏览器编写页面所用的标记的版本。由于 HTML 5 版本还没有得到浏览器的完全认可, 后面介绍时还采用以前的通用标准。

(2) <HTML></HTML>标记。说明本页面是用 HTML 语言编写的, 使浏览器软件能够准确无误地解释和显示。

(3) <HEAD></HEAD>标记。是 HTML 的头部标记, 头部信息不显示在网页中, 此标记内可以包含一些其他标记, 用于说明文件标题和整个文件的一些公用属性。可以通过<style>标记定义 CSS 样式表, 通过<script>标记定义 JavaScript 脚本文件。

(4) <TITLE></TITLE>标记。TITLE 是 HEAD 中的重要组成部分, 它包含的内容显示在浏览器的窗口标题栏中。如果没有 TITLE, 浏览器标题栏将显示本页的文件名。

(5) <BODY></BODY>标记。BODY 包含 HTML 页面的实际内容, 显示在浏览器窗口的客户区中。例如, 在页面中, 文字、图像、动画、超链接以及其他 HTML 相关的内容都是定义在 BODY 标记里面的。

2.1.2 HTML 5 新增的结构标记

HTML 5 新增的结构标记有<footer></footer>和<header></header>标记。但是, 这两个标

记还没有获得大多数浏览器的支持，这里只做简单介绍。

<header>标记定义文档的页眉(介绍信息)，使用示例如下：

```
<header>
<h1>欢迎访问主页</h1>
</header>
```

<footer>标记定义 section 或 document 的页脚。在典型情况下，该元素会包含创作者的姓名、文档的创作日期或者联系信息。使用示例如下：

```
<footer>作者：元澈 联系方式：13012345678</footer>
```

2.2 HTML 5 基本标记详解

HTML 文档最基本的结构主要包括文档类型说明、HTML 文档开始标记、元信息、主体标记和页面注释标记。

2.2.1 文档类型说明

基于 HTML 5 设计准则中的“化繁为简”原则，Web 页面的文档类型说明(DOCTYPE)被极大地简化了。

细心的读者会发现，在第 1 章中使用 Dreamweaver CC 创建 HTML 5 文档时，文档头部的类型说明代码如下：

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

上面为 XHTML 文档类型说明，可以看到，这段代码既麻烦又难记，HTML 5 对文档类型进行了简化，简单到 15 个字符就可以了，代码如下：

```
<!DOCTYPE html>
```



DOCTYPE 声明需要出现在 HTML 5 文件的第一行。

2.2.2 HTML 标记

HTML 标记代表文档的开始。由于 HTML 5 语言语法的松散特性，该标记可以省略，但是为了使之符合 Web 标准和体现文档的完整性，养成良好的编写习惯，这里建议不要省略该标记。

HTML 标记以<html>开头，以</html>结尾，文档的所有内容书写在开头和结尾的中间部分。语法格式如下：

```
<html>
...
</html>
```

2.2.3 头标记 head

头标记 head 用于说明文档头部的相关信息，一般包括标题信息、元信息、定义 CSS 样式和脚本代码等。HTML 的头部信息以<head>开始，以</head>结束，其语法格式如下：

```
<head>
...
</head>
```



<head>元素的作用范围是整篇文档，定义在 HTML 语言头部的内容往往不会在网页上直接显示。

在头标记<head>与</head>之间还可以插入标题标记 title 和元信息标记 meta 等。

1. 标题标记

HTML 页面的标题一般是用来说明页面用途的，它显示在浏览器的标题栏中。在 HTML 文档中，标题信息设置在<head>与</head>之间。标题标记以<title>开始，以</title>结束，其语法格式如下：

```
<title>
...
</title>
```

在标记中间的“...”就是标题的内容，它可以帮助用户更好地识别页面。预览网页时，设置的标题在浏览器的左上方标题栏中显示，如图 2-1 所示。此外，在 Windows 任务栏中显示的也是这个标题。页面的标题只有一个，位于 HTML 文档的头部。



图 2-1 标题栏在浏览器中的显示效果

2. 元信息标记

<meta>元素可提供有关页面的元信息(meta-information)，比如针对搜索引擎和更新频度的描述和关键词。<meta>标记位于文档的头部，不包含任何内容。<meta>标记的属性定义了与文档相关联的名称/值对，<meta>标记提供的属性及其取值如表 2-1 所示。

表 2-1 <meta>标记提供的属性及其取值

属 性	值	描 述
charset	character encoding	定义文档的字符编码
content	some_text	定义与 http-equiv 或 name 属性相关的元信息
http-equiv	content-type expires refresh set-cookie	把 content 属性关联到 HTTP 头部
name	author description keywords generator revised others	把 content 属性关联到一个名称

1) 字符集 charset 属性

在 HTML 5 中，有一个新的 charset 属性，它使字符集的定义更加容易。例如，下面的代码告诉浏览器，网页使用 ISO-8859-1 字符集显示：

```
<meta charset="ISO-8859-1">
```

2) 搜索引擎的关键词

在早期，meta keywords 关键词对搜索引擎的排名算法起到一定的作用，也是很多人进行网页优化的基础。关键词在浏览时是看不到的，其使用的格式如下：

```
<meta name="keywords" content="关键词,keywords" />
```



不同的关键词之间应使用半角逗号隔开(英文输入状态下)，不要使用“空格”或“|”间隔。

是 keywords，不是 keyword。

关键词标签中的内容应该是一个短语，而不是一段话。

例如，定义针对搜索引擎的关键词，代码如下：

```
<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript" />
```

关键词标签 keywords，曾经是搜索引擎排名中很重要的因素，但现在已经被很多搜索引擎完全忽略。如果我们加上这个标签，对网页的综合表现没有坏处。不过，如果使用不恰当的话，对网页非但没有好处，还有欺诈的嫌疑。在使用关键词标签 keywords 时，要注意以下几点。

- 关键词标签中的内容要与网页核心内容相关，应当确信使用的关键词出现在网页文本中。
- 应当使用用户易于通过搜索引擎检索的关键词，过于生僻的词汇不太适合作为 meta

标签中的关键词。

- 不要重复使用关键词，否则可能会被搜索引擎惩罚。
- 一个网页的关键词标签里最多包含 3~5 个最重要的关键词，不要超过 5 个。
- 每个网页的关键词应该不一样。



由于设计者或 SEO 优化者以前对 meta keywords 关键词的滥用，导致目前它在搜索引擎排名中的作用很小。

3) 页面描述

meta description 元标签(描述元标签)是一种 HTML 元标签，用来简略描述网页的主要内容，是通常被搜索引擎用在搜索结果页上展示给最终用户看的一段文字。页面描述在网页中并不显示出来。页面描述使用的格式如下：

```
<meta name="description" content="网页的介绍" />
```

例如，定义对页面的描述，代码如下：

```
<meta name="description" content="免费的 Web 技术教程。" />
```

4) 页面定时跳转

使用<meta>标记可以使网页在经过一定时间后自动刷新，这可通过将 http-equiv 属性值设置为 refresh 来实现。content 属性值可以设置为更新时间。

在浏览网页时经常会看到一些欢迎信息的页面，在经过一段时间后，这些页面会自动转到其他页面，这就是网页的跳转。页面定时刷新跳转的语法格式如下：

```
<meta http-equiv="refresh" content="秒;[url=网址]" />
```



上面的[url=网址]部分是可选项，如果有这部分，页面定时刷新并跳转；如果省略该部分，页面只定时刷新，不进行跳转。

例如，实现每 5 秒刷新一次页面。将下述代码放入 head 标记中即可：

```
<meta http-equiv="refresh" content="5" />
```

2.2.4 网页的主体标记 body

网页所要显示的内容都放在网页的主体标记内，它是 HTML 文件的重点所在。在后面章节所介绍的 HTML 标记都将放在这个标记内。然而，它并不仅仅是一个形式上的标记，它本身也可以控制网页的背景颜色或背景图像，这将在后面进行介绍。主体标记是以<body>开始、以</body>标记结束的，其语法格式如下：

```
<body>
...
</body>
```



在构建 HTML 结构时，标记不允许交错出现，否则会造成错误。

在下列代码中，<body>开始标记出现在<head>标记内，这是错误的：

```
<html>
<head>
<title>标记测试</title>
<body>
</head>
</body>
</html>
```

2.2.5 页面注释标记<!-- -->

注释是在 HTML 代码中插入的描述性文本，用来解释该代码或提示其他信息。注释只出现在代码中，浏览器对注释代码不进行解释，并且在浏览器的页面中不显示。在 HTML 源代码中适当地插入注释语句是一种非常好的习惯，对于设计者日后的代码修改、维护工作很有好处；另外，如果将代码交给其他设计者，他们也能很快读懂该设计者所撰写的内容。

语法：

```
<!--注释的内容-->
```

注释语句元素由前后两半部分组成，前半部分一个左尖括号、一个半角感叹号和两个连字符，后半部分由两个连字符和一个右尖括号组成：

```
<html>
<head>
  <title>标记测试</title>
</head>
<body>
  <!--这里是标题-->
  <h1>HTML 5 网页设计</h1>
</body>
</html>
```

页面注释不但可以对 HTML 中一行或多行代码进行解释说明，而且可以注释掉这些代码。如果希望某些 HTML 代码在浏览器中不显示，可以将这部分内容放在<!--和-->之间。例如，修改上述代码，如下所示：

```
<html>
<head>
  <title>标记测试</title>
</head>
<body>
  <!--
  <h1>HTML 5 网页</h1>
  -->
</body>
</html>
```

修改后的代码将<h1>标记作为注释内容处理，在浏览器中将不会显示这部分内容。

2.3 HTML 5 语法的变化

为了兼容各个不统一的页面代码，HTML 5 的设计在语法方面做了以下变化。

2.3.1 标签不再区分大小写

标签不再区分大小写是 HTML 5 语法变化的重要体现，如以下例子的代码：

```
<!DOCTYPE html>
<html>
<head>
<title>大小写标签</title>
</head>
<body>
<P>这里的标签大小写不一样</p>
</body>
</html>
```

在 IE 11.0 浏览器中预览，效果如图 2-2 所示。

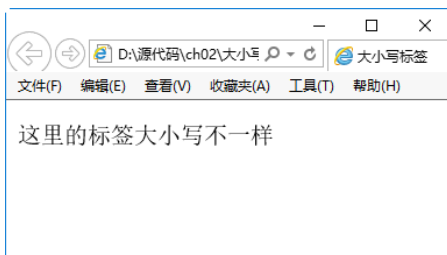


图 2-2 网页预览效果

虽然“<P>这里的标签大小写不一样</p>”中开始标记和结束标记不匹配，但是这完全符合 HTML 5 的规范。用户可以通过 W3C 提供的在线验证页面来测试上面的网页，验证网址为 <http://validator.w3.org/>。

2.3.2 允许属性值不使用引号

在 HTML 5 中，属性值不放在引号中也是正确的，如以下代码片段：

```
<input checked="a" type="checkbox"/>
<input readonly type="text"/>
<input disabled="a" type="text"/>
```

上述代码片段与下面的代码片段效果是一样的：

```
<input checked=a type=checkbox/>
<input readonly type=text/>
<input disabled=a type=text/>
```



尽管 HTML 5 允许属性值可以不使用引号，但是仍然建议读者加上引号。因为如果某个属性的属性值中包含空格等容易引起混淆的属性值，此时可能会引起浏览器的误解。例如以下代码：

```
<img src=mm ch02/01.jpg />
```

此时浏览器就会误以为 src 属性的值就是 mm，这样就无法解析路径中的 01.jpg 图片。如果想正确解析到图片的位置，就必须添加上引号。

2.3.3 允许部分属性的属性值省略

在 HTML 5 中，部分标志性属性的属性值可以省略。例如，以下代码是完全符合 HTML 5 规范的：

```
<input checked type="checkbox"/>
<input readonly type="text"/>
```

其中 checked=“checked”省略为 checked，而 readonly=“readonly”省略为 readonly。在 HTML 5 中，可以省略属性值的属性如表 2-2 所示。

表 2-2 可以省略属性值的属性

属 性	省略属性值
checked	省略属性值后，等价于 checked=“checked”
readonly	省略属性值后，等价于 readonly=“readonly”
defer	省略属性值后，等价于 defer=“defer”
ismap	省略属性值后，等价于 ismap=“ismap”
nohref	省略属性值后，等价于 nohref=“nohref”
noshade	省略属性值后，等价于 noshade=“noshade”
nowrap	省略属性值后，等价于 nowrap=“nowrap”
selected	省略属性值后，等价于 selected=“selected”
disabled	省略属性值后，等价于 disabled=“disabled”
multiple	省略属性值后，等价于 multiple=“multiple”
noresize	省略属性值后，等价于 noresize=“noresize”

2.4 综合案例——符合 W3C 标准的 HTML 5 网页

通过本章的学习，读者了解到 HTML 5 较以前版本有了很大的改变，本章就标记语法部分进行了详细的阐述。

下面制作一个符合 W3C 标准的 HTML 5 网页，具体操作步骤如下。

step 01 启动 Dreamweaver CC，新建 HTML 文档，并单击文档工具栏中的“代码”视图按钮，切换至代码状态，如图 2-3 所示。



图 2-3 使用 Dreamweaver CC 新建 HTML 文档

step 02 图 2-3 中的代码是 XHTML 1.0 格式的, 尽管与 HTML 5 完全兼容, 但是为了简化代码, 可以将其修改成 HTML 5 规范的。修改文档说明部分、<html>标记部分和 <meta>元信息部分, 修改后, HTML 5 结构的代码如下:

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>HTML 5 网页设计</title>
</head>

<body>
</body>
</html>
    
```

step 03 在网页主体中添加内容。例如, 在 body 部分增加如下代码:

```

<!--白居易诗-->
<h1>续座右铭</h1>
<P>
千里始足下,<br>
高山起微尘.<br>
吾道亦如此,<br>
行之贵日新.<br>
</P>
    
```

step 04 保存网页, 在 IE 11.0 中预览, 效果如图 2-4 所示。



图 2-4 网页的预览效果

2.5 跟我学上机——简单的 HTML 5 网页

下面制作一个简单的 HTML 5 网页，具体操作步骤如下。

step 01 打开记事本文件，在其中输入如下代码：

```
<!DOCTYPE html>
<html>
<head>
<title>简单的 HTML 5 网页</title>
</head>
<body>
  <h1>清明</h1>
  <p>
    清明时节雨纷纷,<br>
    路上行人欲断魂。<br>
    借问酒家何处有,<br>
    牧童遥指杏花村。<br>
  </p>
  
</body>
</html>
```

step 02 保存网页，在 IE 11.0 中预览，效果如图 2-5 所示。

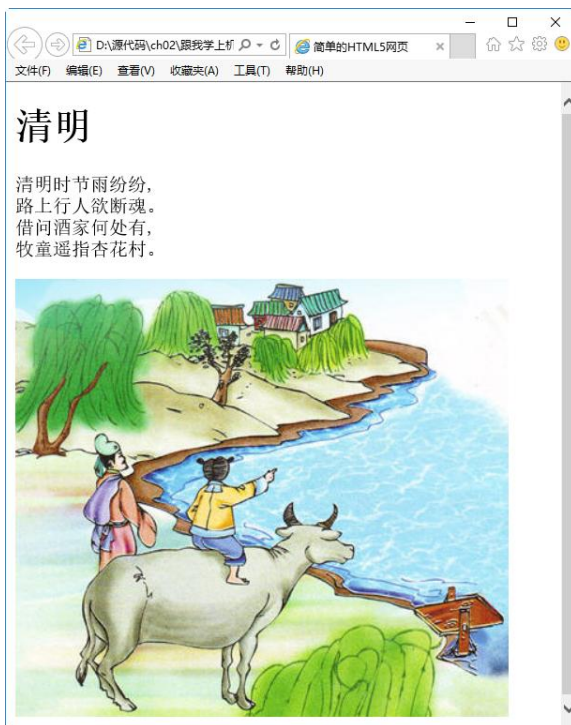


图 2-5 网页的预览效果

2.6 疑难解惑

疑问 1: 在网页中, 语言的编码方式有哪些?

在 HTML 5 网页中, `<meta>` 标记的 `charset` 属性用于设置网页的内码语系, 也就是字符集的类型。对于国内, 经常要显示汉字, 通常设置为 `gb2312`(简体中文)和 `UTF-8` 两种。英文是 `ISO-8859-1` 字符集, 此外还有其他的字符集, 这里不再介绍。

疑问 2: 网页中的基本标签是否必须成对出现?

在 HTML 5 网页中, 大部分标签都是成对出现的。不过也有部分标签可以单独出现, 如 `<p/>`、`
`、``和`<hr/>`等。