



模块 1

使用 HTML5 编写第一个网页

本模块主要介绍 HTML5 的发展、应用,以及如何利用上机环境编写一个简单的 HTML5 页面。

1.1 HTML5 简介

HTML(hyper text markup language)是超文本标记语言,不同于 C 语言、Java 或 C# 等编程语言,它是一种标记语言(markup language),标记语言由一套标记标签(如<html></html>、<head></head>、<title></title>、<body></body>等)组成,HTML 就是使用这些标记标签来描述网页的。

HTML5 是 HTML 最新的修订版本。2014 年 10 月,万维网联盟(World Wide Web Consortium,W3C)完成 HTML5 标准的制定,即 HTML5 是一个标准而不是一种技术。设计 HTML5 的目的是在移动设备上支持多媒体。

1.1.1 HTML5 的概念

HTML5 是新一代 HTML 标准。HTML5 的上一个版本 HTML4.01 诞生于 1999 年。在那以后,Web 世界经历了巨变。目前,HTML5 仍处于完善之中,然而大部分现代浏览器已经支持 HTML5 标准。

1.1.2 HTML5 的发展

为了促进 Web 标准的发展,一些公司联合成立了一个名为 Web 超文本应用技术工作组(Web Hypertext Application Technology Working Group,WHATWG)的组织。WHATWG 致

力于 Web 窗体和应用,而 W3C 的重点在 XHTML 2.0。2006 年,双方决定合作创建新版本的 HTML。

HTML5 的前身名为 Web 应用程序,由 WHATWG 于 2004 年提出,W3C 于 2007 年接受,并成立了一个新的 HTML 工作组。

第一个正式的 HTML5 草案于 2008 年 1 月 22 日发布。目前,大多数浏览器都有一些面向 HTML5 的支持。

2012 年 12 月 17 日,W3C 发布 HTML5 规范的正式草案,该草案凝聚了大量网络工作者的心血。根据 W3C 的声明,HTML5 是开放网络平台的基石。

2013 年 5 月 6 日,HTML 5.1 的正式草案发布。该规范定义了第五个正式版本,第一次要修订万维网的核心语言——超文本标记语言(HTML)。该版本不断地引入新特性和新元素。

2014 年 10 月 29 日,W3C 宣布,经过近 8 年的努力,HTML5 标准终于完成并发布。

HTML5 取代 1999 年建立的 HTML 4.01 和 XHTML1 标准,使其能够满足当代互联网的需要,并在互联网应用程序迅速发展时丰富桌面和移动平台之间无缝连接的内容。

HTML5 也有望成为开放 Web 平台(open Web platform)的基石。例如,其可以进一步促进更深入的跨平台 Web 应用程序的发展。

1.1.3 HTML5 的改进

HTML5 给人们带来了很多人性化和方便的改变,包括新的解析顺序、新的元素、新的方法、新的 input 属性等。

1. 视频

对多媒体的支持可以说是 HTML5 的一大亮点,用户再也不用借助第三方的插件(如 Flash 插件),省去了中间繁杂的配置环节,最主要的是运行环境的变化带来了性能上的提升和资源上的节约。video 元素允许包含多个 source 元素,source 元素可以链接多个不同的视频文件,浏览器会使用第一个可以识别的文件。

2. 音频

对多媒体支持的第二个关键元素则是对音频的支持,使用 audio 标签。其与 video 的使用方法相似,支持的音频文件格式主要包括 wav、mp3、ogg,避免了对 Flash 等外部插件的依赖。

3. 画布

我们可以在 Windows 自带的画图软件中自由地画图,而其中的画布就与此处的画布(canvas)有相似之处。HTML5 中的 canvas 使用 JavaScript 在网页上绘制图形,画布是一个矩形区域,可以控制上面每一个像素的属性。canvas 拥有多种绘制图形、路径、矩阵和添加图形的方法。

4. HTML5 的 Web 存储

HTML5 提供了两种在客户端存储数据的方式:localStorage 和 sessionStorage。

在 HTML4 中,客户端数据的存储主要依靠 cookie 来实现,但 cookie 的缺点是不适合大数据的存储,因为它们的传递依赖于对服务器的请求,这使得 cookie 的速度慢且效率低下。在 HTML5 中,数据不是由每个服务器请求传递的,而是只有在请求时使用数据。它使在不影响网站性能的情况下存储大量的数据成为可能。对于不同的网站,数据存储于不同的区域,并且一个网站只能访问其自身的数据。HTML5 使用 JavaScript 来存储和访问数据。

5. 新的 input 类型

HTML5 提供了新的 input 类型,这些新的类型可以让其对用户输入数据的验证变得更加简单方便,主要有以下几种类型:email、url、number、range、date pickers(date, month, week, time, datetime, datetime-local)、search、color。新的 input 类型提供了新的 email、URL 地址、数字、范围、日期选择、搜索、颜色类型,在提交页面时会自动进行验证。在以前的操作中,主要使用 JavaScript 脚本来验证用户输入数据的合法性。

6. 新的表单类型

HTML5 添加了新的表单类型,主要有以下几种。

(1)datalist。datalist 元素规定输入域的选项列表,列表是通过 datalist 内的 option 元素创建的。

(2)keygen。keygen 元素的作用是提供一种验证用户的可靠方法。

(3)output。output 元素用于不同类型的输出,如计算或脚本输出。

7. 新增加的表单属性

新增加的表单属性包括新的 form 表单和 input 表单的属性,为提升网站整体性能提供了更好的表现。

8. 新增加的语义元素

HTML5 添加了<article>、<aside>等很多语义元素。

1.1.4 HTML5 的应用

据不完全统计,截至 2017 年,超过 80% 的 App 都整合了 HTML5 技术;并且越大的 App 整合 HTML5 的比重越大。微信、淘宝、支付宝、QQ、Facebook、Twitter 等都是典型的例子。

移动网站、轻应用、微网站、手机网页也在迅猛发展,主流媒体和电商平台都已上线移动网站或轻应用版本。

未来 HTML5 必定在“互联网+”的各行各业中获得广泛应用。现有规模巨大的上一代

Web 也必然逐步向 HTML5 迁移。

(1)HTML5 是 HTML、CSS 和 JavaScript 的组合,强化了 Web 网页的表现技能。HTML5 的网页应用如图 1-1 所示。



图 1-1 HTML5 的网页应用

(2)HTML5 可以实现更具结构化、语义化的 Web 文档,方便搜索引擎索引站点。

(3)作为下一代的网页语言,HTML5 拥有很多让人期待的新特性。例如,video 标签让开发者可以在网页中像添加图片一样简单地添加视频。HTML5 能够在移动设备上未安装任何插件的情况下播放多媒体影音,如图 1-2 所示。



图 1-2 HTML5 实现多媒体影音播放

(4)HTML5 是 Web 标准的集合,不仅能够操作前端所见所得,还能操作后台输送到前台的数据,实现离线存储功能。

(5)HTML5 可以实现 Web App 及智能手机 App 应用,如轻应用、微网站、手机网页等,如图 1-3 和图 1-4 所示。



图 1-3 HTML5 实现手机 App 功能

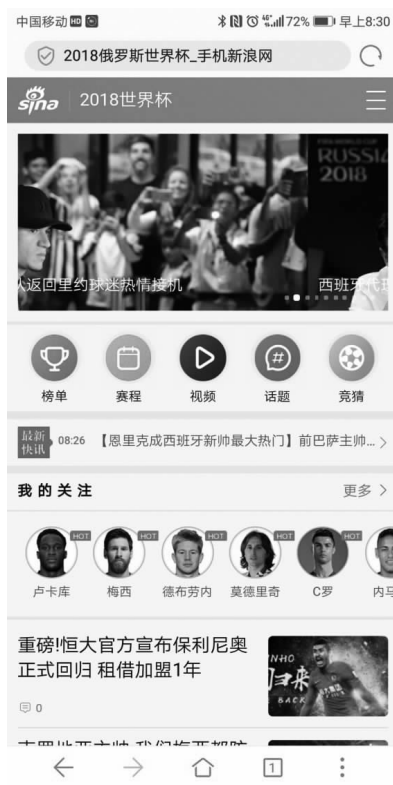


图 1-4 HTML5 实现手机网页功能

(6)HTML5 提供了基于 JavaScript 的 2D 绘图功能和基于 WebGL 的 3D 绘图功能,操作网页游戏无须任何插件,如图 1-5 所示。

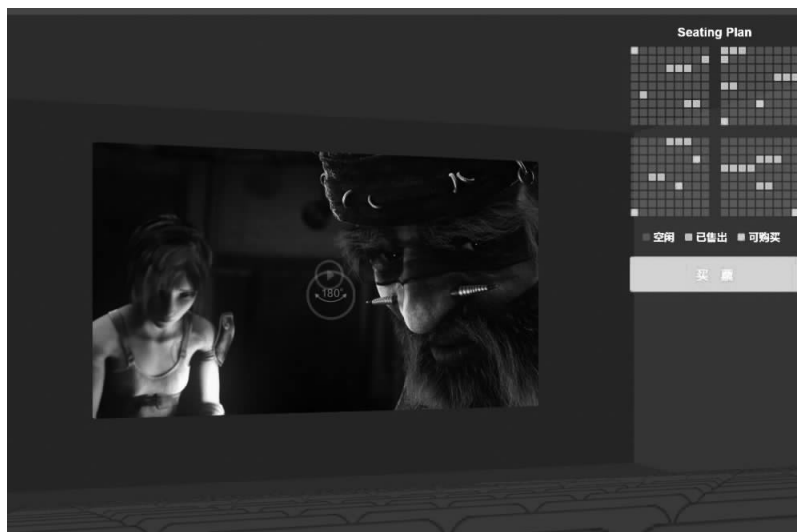


图 1-5 HTML5 实现网页游戏功能

(7)HTML5 可以提供更多的 CSS 属性,可以做出更加丰富的渲染效果。

(8)HTML5 能快速地构建框架,使实现一些功能更加简便,节省开发时间,如图 1-6 所示。



图 1-6 用 HTML5 实现购物网站的搭建

(9)HTML5 图表方案可以更加直观地在网站中用可视化的方式显现大量的数据及信息。

1.2 实现你的第一个 HTML5 页面

在本节中,我们通过实现一个简单的 HTML5 页面,介绍如何在计算机中搭建上机环境 Dreamweaver CC 2017。Dreamweaver CC 2017 的一些初始化设置支持 HTML5 的浏览器、站点的建立和页面的预览等内容。

1.2.1 准备上机环境 Dreamweaver CC 2017

本书采用 Dreamweaver CC 2017 版本软件,双击文件夹中的安装程序(setup)进行安装,安装过程中需要登录(没有 Adobe 账户的用户需要先行注册),登录后安装程序会自动运行。接受软件的许可协议后,会进入软件的第一次使用界面,设置软件的工作区和主题后,即可进入软件主界面,如图 1-7~图 1-10 所示。



图 1-7 登录界面



图 1-8 安装过程提示



图 1-9 软件许可协议界面



图 1-10 软件的初始化设置

1.2.2 支持 HTML5 的浏览器

支持 HTML5 的浏览器有 Firefox、IE 10 和 IE 10 以上版本, Chrome(谷歌浏览器)、Opera、Safari 等; Maxthon Browser(傲游)、360 浏览器、QQ 浏览器、搜狗浏览器、猎豹浏览器等国产浏览器也有支持 HTML5 的能力。

- (1)IE。IE 10 以上版本支持。
- (2)Firefox。Firefox 4.0 以上版本支持。
- (3)Chrome。Chrome 6 以上版本支持。
- (4)Opera。Opera 10.6 以上版本支持。
- (5)Safari。Safari 3.1 以上版本支持大部分。

本书采用的浏览器版本为 IE 11。安装了相应的浏览器后, 为了进一步了解浏览器支持 HTML5 的情况, 还可以通过编写 JavaScript 代码来检测浏览器是否支持。下面以加入画布标记为例检测浏览器的支持情况。在 HTML 页面中插入一段 HTML5 画布标记, 当浏览器支持时, 则会出现一个矩形, 否则将会在页面中显示“您的浏览器不支持 HTML5 画布”。其代码如下。


```
<!doctype html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;charset=utf-8">
<title>浏览器测试</title>
<style type="text/css">
#myCanvas{
background:blue;
width:100px;
height:100px;
}
</style>
</head>
<body>
<canvas id="myCanvas">您的浏览器不支持 HTML5 画布</canvas>
</body>
</html>
```

在 Dreamweaver CC 2017 中运行后,生成 HTML 文件,在 IE 8 中执行页面文件,由于 IE 8 不支持 HTML5 的画布标记,因而在浏览器中将显示“您的浏览器不支持 HTML5 画布”,如图 1-11 所示;如果在 IE 11 中执行,由于 IE 11 支持 HTML5 的画布标记,所以在浏览器中将显示一个蓝色的正方形,如图 1-12 所示。



图 1-11 IE 8 浏览器测试结果



图 1-12 IE 11 浏览器测试结果

1.2.3 建立站点和编写第一个网页

1. 建立站点

(1) 在计算机中新建一个文件夹。例如,在 D 盘建立一个名为“测试”的文件夹,如图 1-13 所示,将所有相关网页文件和文件夹都存储在该新建的文件夹中。



图 1-13 新建的“测试”文件夹

(2) 启动 Dreamweaver, 选择“站点”→“新建站点”菜单命令, 在弹出的对话框的“站点名称”文本框中输入站点名称为“测试”, 设置“本地站点文件夹”为在 D 盘中新建的“测试”文件夹, 如图 1-14 所示。

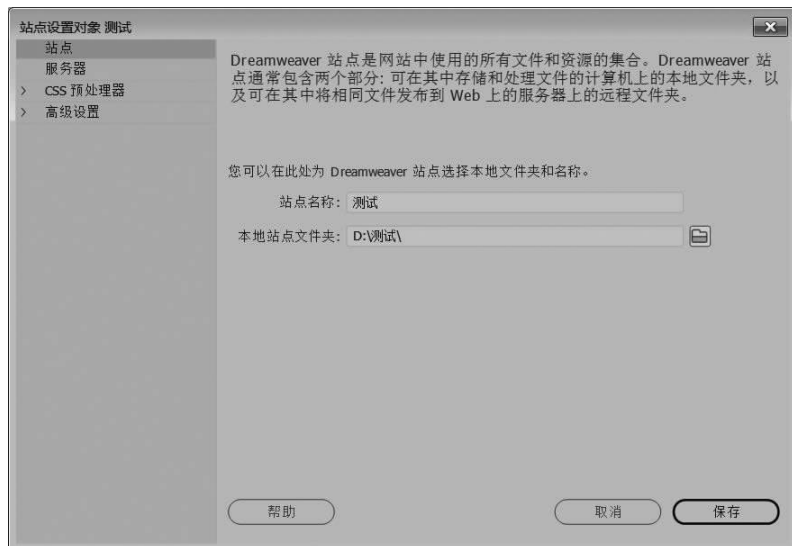


图 1-14 新建站点

2. 编写第一个网页

(1) 启动 Dreamweaver, 选择“文件”→“新建”菜单命令, 弹出“新建文档”对话框(默认选中 HTML 文档类型), 在“标题”文本框中输入“我的第一个网页”, 在“文档类型”下拉列表框中选择 HTML5, 如图 1-15 所示。



图 1-15 新建 HTML 文档

(2) 在新建的文档中选择“拆分”模式(默认),主页面被分成上下两部分,上半部分为页面预览,下半部分为代码部分,如图 1-16 所示。

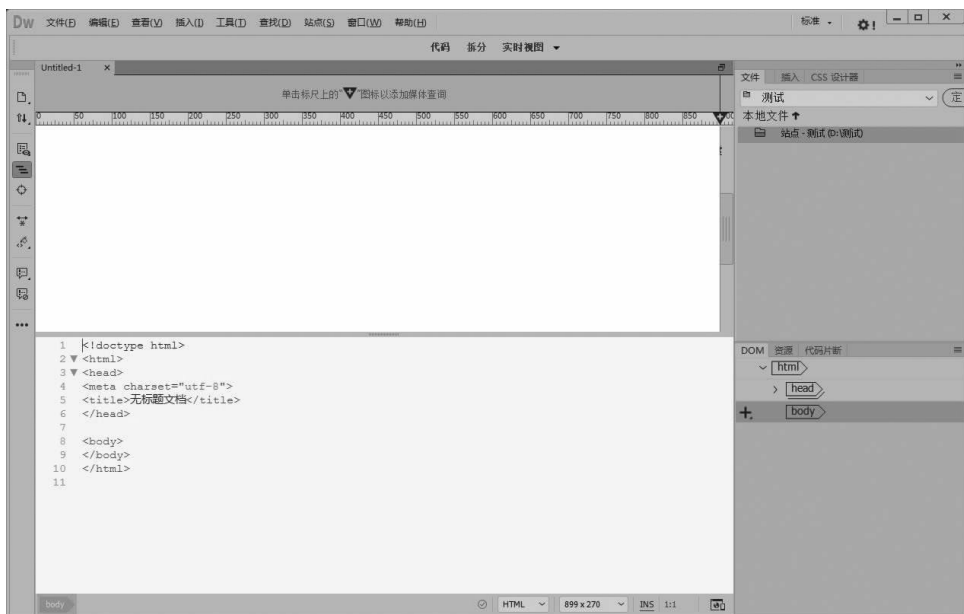


图 1-16 新建的 HTML 文档界面

(3) 新建文档中有默认的代码,共 10 行,在<body></body>之间插入代码,在代码中添加文字,如图 1-17 所示。

```
<h1>我的第一个网页</h1>
<p>我的第一个段落</p>
```

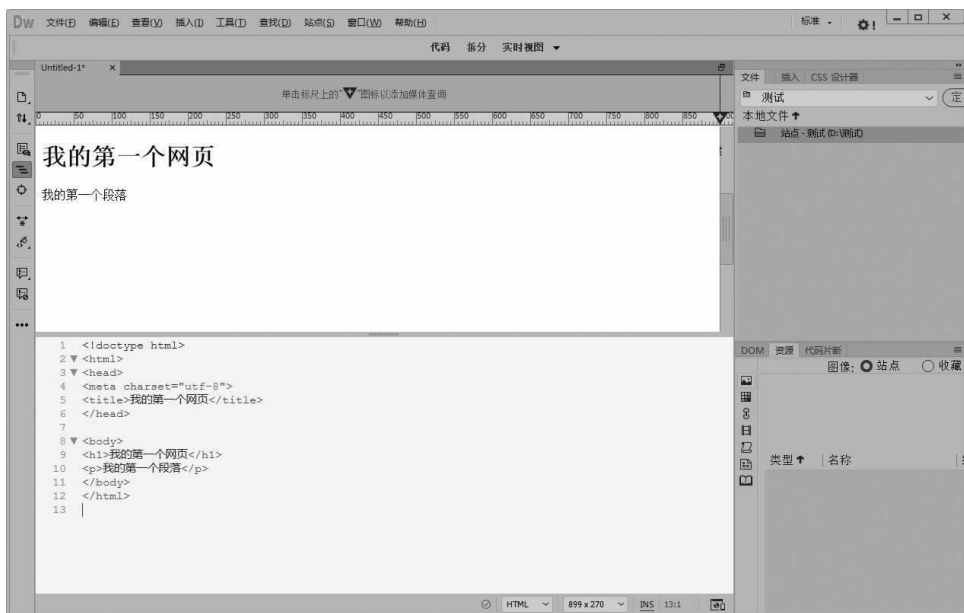


图 1-17 在代码中添加文字

(4)保存页面。选择“文件”→“保存”菜单命令,在弹出的“另存为”对话框中选择将编辑好的文件存储到 D 盘“测试”文件夹中,并更改文件名为“我的第一个网页”,如图 1-18 所示。



图 1-18 保存页面

(5)预览页面。打开 D 盘“测试”文件夹,用 IE 11 打开“我的第一个网页.html”文件,效果如图 1-19 所示。

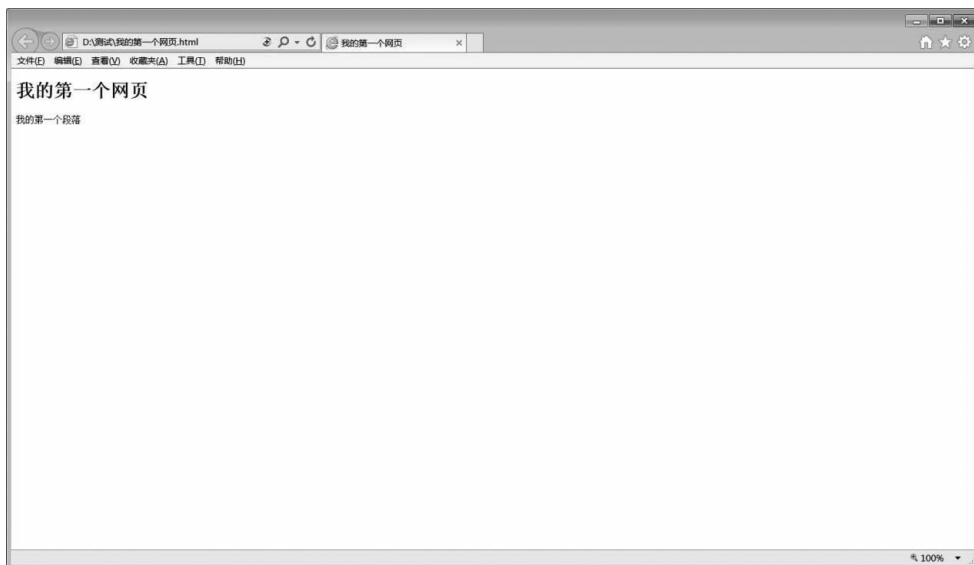


图 1-19 用 IE 11 预览页面的效果

1.3 案例

【案例描述】

学习完本模块的内容,小张同学已经迫不及待地想做一个网页,以展示自己的学习成果。小张同学希望做一个能显示“让我们一起开始 HTML5 语言的旅程吧!”的网页。

【源代码展示】

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>制作的第一个 HTML5 文件</title>
</head>
<body>
  让我们一起开始 HTML5 语言的旅程吧!
</body>
</html>
```

【运行结果】

运行结果如图 1-20 所示。



图 1-20 模块 1 案例运行结果

【程序分析】

小张同学结合本模块所学的知识完成了网页的设计。在设计中,小张同学运用<title></title>标签定义了页面标题,使用<body></body>标签定义了页面内容。

1.4 思考与练习

一、填空题

1. HTML5 给人们带来了很多人性化和方便的变化,包括新的解析顺序、_____、_____,新的 input 属性等。
2. HTML5 设计的目的是在_____支持多媒体。

二、选择题

1. 网页的基本语言是()。
A. JavaScript B. VBScript C. HTML D. XML
2. 借助网页中所谓的(),通过已经定义好的关键字和图形,只要用鼠标轻轻一点,就可以自动跳转到相应的其他文件,获得相应的信息。
A. 超链接 B. 热区 C. 图片 D. 多媒体
3. 网页的扩展名是()。
A. .html B. .doc C. .bat D. .ppt
4. HTML 指的是()。
A. 超文本标记语言(hyper text markup language)
B. 家庭工具标记语言(home tool markup language)
C. 超链接和文本标记语言(hyperlinks and text markup language)
D. 以上都不正确

三、判断题

1. HTML5 标记符的属性一般不区分大小写。 ()
2. HTML5 能快速地构建框架,更加简便地实现一些功能,节省开发时间。 ()

2

模块 2

从 HTML 到 HTML5 的过渡

通过模块 1 我们初步了解了 HTML5 的基础知识,并学会了如何创建一个简单的 HTML5 网页。本模块将介绍 HTML 的基础知识和 HTML5 的新元素。

2.1 HTML 简介

HTML 是目前网络上应用广泛的一种语言,也是构成网页文档的主要语言。

用 HTML 编写的超文本文档称为 HTML 文档,HTML 文档主要包括由 HTML 标签组成的文本、图形、声音、动画、超链接等。它能独立于各种操作系统平台(如 UNIX、Windows 等)。HTML 是一种建立网页文件的语言,通过标记式的指令,将声音、图片、文字、动画、视频等内容显示出来。它可以从一个文件跳转到另一个文件,与世界各地主机中的文件链接。HTML 的普遍应用就是超文本的技术,即通过单击鼠标从一个主题跳转到另一个主题,从一个页面跳转到另一个页面,与世界各地主机中的文件链接。

2.1.1 HTML 网页结构

一个完整的 HTML 文档至少由以下四部分组成。

- (1)文档声明: `<!doctype html>`。
- (2)html 标签对: `<html></html>`。
- (3)head 标签对: `<head></head>`。
- (4)body 标签对: `<body></body>`。

首先应该对文档进行声明,使用 `<!doctype html>`,声明文档为 HTML 文档。声明有助于浏览器正确显示网页。网络上有很多不同的文件,正确声明 HTML 的版本,浏览器就

能正确显示网页内容。声明是不区分大小写的,以下方式均可: <!DOCTYPE html>、<!DOCTYPE HTML>、<!doctype html>、<!Doctype Html>。声明位于文档中最前面的位置,处于 <html> 标签之前。它不是一个 HTML 标签,是用来告知 Web 浏览器页面使用了哪种 HTML 版本。HTML 4.01 规定了三种不同的 <!DOCTYPE> 声明,分别是 Strict、Transitional 和 Frameset。HTML5 中仅规定了一种,即 <!doctype html>。

无论是动态页面还是静态页面,都是以“<html>”开始,以“</html>”结尾的。“<html>”后接着是“<head>”,<head></head> 中的内容在浏览器页面中是无法显示的,在“<title></title>”中放置的是网页标题。接着就是正文“<body></body>”,也就是常说的 body 区,这里放置的内容通过浏览器页面呈现给用户,即网页的内容呈现区。最后以“</html>”结尾,也就是网页闭合。以上是一个完整的结构,还可以再增加更多的样式和内容来充实网页。

2.1.2 HTML 实例解析

对 HTML 文档进行声明后,文档中的元素以开始标签起始,并以结束标签终止,元素的内容是开始标签与结束标签之间的内容。下面通过一个简单的实例解析 HTML 文档。

1. HTML 文档代码

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>我的第一个页面</title>
</head>
<body>
  <p>我的第一个段落</p>
</body>
</html>
```

运行结果如图 2-1 所示。

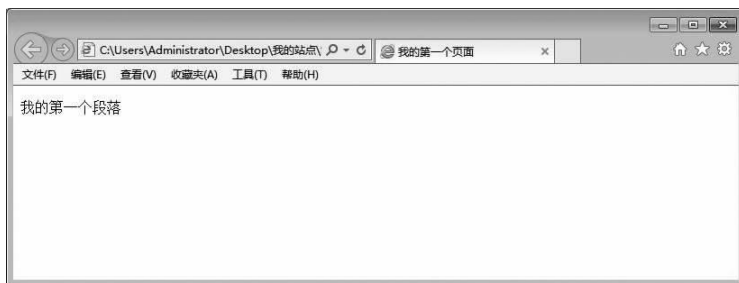


图 2-1 HTML 文档实例运行结果

2. HTML 文档解析

1) <p>元素

```
<p>我的第一个段落</p>
```

<p>元素定义了 HTML 文档中的一个段落。该元素拥有一个开始标签 <p> 和一个结束标签 </p>。元素内容是：我的第一个段落。

2) <body>元素

```
<body>  
  <p>我的第一个段落</p>  
</body>
```

<body>元素定义了 HTML 文档的主体。该元素拥有一个开始标签 <body> 和一个结束标签 </body>。元素内容是另一个 HTML 元素(p 元素)。

3) <title>元素

```
<title>我的第一个页面</title>
```

<title>元素定义了页面的标题,该元素拥有一个开始标签 <title> 和一个结束标签 </title>。元素内容是:我的第一个页面。

4) <head>元素

```
<head>  
  <meta charset="utf-8">  
  <title>我的第一个页面</title>  
</head>
```

<head>元素定义了 HTML 文档信息。该元素拥有一个开始标签 <head> 和一个结束标签 </head>。元素内容是另一个 HTML 元素(title 元素)。

5) <html>元素

```
<html>  
  <head>  
    <meta charset="utf-8">  
    <title>我的第一个页面</title>  
  </head>  
  <body>  
    <p>我的第一个段落</p>  
  </body>  
</html>
```

<html>元素定义了整个 HTML 文档。该元素拥有一个开始标签 <html> 和一个结

束标签</html>。元素内容是另外的 HTML 元素(head 元素和 body 元素)。

2.1.3 HTML 标签与元素

HTML 标签是 HTML 中最基本的单位, HTML 标签不区分大小写。例如,“主体”<body>与<BODY>表示的意思是相同的,不过,推荐使用小写。HTML 文档是由 HTML 元素定义的,HTML 元素指的是从开始标签到结束标签的所有代码。下面通过一个 HTML 文档实例来了解 HTML 的标签与元素。

1. HTML 实例

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>我的第一个网页</title>
</head>
<body>
  <h1>我的第一个网页</h1>
</body>
</html>
```

上面的例子包含五个 HTML 元素。

2. HTML 实例解释

1) <h1>元素

```
<h1>我的第一个网页</h1>
```

<h1>元素定义了 HTML 标题。该元素拥有一个开始标签 <h1> 和一个结束标签 </h1>。元素内容是:我的第一个网页。

2) <body>元素

```
<body>
  <h1>我的第一个网页</h1>
</body>
```

<body>元素定义了 HTML 文档的主体。该元素拥有一个开始标签<body> 和一个结束标签</body>。元素内容是另一个 HTML 元素(h1 元素)。

3) <title>元素

```
<title>我的第一个网页</title>
```

<title>元素定义了 HTML 页面的标题。该元素拥有一个开始标签<title> 和一个结

束标签</title>。元素内容是:我的第一个网页。

4) <head> 元素

```
<head>
  <meta charset="utf-8">
  <title>我的第一个网页</title>
</head>
```

<head>元素定义了关于文档的信息。该元素拥有一个开始标签<head>和一个结束标签</head>。

5) <html> 元素

```
<html>
<head>
  <meta charset="utf-8">
  <title>我的第一个网页</title>
</head>
<body>
  <h1>我的第一个网页</h1>
</body>
</html>
```

<html>元素定义了整个 HTML 文档。该元素拥有一个开始标签 <html> 和一个结束标签 </html>。

2.1.4 HTML 浏览器的中文编码

目前,在大部分浏览器的 HTML 文档中直接输出中文会出现中文乱码的情况,这时就需要在头部将字符声明为 utf-8。

先看下面一个例子,代码如下。

```
<!doctype html>
<html>
<body>
  <p>汉字显示正常</p>
</body>
</html>
```

其在 Dreamweaver CC 中运行的结果如图 2-2 所示,在预览的过程中,中文编码变成了乱码。如果希望中文编码能正常显示,需要在 HTML 文件头部的“content-type”中设置“charset”,告诉浏览器打开该文件的编码方式。其代码如下。

```
<html>
<meta charset="utf-8"/>
<head>
```

HTML 浏览器支持的中文编码有两种方式：一种是“charset=utf-8”，另一种是“charset=gbk”。运行情况如图 2-3 所示。

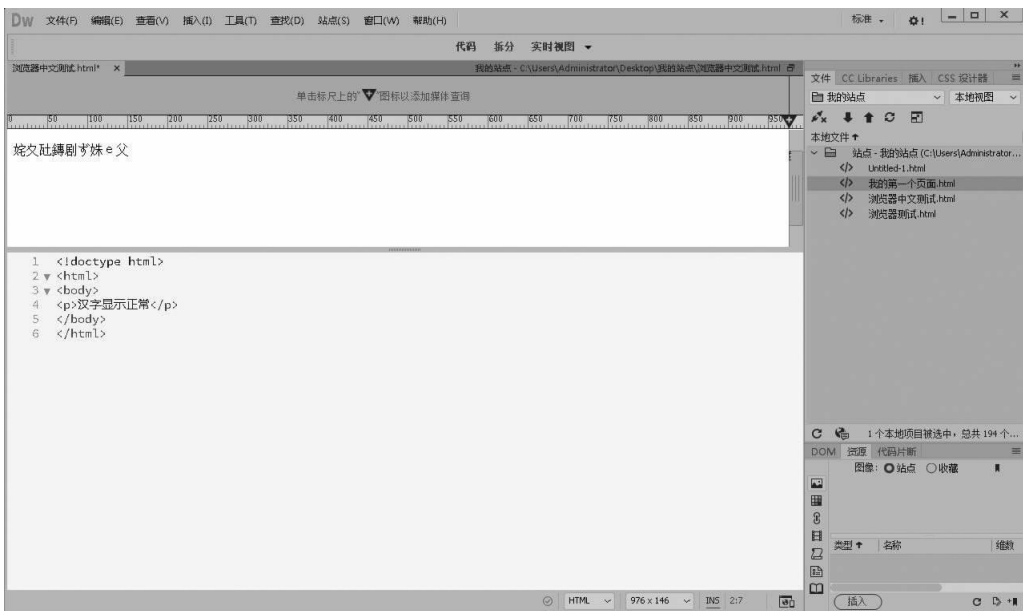


图 2-2 中文编码运行异常

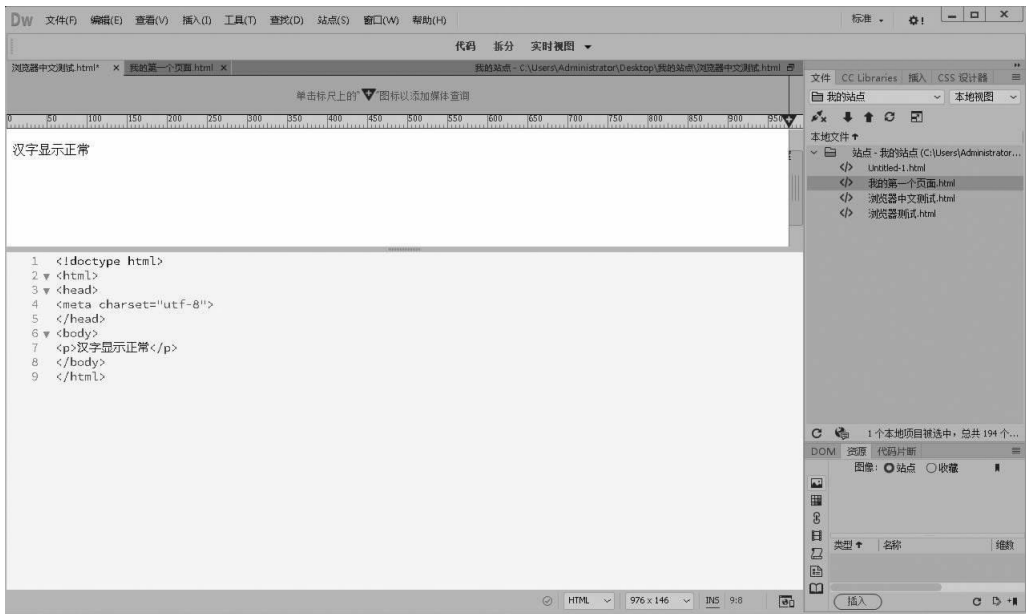


图 2-3 中文编码运行正常

页面源文件不是乱码,但是在浏览器中显示乱码,可以通过修改浏览器的编码方式看到正确的中文显示。HTML 文档中设定的中文编码格式为 utf-8(<meta charset="utf-8">),在浏览器中也要选择“Unicode(UTF-8)”编码;如果 HTML 文档中设定的中文编码格式为“charset=gbk”,在浏览器中要选择“简体中文(GB2312)”编码。浏览器编码设置如图 2-4 所示。



图 2-4 浏览器编码设置

2.2 HTML5 新元素

2.2.1 HTML5 新元素简介

有一部分元素在 HTML 4.01 中已经被废弃,这些元素在 HTML5 中已经被移除或重新定义。为了更好地处理今天的互联网应用,HTML5 新增了很多新元素及功能,如图形的绘制元素、多媒体媒介元素、用来创建更好的页面结构的新的语义和结构元素等。

下面介绍 HTML5 定义的新元素。

1. <canvas> 标签

<canvas> 标签用于定义图形, 如图表和其他图像。该标签基于 JavaScript 的绘图 API。

2. 新的媒介元素

新的媒介元素见表 2-1。

表 2-1 新的媒介元素

| 标 签 | 描 述 |
|----------|-----------------------------------|
| <audio> | 定义音频内容 |
| <video> | 定义视频(video 或 movie) |
| <source> | 定义多媒体资源<video>和<audio> |
| <embed> | 定义嵌入的内容, 如插件 |
| <track> | 为诸如<video>和<audio>元素之类的媒介规定外部文本轨道 |

3. 新的表单元素

新的表单元素见表 2-2。

表 2-2 新的表单元素

| 标 签 | 描 述 |
|------------|-------------------------------------|
| <datalist> | 定义选项列表。与 input 元素配合使用来定义 input 可能的值 |
| <keygen> | 规定用于表单的密钥对生成器字段 |
| <output> | 定义不同类型的输出, 如脚本的输出 |

4. 新的语义/结构元素

新的语义/结构元素见表 2-3。

表 2-3 新的语义/结构元素

| 标 签 | 描 述 |
|-----------|---------------------------|
| <article> | 定义页面独立的内容区域 |
| <aside> | 定义页面的侧边栏内容 |
| <bdi> | 允许设置一段文本, 使其脱离其父元素的文本方向设置 |
| <command> | 定义命令按钮, 如单选按钮、复选框或按钮 |
| <details> | 用于描述文档或文档某个部分的细节 |
| <dialog> | 定义对话框, 如提示框 |

| 标 签 | 描 述 |
|--------------|--------------------------------------|
| <summary> | 标签包含<details>元素的标题 |
| <figure> | 规定独立的流内容(图像、图表、照片、代码等) |
| <figcaption> | 定义<figure>元素的标题 |
| <footer> | 定义 section 或 document 的页脚 |
| <header> | 定义文档的头部区域 |
| <mark> | 定义带有记号的文本 |
| <meter> | 定义度量衡。仅用于已知最大值和最小值的度量 |
| <nav> | 定义导航链接的部分 |
| <progress> | 定义任何类型的任务的进度 |
| <ruby> | 定义 ruby 注释(中文注音或字符) |
| <rt> | 定义字符(中文注音或字符)的解释或发音 |
| <rp> | 在 ruby 注释中使用,定义不支持 ruby 元素的浏览器所显示的内容 |
| <section> | 定义文档中的节(section、区段) |
| <time> | 定义日期或时间 |
| <wbr> | 规定在文本中的何处适合添加换行符 |

2.2.2 绘制图形元素

HTML5 用<canvas>标签定义图形,可以通过多种方法使用<canvas>绘制路径、盒、圆、字符及添加图像。

1. 创建一个矩形

创建一个画布(canvas),画布在网页中是一个矩形框,通过<canvas>元素来绘制。默认情况下,<canvas>元素没有边框和内容。

<canvas>简单实例如下。

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

<canvas>标签通常需要指定一个 id 属性, width 和 height 属性定义画布的大小,

style 属性定义线条宽度和线条颜色。在 HTML 页面中允许使用多个 <canvas> 元素。运行情况如图 2-5 所示。



图 2-5 绘制矩形

2. 创建一个红色矩形

<canvas> 元素本身是没有绘图能力的，所有的绘图工作必须在 JavaScript 内部完成。例如，要创建一个红色的矩形，代码如下。

```

var c=document.getElementById("myCanvas");    /* 找到<canvas>元素 */
var ctx=c.getContext("2d");                  /* 创建 context 对象 */
ctx.fillStyle="#FF0000";                    /* 定义线条颜色 */
ctx.fillRect(0,0,150,75);                   /* 定义矩形当前的填充方式 */
    
```

运行情况如图 2-6 所示。

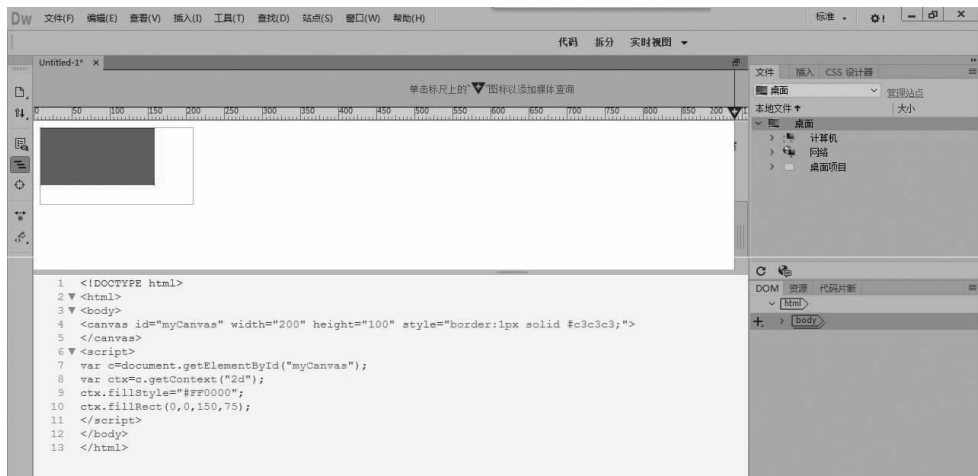


图 2-6 绘制红色矩形

3. 创建一条直线

在 canvas 中画线,我们经常使用 `moveTo(x,y)` 方法定义线条开始坐标,使用 `lineTo(x,y)` 方法定义线条结束坐标,绘制线条可以用 `stroke()` 方法。

例如,定义开始坐标(0,0)和结束坐标(200,100),然后使用 `stroke()` 方法绘制线条,代码如下。

```
var c=document.getElementById("myCanvas");    /* 找到 <canvas> 元素 */
var ctx=c.getContext("2d");                    /* 创建 context 对象 */
ctx.moveTo(0,0);                               /* 定义线条起始位置 */
ctx.lineTo(200,100);                           /* 定义线条结束位置 */
ctx.stroke();                                   /* 开始绘图 */
```

运行情况如图 2-7 所示。

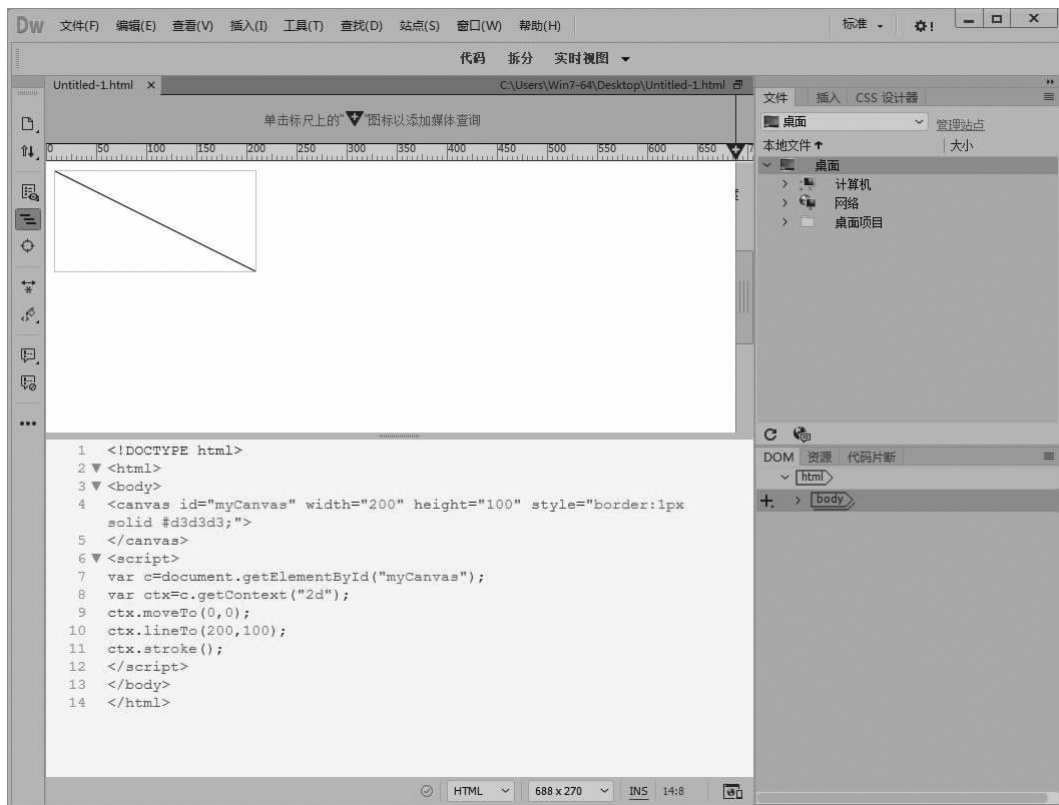


图 2-7 绘制直线

4. 创建一个简单的圆形

在 canvas 中绘制圆形,我们使用 `arc(x,y,r,start,stop)` 方法,实际上在绘制圆形时使用了“ink”的方法,如 `stroke()` 或 `fill()`。

例如,使用 `arc()` 方法绘制一个圆,代码如下。

```

var c=document.getElementById("myCanvas");    /* 找到 <canvas>元素 */
var ctx=c.getContext("2d");                  /* 创建 context 对象 */
ctx.beginPath();                             /* 开启绘画状态 */
ctx.arc(95,50,40,0,2*Math.PI);              /* 定义圆心坐标、半径、起始角等 */
ctx.stroke();                                /* 开始绘图 */

```

运行情况如图 2-8 所示。

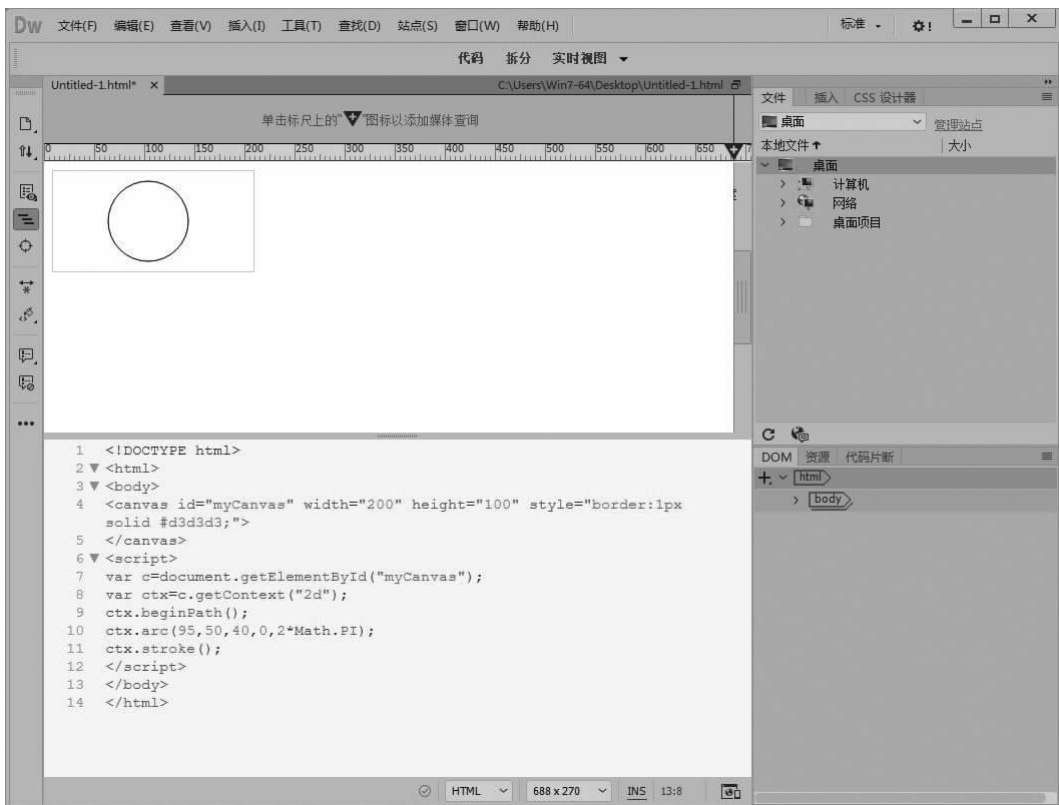


图 2-8 绘制圆形

2.2.3 多媒体元素

HTML 多媒体元素是指音效、音乐、视频和动画等。多媒体具有多种不同的格式。它可以是听到或看到的任何内容,如文字、图片、音乐、音效、录音、电影、动画等。在互联网上,人们会经常发现嵌入网页中的多媒体元素,现代浏览器已支持多种多媒体格式。在本书中,读者将了解到不同的多媒体格式,以及如何在网页中使用它们。

1. 浏览器的支持

第一款互联网浏览器只支持文本,而且即使是对文本的支持也仅限于单一字体和单一颜色。随后诞生了支持多种颜色、字体和文本样式的浏览器,图片支持功能也被加入。不同

的浏览器以不同的方式处理对音效、动画和视频的支持。某些元素能够以内联的方式处理，而某些元素则需要额外的插件。

2. 多媒体格式

确定媒体类型的最常用的方法是查看文件扩展名。当浏览器得到的文件扩展名为 .htm 或 .html 时，它会假定该文件是 HTML 页面。多媒体元素也拥有带有不同扩展名的文件格式，如 .swf、.wmv、.mp3 及 .mp4。

1) 常见的视频文件格式

常见的视频文件格式见表 2-4。

表 2-4 常见的视频文件格式

| 格 式 | 文件扩展名 | 描 述 |
|-----------|---------------|--|
| AVI | .avi | AVI 格式是由微软开发的。所有运行 Windows 的计算机都支持 AVI 格式。它是互联网上很常见的格式，但非 Windows 计算机并不总是能够播放它 |
| WMV | .wmv | WMV(Windows media video) 格式是由微软开发的。WMV 在互联网上很常见，但是如果未安装额外的(免费)组件，就无法播放 |
| MPEG | .mpg .mpeg | MPEG (moving pictures expert group) 格式是互联网上最流行的格式。它是跨平台的，得到了所有流行浏览器的支持 |
| QuickTime | .mov | QuickTime 格式是由苹果公司开发的。QuickTime 是互联网上常见的格式，但是 QuickTime 视频不能在没有安装额外的(免费)组件的 Windows 计算机上播放 |
| RealVideo | .rm .ram | RealVideo 格式是由 Real Media 针对互联网开发的。该格式允许低带宽条件下(在线视频、网络电视)的视频流。由于其是低带宽优先的，质量常会降低 |
| Flash | .swf .flv | Flash (Shockwave) 格式是由 Macromedia 开发的。Shockwave 格式需要额外的组件来播放。该组件会预装到 Firefox 或 IE 之类的浏览器上 |
| Mpeg-4 | .mp4 | Mpeg-4(MP4)是一种针对互联网的新格式。事实上，YouTube 推荐使用 MP4。YouTube 接收多种格式视频，然后将其全部转换为 .flv 或 .mp4 格式以供分发。越来越多的视频发布者转到 MP4，将其作为 Flash 播放器和 HTML5 的互联网共享格式 |

2) 常见的声音文件格式

MP3 是一种音频压缩技术，其全称是动态影像专家压缩标准音频层面 3 (moving picture experts group audio layer III)，简称 MP3。它被设计用来大幅度地降低音频数据量。如果你的站点是音乐类型的，可以选择 mp3 格式。常见的声音文件格式见表 2-5。

表 2-5 常见的声音文件格式

| 格 式 | 文件扩展名 | 描 述 |
|------|---------------|---|
| Wave | .wav | Wave (waveform) 格式是由 IBM 和微软开发的。所有运行 Windows 的计算机和除了 Google Chrome 之外的所有网络浏览器都支持它 |
| WMA | .wma | WMA(Windows media audio)格式的质量优于 MP3,兼容大多数播放器,除了 iPod。WMA 文件可作为连续的数据流来传输,这使其对于网络电台或在线音乐很实用 |
| MP3 | .mp3 .mpga | MP3 文件实际上是 MPEG 文件的声音部分。MPEG 格式最初是由运动图像专家组开发的。MP3 是其中最受欢迎的针对音乐的声音格式 |

2.2.4 表单元素

HTML5 定义了新的表单元素: <datalist>、<keygen> 和 <output>。并不是所有的浏览器都支持 HTML5 新的表单元素,但是可以使用它们;即使部分浏览器不支持表单属性,仍然可以将其显示为常规的表单元素。

1. <datalist> 元素

<datalist> 规定 form 或 input 域应该拥有自动完成功能。当用户在自动完成域中开始输入时,浏览器应该在该域中显示填写的选项,使用 <input> 元素的列表属性与 <datalist> 元素绑定。

【例 2-1】 <input> 元素使用 <datalist> 预定义值。

```
<!doctype html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
</head>
<body>
  <form action="demo-form.php" method="get">
    <input list="browsers" name="browser">
      <datalist id="browsers">
        <option value="Internet Explorer">
          </datalist>
      <input type="submit">
    </form>
  </body>
</html>
```

运行页面文件,结果如图 2-9 所示。



图 2-9 使用<datalist>元素创建表单

2. <keygen>元素

<keygen>元素的作用是提供一种验证用户的可靠方法,规定用于表单的密钥对生成器字段,当提交表单时,会生成两个键:一个是私钥,另一个是公钥。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。

例如,带有 keygen 字段的表单如下。

```
<!doctype html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
</head>
<body>
<form action="demo_keygen.php" method="get">
  用户名: <input type="text" name="usr_name">
  加密: <keygen name="security">
  <input type="submit">
</form>
</body>
</html>
```

IE 11 不支持<keygen>标签,将页面文件在 Chrome 浏览器中运行,结果如图 2-10 所示。



图 2-10 使用<keygen>元素创建表单

3. <output>元素

<output>元素用于不同类型元素的输出,如计算或脚本输出,代码如下。

```
<!doctype html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
</head>
<body>
  <form oninput="x.value=parseInt(a.value)+parseInt(b.value)">0
    <input type="range" id="a" value="50">100
    +<input type="number" id="b" value="50">
    =<output name="x" for="a b"></output>
  </form>
</body>
</html>
```

IE 11 同样不支持<output>元素,将页面文件在 Chrome 浏览器中运行,结果如图 2-11 所示。



图 2-11 使用<output>元素创建表单

2.2.5 语义和结构元素

一个语义元素能够给浏览器和开发者清楚地描述其语法的意义。例如，<div>和为无语义元素，<form>、<table>和清楚地定义了其内容为语义元素。

HTML 以前只能用<div.../>来定义结构元素，在 HTML5 中，新增加了<header>、<nav>、<section>、<article>、<aside>等文档结构元素来明确 Web 页面的不同部分。HTML5 中新定义的结构元素详见 2.2.1 节中 HTML5 新增的结构元素表，下面用一个简单的实例来介绍新的结构元素的使用。目前，大多数页面还都是使用<div>来定义页面结构的，如苹果公司官网(美国)主页的代码和页面如图 2-12 和图 2-13 所示。

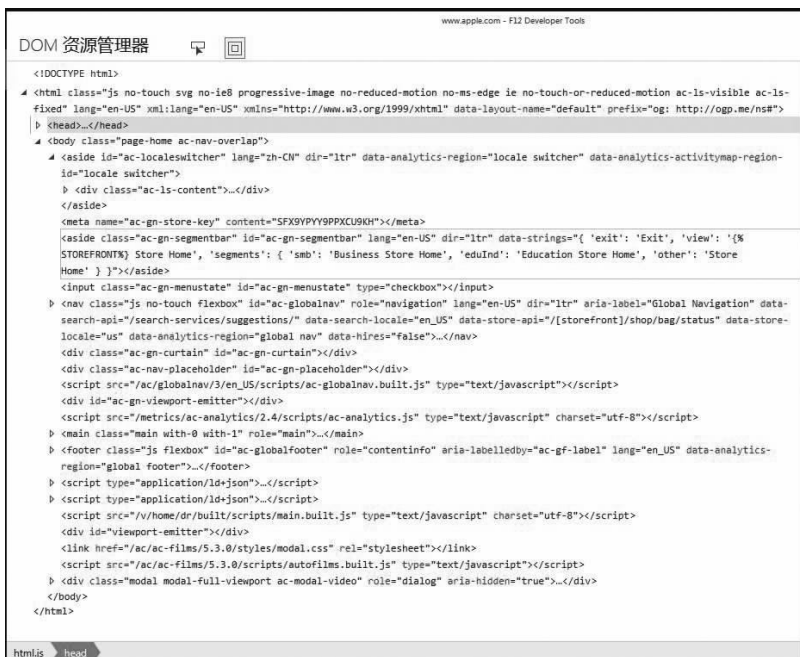


图 2-12 苹果公司官网(美国)主页的代码

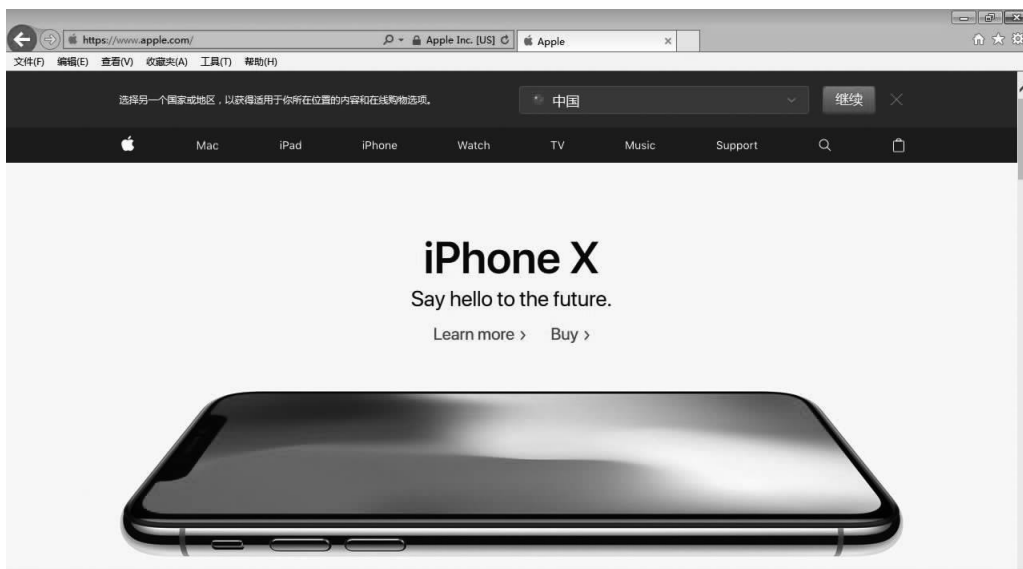


图 2-13 苹果公司(美国)官网主页面

1. <section>元素

<section>标签定义文档中的节(section、区段),如章节、页眉、页脚或文档中的其他部分。

【例 2-2】 <section>实例。

```

<!doctype html>
<html>
  <meta charset="utf-8">
  <body>
    <section>
      <h1>HTML5</h1>
      <p>HTML5 是 HTML 最新的修订版本,谈起 HTML 就必须了解什么是 HTML。HTML(Hyper Text Markup Language)是超文本标记语言</p>
    </section>
    <section>
      <h1>安比</h1>
      <p>热带风暴安比(英文:tropical storm Ampil,国际编号:1810),为 2018 年太平洋台风季中第 10 个命名的风暴,“安比”一名由柬埔寨提供,是首次使用,名字意义为一种原产于东部非洲的酸豆树。</p>
    </section>
  </body>
</html>

```

运行结果如图 2-14 所示。



图 2-14 section 元素实例

2. <article>元素

<article>标签定义独立的内容，其定义的内容本身必须是有意义的且必须独立于文档的其余部分。

3. <header>元素

<header>标签定义文档或者文档的一部分区域的页眉。<header>元素应该作为介绍内容或者导航链接栏的容器。在一个文档中可以定义多个 <header>元素。

【例 2-3】 <article>元素和<header>元素实例。

```
<!doctype html>
<html>
<meta charset="utf-8">
<body>
<article>
<header>
<h1>Internet Explorer 11</h1>
</header>
<p> Windows Internet Explorer 11(缩写为 IE 11 )</p>
</article>
</body>
</html>
```

运行结果如图 2-15 所示。



图 2-15 <article>、<header>元素实例运行结果

4. <nav>元素

<nav>标签定义导航链接的部分。<nav>元素用于定义页面的导航链接部分区域，但是，不是所有的链接都需要包含在<nav>元素中。

【例 2-4】 <nav>元素实例。

```
<!doctype html>
<html>
<meta charset="utf-8">
<body>
  <nav>
    <a href="http://xinxi.tjmvi.cn/info/1005/1019.htm">计算机网络技术</a> |
    <a href="http://xinxi.tjmvi.cn/info/1005/1018.htm">软件技术</a> |
    <a href="http://xinxi.tjmvi.cn/info/1005/1849.htm">电子商务技术</a> |
    <a href="http://xinxi.tjmvi.cn/info/1005/1393.htm">船舶通信与导航</a> |
  </nav>
</body>
</html>
```

页面导航链接如图 2-16 所示，页面导航链接运行结果如图 2-17 所示。



图 2-16 页面导航链接



图 2-17 页面导航链接运行结果

2.2.6 已被移除的元素

以下 HTML 4.01 中的元素在 HTML5 中已经被移除。

- (1) <applet>。
- (2) <basefont>。

- (3)<big>。
- (4)<center>。
- (5)<dir>。
- (6)。
- (7)<frame>。
- (8)<frameset>。
- (9)<noframes>。
- (10)<strike>。
- (11)<tt>。

2.3 案例

【案例描述】

学习完模块 2 的内容,小张同学希望做一个网页,在网页中有一个链接标签,单击该超链接可以跳转到百度网站首页。

【源代码展示】

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>我的超链接</title>
</head>
<body>
  <a href="www.baidu.com">百度首页</a>
</body>
</html>
```

【运行结果】

运行结果如图 2-18 和图 2-19 所示。



图 2-18 模块 2 案例运行结果 1



图 2-19 模块 2 案例运行结果 2

【程序分析】

小张同学根据本模块所学的知识完成了页面的设计,单击超链接,成功跳转到百度网站首页。在设计中,小张运用了<a>标签定义了超链接,并成功跳转到百度首页。

2.4 思考与练习

一、填空题

- HTML 文档主要由 HTML 标签、_____、_____、_____、动画、超链接等组成。
- 一个完整的 HTML 文档至少由文档声明、_____、_____ 和 _____ 四部分组成。

二、选择题

- 在 HTML 中,() 标记不可出现在 `<body>` 和 `</body>` 标记符之间。

| | |
|-------------------------------|--------------------------------|
| A. <code><hr></code> | B. <code> </code> |
| C. <code><title></code> | D. <code><! ----></code> |
- 以下字号最大的标题是()。

| | |
|----------------------------|----------------------------|
| A. <code><h7></code> | B. <code><h6></code> |
| C. <code><h2></code> | D. <code><h1></code> |
- 下列哪种文件不是视频文件? ()

| | |
|------------|------------|
| A. .jpg 文件 | B. .avi 文件 |
| C. .mov 文件 | D. .mpg 文件 |
- 设置网页名称的标签是()。

| | |
|---|---|
| A. <code><title></code> 和 <code></title></code> | B. <code><head></code> 和 <code></head></code> |
| C. <code><titles></code> 和 <code></titles></code> | D. <code><name></code> 和 <code></name></code> |
- HTML 文档结构由() 组成。

| |
|---|
| A. <code><html>.....</html></code> 、 <code><title>.....</title></code> 、 <code><body>.....</body></code> |
| B. <code><body></code> 、 <code></body></code> <code><h1></h1></code> <code><html></code> 、 <code></html></code> |
| C. <code><html>.....</html></code> 、 <code><head>.....</head></code> 、 <code><body>.....</body></code> |
| D. <code><head>.....</head></code> <code><meta>.....</meta></code> <code><title>.....</title></code> |
- 下列哪个标签不是文档标签? ()

| | |
|-------------------------------|------------------------------|
| A. <code><html></code> | B. <code><body></code> |
| C. <code><title></code> | D. <code><td></code> |
- HTML 不包括() 标签。

| | |
|---------|-------|
| A. 文档 | B. 图形 |
| C. Java | D. 框架 |

8. HTML 文件的标题显示在浏览器的()。
- A. 菜单栏
 - B. 标题栏
 - C. 地址栏
 - D. 状态栏
9. 设置网页标题时,在 HTML 源代码中需要使用()标记。
- A. `<head></head>`
 - B. `<title></title>`
 - C. `<body></body>`
 - D. `<table></table>`

3

模块 3

创建普通文档

在 HTML5 出现之前,各个浏览器厂商对 HTML、JavaScript 的支持很不统一,这样就造成了同一页面在不同浏览器中显示不同的情况。HTML5 的出现对于 Web 来说意义重大,HTML5 的目标是详细分析各浏览器所具有的功能,以此为基础制定一个通用的规范,并要求各浏览器能支持该通用的规范。

3.1 HTML5 文档的基本结构

一个 HTML 文件是由一系列元素和标签组成的。元素是 HTML 文件的重要组成部分,如 title(HTML 文件标题)、img(图像)、table(表格)等。元素名不区分大小写。HTML 用标签来规定元素的属性和其在文件中的位置。

HTML 的标签一般成对出现,但也会单独出现。首标签的格式为<元素名称>,尾标签的格式为</元素名称>。其完整的语法如下。

```
<元素名称>被控制的元素</元素名称>
```

成对标签仅对包含于其中的内容起作用。例如,<title>和</title>标签用于界定标题内容,也就是说,<title>和</title>之间的内容是此 HTML 文件的标题。

单独标签格式为<元素名称>,其作用是在相应的位置插入元素。例如,
标签便是在该标签所在的位置插入一个换行符。

在每个 HTML 标签中,还可以设置一些属性,控制 HTML 的元素,这些属性位于所建立元素的首标签内,其基本语法如下。

```
<元素名称 属性 1="值 1" 属性 2="值 2".....>  
内容  
</元素名称>
```

需要说明的是,HTML 是对大小写不敏感的语言,也就是同一字母的大写和小写字符被认定为同一个字符。例如,<HTML>和<html>的意义是一样的,因此在每一个<html>标签内,大写、小写、混写均可;但如果嵌入其他语言,如 JavaScript,则需要按照其他语言的规范来判断是否区分大小写字符。统一大小写规范是一个优秀的程序编写人员必备的素质,建议大家按照统一的规范编写,本书中范例的标签均采用小写。

3.2 HTML5 基本标记

3.2.1 文档类型标记<!doctype>

<!doctype>是文档类型标记,声明必须位于 HTML5 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>声明不属于 HTML 标记,它是一种指令,告知浏览器编写页面所用的标记版本,写法如下。

```
<!doctype html>
```

3.2.2 HTML 文档标记<html>

在任何一个 HTML 文件中,最先出现的 HTML 标签就是<html>,它用于表示该文件是以超文本标记语言编写的。<html>是成对出现的,首标签<html>和尾标签</html>分别位于文件的最前面和最后面,文件中所有内容和 HTML 标签都要包含其中。写法如下。

```
<html>  
文件内容  
</html>
```

HTML 标签不带任何属性。

3.2.3 头部标记<head>

HTML 的头元素一般需要包括标题、基底信息、元信息等。HTML 的头元素是以<head>为开始标记,以</head>为结束标记的。一般情况下,CSS 和 JavaScript 都定义

在头元素中,而定义 HTML 头部的内容往往不会在网页上直接显示,它用于设定当前文档的相关信息。头部标记见表 3-1。

表 3-1 头部标记

| 标 记 | 描 述 |
|-------------|---------------------------------|
| <base> | 当前文档的 URL 全称(基底网址) |
| <base font> | 设定基准的文字字体、字号和颜色 |
| <title> | 设定显示在浏览器左上方的标题内容 |
| <is index> | 表明该文档是一个可用于检索的网关脚本,有服务器 |
| <meta> | 设定有关文档本身的元信息,如用于查询的关键字、该文档的有效期等 |
| <style> | 设定 CSS 层叠样式表的内容 |
| <link> | 设定外部文件的链接 |
| <script> | 设定页面中程序脚本的内容 |

<head>与</head>之间的内容不会在浏览器的文档窗口显示,但是其间的元素有特殊重要的意义,下面就来分别介绍这些标记的作用。

3.2.4 标题标记<title>

每个 HTML 文件都需要有一个文件名称。在浏览器中,文件名称作为窗口名称显示在该窗口的上方。这对浏览器的收藏功能很有用。如果浏览者认为某个网页对自己很有用,今后想经常阅读,就可以选择 IE 浏览器“收藏”菜单中的“添加到收藏夹”命令将它保存起来,供以后调用。网页的名称应写在<title>和</title>之间,并且<title>标签应包含在<head>与</head>标签之中。

HTML 文件标签是可以嵌套的,即在一对标签中可以嵌入另一对子标签,用来规定母标签范围的属性或其中某一部分内容,嵌套在<head>标签中的主要是<title>标签。

【例 3-1】 标题标记的应用。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>标题样例</title>
</head>
<body>

</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-1 所示的结果。



图 3-1 标题样例的显示结果

3.2.5 主体标记<body>

网页所要显示的内容都被放在网页的主体标记内,是 HTML 文件的重点。主体标记以 <body> 开始,以 </body> 结束,语法如下。

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

3.2.6 换行标记

段落与段落之间是需要换行的,这时就需要使用换行标记
。
 元素是一个空的 HTML 元素。由于关闭标签没有任何意义,因此它没有结束标签。

【例 3-2】 换行标记的应用。

```
<!doctype html>  
<html>  
<head>  
<meta charset="utf-8">  
<title>换行样例</title>  
</head>  
<body>
```

HTML(Hypertext Markup Language,超文本标记语言)诞生于 20 世纪 90 年代初,用于指定构建网页的元素,这些元素中的大多数都用于描述网页内容,如标题、段落、列表、指向其他网页的链接等。
 HTML5 是 HTML 的最新版本,它的大部分内容都可以兼容新旧浏览器,并新增了大量新的功能。HTML5 还引入了原生的音频和视频播放功能。


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

在浏览器中运行上面的代码,得到图 3-2 所示的结果。

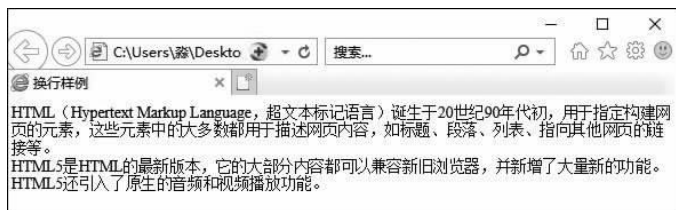


图 3-2 换行标记的效果

3.2.7 水平线标记<hr>

水平线用于段落与段落之间的分隔,使文档结构更清晰。水平线自身有很多属性,如宽度、高度、颜色、排列对齐等。在 HTML 文档中经常会用到水平线,合理地运用水平线可以得到非常好的效果。

1. 添加水平线

语法如下。

```
<hr>
```

在网页中输入一个<hr>标记,就添加了一条默认样式的水平线。

【例 3-3】 添加水平线。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>添加一条水平线</title>
</head>
<body>
<hr>
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-3 所示的效果。



图 3-3 添加水平线的效果

2. 设置水平线的宽度和高度属性

在默认情况下,在网页中插入的水平线的宽度是 100%,高度是 1。在实际网页中,通过设置 width、size 的属性参数可以对水平线的宽度和高度进行设置。

语法如下。

```
<hr width=水平线宽度 size=水平线高度>
```

在该语法中,水平线的宽度值可以是确定的像素值,也可以是相对于浏览器窗口宽度的百分比。而水平线的高度值只能是像素值;如果在创建水平线时只设置一个参数,那么另外一个没有设置的参数取默认值。

【例 3-4】 设置水平线的宽度和高度。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置水平线的宽度和高度</title>
</head>
<body>
<hr width="50 %">
<hr width=150 size=10>
</body>
</html>
```

在浏览器中运行代码,可以看到两条宽度和高度不同的水平线效果,如图 3-4 所示。



图 3-4 设置水平线的宽度和高度效果

3. 设置水平线的颜色

为了使水平线更美观,还可以设置水平线的颜色。

语法如下。

```
<hr color="颜色代码">
```

需要说明的是,颜色代码在 HTML 里有两种表现形式:一种是 6 位十六进制数,如 #FFB6C1、#FFFFFF 等;另一种是英文代码,如 pink、red、black 等。

【例 3-5】 设置水平线的颜色。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置水平线的颜色</title>
</head>
<body>
<hr width="50%" color="#f0ff00">
<hr width=150 size=10 color="red">
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-5 所示的效果。

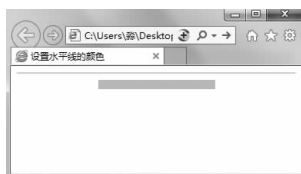


图 3-5 设置水平线颜色的效果

4. 设置水平线的对齐方式

水平线在默认情况下是居中对齐的,如果希望水平线左对齐或右对齐,就需要设置 align 参数。

语法如下。

```
<hr align=对齐方式>
```

水平线的对齐方式有三种,包括 left、center 和 right。其中,center 的效果与默认效果相同。

【例 3-6】 设置水平线的对齐方式。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>设置水平线的对齐方式</title>
</head>
<body>
<hr width="50%" color="#f0ff00" align="left">
<hr width=150 size=10 color="red" align="right">
```

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

在浏览器中运行上面的代码,得到图 3-6 所示的效果。

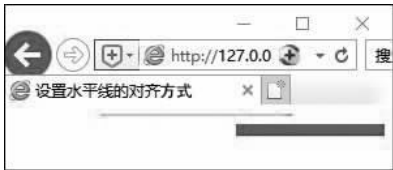


图 3-6 设置水平线对齐方式的效果

3.2.8 注释标记<!--...-->

在网页中,除了能够在页面显示出来的元素外,还包含一种不显示在页面中的元素,也就是注释文字。适当的注释可以帮助用户更好地了解网页中各个模块的功能,也有助于以后对代码的修改。给代码加注释,是一种很好的编程习惯。

语法如下。

```
<!-- 注释的文字 -->
```

例如:

```
<----->
<!-- 文件说明:这是第一个文件 -->
<----->
```

上面的注释语句只出现在源代码中,而不会在浏览器页面中显示。

3.3 在网页中添加文本

3.3.1 文档作者或拥有者联系信息标记<address>

<address>标签定义文档作者或拥有者的联系信息。如果<address>元素位于<article>元素内部,则它表示该文档作者或拥有者的联系信息。通常的做法是将<address>元素添加到网页的头部或底部。

【例 3-7】 添加文档作者信息。

```
<!doctype html>
<html>
```



```

<head>
<meta charset="utf-8">
<title>添加文档作者信息</title>
</head>
<body>
<address>
    Written by html 教材编写组<br>
    <a href="mailto:us@example.org">Email us</a><br>
    Address: Box 564, Disneyland<br>
    Phone: +12 34 56 78
</address>
</body>
</html>

```

在浏览器中运行上面的代码,得到图 3-7 所示的效果。



图 3-7 添加文档作者或拥有者联系信息标记的效果

3.3.2 加粗文本标记

在编辑网页时,经常需要对一些特殊文字进行加粗显示,该效果可以通过标签来实现。

语法如下。

```
<b>需要加粗的文字</b>
```

【例 3-8】 加粗文本标记的用法。

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>加粗标记</title>
</head>
<body>

```

```
<b>加粗标记</b>
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-8 所示的效果。

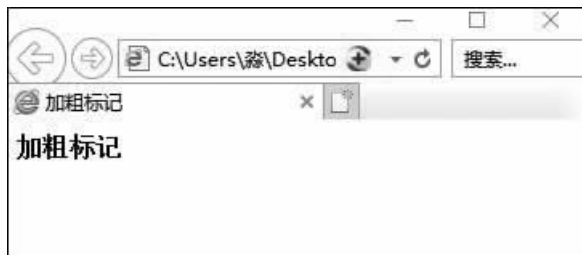


图 3-8 加粗标记的效果

HTML5 通过标记来控制文字的显示效果,读者将改为<i></i>就能得到斜体的效果,改为<u></u>就能得到下划线的效果。

这种标签还可以嵌套使用。例如,如果需要得到加粗且斜体的效果,就可以写成如下形式。

```
<b><i>加粗斜体效果字</i></b>
```

【例 3-9】 文本加粗和斜体标记的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>加粗和斜体标记</title>
</head>
<body>
<b><i>加粗和斜体标记</i></b>
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-9 所示的效果。



图 3-9 加粗和斜体标记的效果

3.3.3 定义计算机代码文本标记<code>

<code> 标签用于表示计算机源代码或其他机器可以阅读的文本内容。

软件代码的编写者已经习惯了编写源代码时文本表示的特殊样式。<code> 标签就是为他们设计的。包含在该标签内的文本将以等宽、类似电传打字机样式的字体(courier)显示出来,对于大多数程序员和 W3School 的用户来说,这应该是十分熟悉的。

只能在表示计算机程序源代码或其他机器可以阅读的文本内容上使用 <code> 标签。虽然 <code> 标签通常只是把文本变成等宽字体,但它暗示着这段文本是源程序代码。将来的浏览器有可能会加入其他显示效果。例如,程序员的浏览器可能会寻找 <code> 片段,并执行某些额外的文本格式化处理,如循环和条件判断语句的特殊缩进等。

3.4 文本排版标记

3.4.1 换行标记

段落与段落之间是隔行换行的,此时文字的行间距过大,可以使用换行标记来完成文字的换行。

【例 3-10】 文本换行标记的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>换行样例</title>
</head>
<body>
```

HTML(Hypertext Markup Language,超文本标记语言)诞生于 20 世纪 90 年代初,用于指定构建网页的元素,这些元素中的大多数都用于描述网页内容,如标题、段落、列表、指向其他网页的链接等。
 HTML5 是 HTML 的最新版本,它的大部分内容都可以兼容新旧浏览器,并新增了大量新的功能。HTML5 还引入了原生的音频和视频播放功能。


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

上面的代码在浏览器中的显示效果如图 3-10 所示。

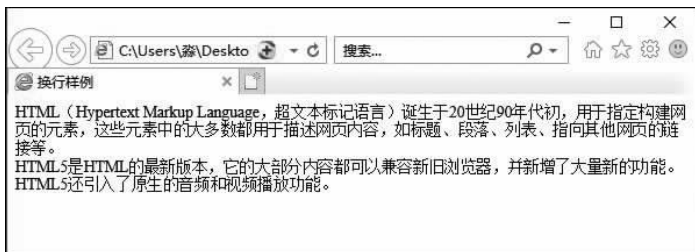


图 3-10 段落文字换行的效果

3.4.2 段落标记<p>

在 HTML 中,段落通过<p>标记来表示。
语法如下。

```
<p>段落文字</p>
```

【例 3-11】 段落标记的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>段落标记</title>
</head>
<body>
<p>信息是用文字、数字、符号、语言、图像等介质来表示事件、事物、现象等内容、数量或特征,从而向人(或系统)提供关于现实世界新的事实和知识,作为生产、建设、经营、管理、分析和决策的依据。</p>
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-11 所示的效果。



图 3-11 段落标记的效果

3.4.3 标题标记<h1>至<h6>

<h1>至<h6>标签可定义标题。<h1>定义最大的标题,<h6>定义最小的标题。

由于 h 元素拥有确切的语义,应慎重地选择恰当的标签层级来构建文档的结构。因此,不应利用标题标签来改变同一行中的字体大小。相反,应当使用层叠样式表定义来实现漂亮的显示效果。

<h1>元素用来描述网页中最上层的标题。

由于一些浏览器会默认地把<h1>元素显示为很大的字体,因此会有一些 Web 开发者使用<h2>元素代替<h1>元素来显示最上层的标题。这样做不会对读者产生影响,但会使那些试图“理解网页结构”的搜索引擎和其他软件感到“迷惑”。

应该确保把<h1>用于最顶层的标题,<h2>和<h3>用于较低层级的标题。

【例 3-12】 标题标记的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>h1-h6</title>
</head>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
</body>
</html>
```

在浏览器中运行上面的代码,得到图 3-12 所示的效果。

如果不喜欢默认的标题字体尺寸,可以使用样式或样式表来改变。

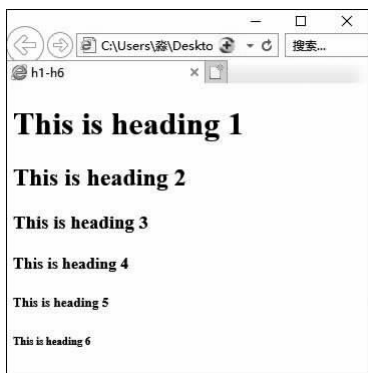


图 3-12 h1 至 h6 的效果

3.5 列表

HTML 网页中的文字列表如同文字编辑软件 Word 中的项目符号和自动编号。

3.5.1 建立有序列表

有序表类似 Word 中的自动编号功能,有序列表使用的标记是,每一个列表项前面使用。每个项目都有前后顺序之分,一般用数字表示,其结构如下。

```
<ol>
<li>有序列表项 1</li>
<li>有序列表项 2</li>
<li>有序列表项 3</li>
<li>有序列表项 4</li>
</ol>
```

【例 3-13】 有序列表的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>有序列表</title>
</head>
<body>
<ol>
<li>HTML5 基本结构</li>
<li> HTML 文档标记</li>
<li>头部标记 HEAD</li>
<li> 标题标记</li>
<li>主体标记</li>
<li>无序表</li>
</ol>
</body>
</html>
```

上述代码在浏览器中的显示结果如图 3-13 所示。

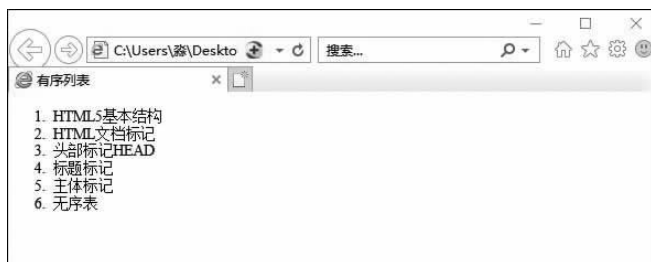


图 3-13 有序列表的显示结果

在默认情况下,得到的列表序号是 1、2、3……,HTML 允许通过设置 type 属性来自定义序号的样式。例如,把上面的代码的标签改为<ol type="A">,在 IE 11 中预览可以得到图 3-14 所示的效果。

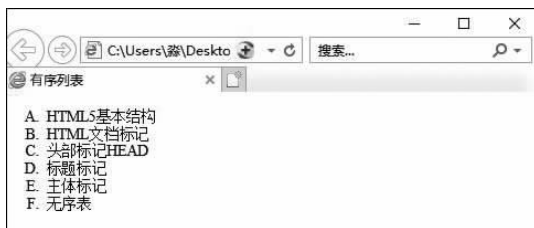


图 3-14 有序列表的自定义序号结果

其中,type 的属性还可设置为“I”“a”“i”等,可以根据需要进行设置。

3.5.2 建立无序列表

无序列表相当于 Word 中的项目符号,无序列表的项目排列没有顺序,只以符号作为分项标识。无序列表使用一对标记,其中每一个列表使用,其结构如下。

```
<ul>
<li>无序列表项</li>
<li>无序列表项</li>
<li>无序列表项</li>
<li>无序列表项</li>
</ul>
```

在无序列表结构中,使用标记表示无序列表的开始和结束,则表示一个列表项的开始。在一个无序列表中可以包含多个列表项,并且可以省略结束标记。

【例 3-14】 无序列表的用法。

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>无序列表</title>
</head>
<body>
<ul>
<li>HTML5 基本结构</li>
<li> HTML 文档标记</li>
<li>头部标记 HEAD</li>
<li>标题标记</li>
<li>主体标记</li>
<li>无序表</li>
</ul>
</body>
</html>

```

上述代码在浏览器中的显示结果如图 3-15 所示。



图 3-15 无序列表的显示结果

在无序列表中,可以通过修改 type 属性值来修改无序列表标记的样式。

【例 3-15】 无序列表修改标记的用法。

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>无序列表</title>
</head>
<body>

```

Disc 项目符号列表:


```
<ul type="disc">  
<li>苹果</li>  
<li>香蕉</li>  
<li>柠檬</li>  
<li>橘子</li>  
</ul>
```

Circle 项目符号列表:

```
<ul type="circle">  
<li>苹果</li>  
<li>香蕉</li>  
<li>柠檬</li>  
<li>橘子</li>  
</ul>
```

Square 项目符号列表:

```
<ul type="square">  
<li>苹果</li>  
<li>香蕉</li>  
<li>柠檬</li>  
<li>橘子</li>  
</ul>
```

```
</body>
```

```
</html>
```

上述代码在浏览器中的显示结果如图 3-16 所示。

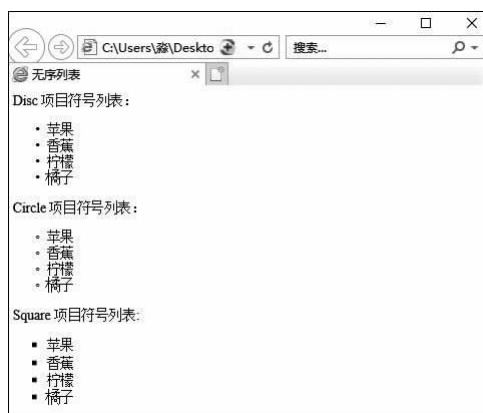


图 3-16 无序列表项目符号的不同效果

HTML 不仅能够实现有序列表和无序列表,在和之间使用

或``还可以制作网页中的嵌套列表。

【例 3-16】 列表嵌套的用法。

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>嵌套列表</title>
</head>
<body>
<ul>
<li>咖啡</li>
<li><ol>茶
    <li>红茶</li>
    <li><ul>绿茶
        <li>龙井</li>
        <li>碧螺春</li>
    </ul>
</li>
</ol>
</li>
<li><ul>牛奶
    <li>蒙牛</li>
    <li>伊利</li>
</ul>
</li>
</ul>
</body>
</html>
```

在 IE 浏览器中预览上述代码,结果如图 3-17 所示。



图 3-17 列表嵌套的结果

3.6 案例

【案例描述】

为了参加信息化教学评比,讲授思想政治课的孙老师找到小张同学,想让小张同学帮忙设计一个 HTML 页面,展示马克思主义哲学课程中绪论的教学内容。

【源代码展示】

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>马克思主义哲学绪论</title>
</head>
<body>
<h1>绪论 </h1>
<p>马克思主义是关于无产阶级和人类解放的科学</p>
<h2>【教学目的和要求】</h2>
<p>从总体上理解和把握什么是马克思主义,为什么要学习马克思主义,怎样学习马克思主义。
</p>
<h2>【教学重点和难点】</h2>
<ol>
<li>马克思主义是关于无产阶级和人类解放的科学</li>
<li>马克思主义在 19 世纪 40 年代产生的必然性</li>
<li>马克思主义是对人类文明成果的继承与创新</li>
<li>马克思主义发展的阶段和成果</li>
<li>马克思主义的根本理论特征</li>
<li>马克思主义是我们行动的指南</li>
<li>学习、运用和发展马克思主义</li>
</ol>
<h2>【计划课时】3 课时</h2>
<h2>【授课提纲】</h2>
<h3>一、什么是马克思主义</h3>
<p>阶级属性</p>
<p>研究对象和主要内容</p>
<h3>二、马克思主义的产生和发展</h3>
```

```

<ol type="1">
<li>马克思主义是时代的产物</li>
<li>马克思恩格斯的革命实践和对人类文明成果的继承与创新</li>
<li>马克思主义在实践中不断发展
  <ol type="a">
    <li>马克思主义哲学是时代精神的精华</li>
    <li>马克思主义具有与时俱进的品格</li>
    <li>马克思主义的发展</li>
  </ol>
</ol>
</ol>
<h3>三、马克思主义科学性与革命性的统一</h3>
<ol style="1">
<li><p>马克思主义最根本的理论特征</p>
  <ul>
<li>辩证唯物主义与历史唯物主义是无产阶级的世界观与方法论</li>
<li>辩证唯物主义与历史唯物主义也是马克思主义理论的思想基础</li>
</ul>
</li>
<li>马克思主义最鲜明的政治立场
  <ul>
<p>马克思主义政党的一切理论和奋斗都应致力于实现最广大人民的根本利益,这是马克思主义最鲜明的政治立场</p>
<li>这是由马克思主义理论的本性决定的</li>
<li>这是由无产阶级的历史使命决定的</li>
<li>是否始终站在最广大人民的立场上,是唯物史观和唯心史观的分水岭,也是判断马克思主义政党的试金石</li>
</ul></li>
<li>马克思主义最重要的理论品质
  <p>坚持一切从实际出发,理论联系实际,实事求是,在实践中检验真理和发展真理,是马克思主义最重要的理论品质</p>
  <ul>
<li>实事求是、与时俱进是马克思主义理论的本质要求</li>
<li>富于创造性,是马克思主义与时俱进品质的生动体现</li>
<li>与时俱进的理论品质,是150多年来马克思主义始终保持蓬勃生命力的关键</li>
</ul>
</li>
<li>马克思主义最崇高的社会理想</li>

```

```

</ol>
<h2>【本章小结】</h2>
<p>1. 每一个时代的理论思维都是一种历史的产物。马克思主义正是适应了时代的需要而产生的无产阶级的科学理论。</p>
<p>2. 马克思主义在总结人类思想成果的基础上形成,也在不断汲取思想精华中发展。这是一个理论与实践统一的过程,也是一个在开放中不断发展的过程。</p>
<p>3. 马克思主义是革命性和科学性的有机统一,这是马克思主义的根本特征,也是其内在要求。</p>
<p>4. 马克思主义的学说不是僵硬的教条,而是行动的指南。这是学习掌握马克思主义基本原理的原则,也是基本要求。</p>
</body>
</html>

```

【运行结果】

运行结果如图 3-18 所示。



图 3-18 模块3 案例运行结果

【程序分析】

小张同学根据孙老师给的教学内容,结合本模块所学的知识完成了页面的设计。在设计中,小张同学运用了<h1></h1>标题样式为绪论、教学目的和要求等标题添加了标签,使内容的层次分明,还充分利用有序列表和无序列表对授课提纲中的内容进行了设计,使授课内容层次非常清晰。

3.7 思考与练习

一、选择题

- 创建最小的标题的文本标签是()。
 - <pre></pre>
 - <h1></h1>
 - <h6></h6>
 -
- 设置水平线高度的 HTML 代码是()。
 - <hr>
 - <hrsize=? >
 - <hrwidth=? >
 - <hrnoshade>
- 在 HTML 中,段落标签是()。
 - <HTML>...</HTML>
 - <HEAD>...</HEAD>
 - <BODY>...</BODY>
 - <P>...</P>
- 在 HTML 中,表示换行的标签是()。
 - <u>
 -
 -

 - <h1>
- 在 HTML 中,加粗字的文本标签是()。
 - <pre></pre>
 - <h1></h1>
 - <h6></h6>
 -

二、填空题

- HTML5 文件中用_____标签表示文件头部,其中_____标签用来定义页面的标题。
- 是_____标签,如果希望自定义序号的样式,需要设置_____的属性。
- HTML5 中用_____来表示注释内容。

三、简答题

- HTML5 文件头部包括哪些内容?
- HTML5 包括哪些基本标记?