

**SAMS**  
**Teach**  
**Yourself**

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- 秉承 Read Less, Do More (精读多练) 的教学理念
- 以示例引导读者完成最常见的任务

# HTML与CSS

## 入门经典

(第7版) [美] Dick Oliver 著  
Michael Morrison 译  
陈秋萍 译

每章内容针对初学者精心设计，**1** 小时轻松  
阅读学习，**24** 小时彻底掌握关键知识

每章 **案例与练习题** 助你轻松完成常  
见任务，通过 **实践** 提高应用技能，巩固  
所学知识

**in 24**  
**Hours**

 **人民邮电出版社**  
POSTS & TELECOM PRESS



# HTML与CSS

## 入门经典

(第7版)

[美] Dick Oliver 著  
Michael Morrison 著  
陈秋萍 译

人民邮电出版社  
北京





## 图书在版编目 (CIP) 数据

HTML 与 CSS 入门经典 第 7 版/ (美) 奥利弗 (Oliver, D.) (美) 莫里森 (Morrison, M.) 著; 陈秋萍译. —北京: 人民邮电出版社, 2007.3

ISBN 978-7-115-15625-9

I. H... II. ①奥...②莫...③陈... III. ①超文本标记语言, HTML—主页制作—程序设计②主页制作—软件工具, CSS IV. TP312 TP393.092

中国版本图书馆 CIP 数据核字 (2006) 第 152514 号

## 版权 声 明

Dick Oliver, Michael Morrison: Sams Teach Yourself HTML and CSS in 24 Hours, Seventh Edition  
ISBN: 0672328410

Copyright©2006 by Sams Publishing.

Authorized translation from the English language edition published by Sams.

All rights reserved.

本书中文简体字版由美国 Sams 出版公司授权人民邮电出版社出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

## HTML 与 CSS 入门经典 (第 7 版)

- ◆ 著 [美] Dick Oliver Michael Morrison
- 译 陈秋萍
- 责任编辑 李 际
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
人民邮电出版社河北印刷厂印刷  
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16  
印张: 26.25  
字数: 657 千字 2007 年 3 月第 1 版  
印数: 1—5 000 册 2007 年 3 月河北第 1 次印刷

著作权合同登记号 图字: 01-2006-3975 号

ISBN 978-7-115-15625-9/TP

定价: 39.00 元

读者服务热线: (010)67132705 印装质量热线: (010)67129223

# 内容提要

本书将使用 HTML 和 CSS 创建并发布网页的过程分解成简单的步骤，详细阐述其中的每一步，让读者能够迅速掌握。全书分 6 个部分，包括 24 章和 4 个附录。第一部分简要地介绍了 HTML 和 XHTML，阐述了网页的创建过程和发布步骤，讨论了如何连接到其他网页；第二部分介绍了文本的对齐和格式化，讨论了文本链接、图像链接和图像映射，阐述了如何处理用于网页中的图像，还介绍了如何设置网页的背景和颜色；第三部分讨论了如何使用表格、CSS 和框架设计网页布局；第四部分探讨了如何使用脚本、表单和多媒体设计动态网页，还介绍了 HTML 和 CSS 技术的两种实际应用——创建令人印象深刻的 eBay 拍卖网页和引人入胜的博客；第五部分阐述如何创建网站，包括网站的组织和管理以及如何让网站在搜索结果中靠前；第六部分为附录，列出了重要的 HTML 和 CSS 资源网站，提供了完整的 XHTML 1.1 和 CSS 2 快速参考，讨论了如何将 HTML 文档转换为 XHTML。

本书通过大量的示例，以循序渐进的方式引导读者通过实践迅速、全面地掌握 HTML 和 CSS 的基础知识，适合对制作网页感兴趣的初学者。





## 作者简介

Michael Morrison 是一位作家、开发人员、玩具发明者，还是许多计算机技术书籍和基于 web 的交互课程的作者。除作为作家和自由职业者外，Michael 还是 Stalefish 实验室创意领导人，这是他同妻子 Masheed 创建的一家娱乐公司。Stalefish 实验室的第一个基于 web 的项目是一个在线音乐游戏，名为 Guess That Groove (<http://www.guessthatgroove.com/>)。不使用计算机时，Michael 玩曲棍球、滑板或与妻子去看电影。他还喜欢一边构想 Web 创意一边玩 koi pond。读者可以通过 <http://www.michaelmorrison.com/> 拜访 Michael 并同他讨论本书。



# 前言

## 现在就将 HTML 网页放到网上去

在 2005 年，大约将有 9 亿人能够访问 Internet，其中仅美国就有 1.88 亿。日本有 5800 万用户，德国有 4500 万用户，英国有 3500 万用户，俄罗斯有 1800 万，加拿大有 1700 万，从中可以看出万维网（World Wide Web）中“world”的涵义。在所有使用 Internet 的这些用户当中，有许多人为 Web 创建新内容。虽然很难统计网页的精确数量，但在 2005 年年底搜索引擎 Google 报道了它索引的网页多达 40 亿。

每天都有数万个新网页发布到 Internet 上可公开访问的地方。至少还有同样多的网页将放在私有的内部网中，这些网页只能由连入局域网的企业内部人员访问。这些网页和已有的 40 亿网页一样，都将使用超文本标记语言（Hypertext Markup Language, HTML）。

如果继续阅读下去，在学完本书内容后，你的网页也将出现在 Internet 上，同时你也将掌握世界上最有价值的技能之一：HTML。

你真的能够学会创建顶级网页——不使用专门软件，时间比申请并等待昂贵的 HTML 课程更少？这本薄薄的、易于阅读的图书是否真的能够帮助你自学网页发布技能？

答案是肯定的。实际上，在本书开始的两章内，毫无 HTML 经验的读者就能够创建可发布到万维网的网页。

怎么能够这么快学会 Web 语言呢？通过示例。本书将 HTML 的学习分解为简单的步骤，教你如何完成每一步，任何人都能够很快掌握。创建每个 HTML 示例前，列出其显示结果，然后简要地介绍如何实现，并让读者创建相同的网页。十分钟后，就将进入下一步。

学完本书，你将对自己创建的网页感到惊讶。

## 超越 HTML

本书不仅介绍 HTML，因为现在创建网页需要掌握的不仅是 HTML。我的目标是在 24



章简短的课程内教会你创建优秀的网站所需要的一切技能。我已经收到过很多读者的电子邮件，他们告诉本书以前的版本比其他任何书都更好地实现了这一目标。

去书店看看，你将发现本书是市场上少见的以初学者也能理解的平实语言介绍了下列所有重要技能和技术的图书。

- XHTML (Extended Hypertext Markup Language, 扩展超文本标记语言) 和 XML (eXtensible Markup Language, 可扩展标记语言) 是创建网页的新标准。本书 (及配套网站) 的每个示例都完全同 XHTML 和 XML 兼容, 当 XHTML 和 XML 取代老式 HTML 时, 读者不必重新学习任何东西。

### By the Way

**注意:** 如果已有的网页需要更新以便与新标准兼容, 可参考附录 C, 它详尽地介绍了如何将 HTML 页面转换为 XHTML 页面。

- 同时, 本书所有的示例都使用每个主流网页浏览器的最新版本测试过, 包括 Microsoft Internet Explorer、Mozilla Firefox、Opera 和 Safari。一开始你就将学习如何与过去保持兼容, 并为未来做好准备。
- 本书详尽地介绍了级联样式表 (Cascading Style Sheets, CSS), 它让你能够详细地控制网页的每一方面: 文本和图像的布局、字体、颜色和格式化。创建网页时, CSS 可以实现传统 HTML 页面本身不能实现的功能。例如, 除正常浏览外, CSS 还让你能够定制用于打印的网页。
- 第 7 至 10 章介绍使用业界标准软件设计和创建网页图像 (包括动画), 这些软件可免费从网上下载并试用。创建图像是创建美观网站的最重要部分, 这是大多数 HTML 图书没有涉及的。
- 除 HTML 和 CSS 外, 第 17 至 19 章还将介绍如何使用 JavaScript、动态 HTML (DHTML) 和嵌入多媒体。你的网页将是交互性的、迷人的, 而不是静态的、没有响应的。
- 你是否希望能够使用 HTML 在 eBay 上创建精致的拍卖清单? 第 20 章将介绍如何使用 HTML 和 CSS 知识来创建优美的 eBay 拍卖清单, 使拍卖交易获得最大价值。
- 你可能听说过博客在 Internet 上掀起的狂热。第 21 章将介绍如何创建博客并将它链接到个人网站。你马上就能成为真正的博客!
- 仅有技术是不够的, 因此我还将在如何设置网站以满足现实需要这方面给予指导。设计有效的页面布局, 使用 FTP 软件将页面上传到 Internet, 组织和管理多个页面, 以及使页面显示在所有主要 Internet 搜索网站查询列表的前面, 对这些重要细节都进行了足够深入的介绍, 让你能够避免发生常见的障碍。
- 你可能知道可视化 Web 网站编辑器, 如 Microsoft FrontPage 和 Macromedia DreamWeaver, 这些编辑器让根本不懂 HTML 的人也能设计网站。但它们实际上要求你更深入地了解 HTML, 这样才能够创建出易于阅读和维护的网页。在本书中, 我给出了一些提示, 指出什么时候使用“所见即所得”编辑器更好, 什么时候自己编写 HTML 代码更好。

正是众多的基本知识让本书的前 6 版畅销不衰。在本版中, 我考虑了众多读者通过电子邮件提供的反馈意见, 使每个课程更容易, 可以更快速地学习。我修订了书中的示例, 让读

者能够在线体验，然后根据需要进行修改——总共有将近 300 个示例网页。书中的快速参考表和更新的附录，供你成为熟练的 Web 设计人员后参考。

## 如何使用本书

有多种阅读本书的方法，最好的方法取决于读者的具体情况。这里推荐了 5 种方法，请根据需要选择一种。

1. “我现在就需要将一些文本放到 Internet 上，然后再考虑把它弄得好看点”。

- 阅读第 1 章。
- 将第一个网页发布到 Internet!
- 阅读第 2 章。
- 阅读本书的其他内容，学到更多 HTML 知识后再更新网页。
- 阅读第 4 章。

2. “我需要尽快在 Internet 上发布包含文本和图像的基本网页，然后改进它并添加更多的网页”。

- 阅读第 1 章。
- 阅读第 4 章。
- 阅读第 2 章。
- 将第一个网页发布到 Internet!
- 阅读第 7 章。
- 阅读本书的其他内容，学到更多 HTML 知识后再更新网页。
- 阅读第 8 章。

3. “我现在就需要包含订单表单、看起来很专业的商业网站。然后可以继续开发和改进我的网站”。

- 阅读第一部分。
- 阅读第 8 章。
- 阅读第 17 章。
- 阅读第 9 章。
- 阅读第 18 章。
- 阅读本书的其他内容，学到更多 HTML 知识后再更新网页。
- 阅读第 7 章。

4. “我需要在很紧的时间内开发具有创意和吸引力的‘公司标识’网站。然后还要为公司的内部网开发许多页面”。

- 阅读第一部分。
- 将网页发布到 Internet 和内部网。
- 阅读第二部分。
- 阅读本书的其他内容，学到更多 HTML 知识后再更新网页。
- 阅读第三部分。

5. “我需要快速创建极具特色的交互性网站或基于 HTML 的多媒体演示文档!”

- 阅读整本书。
- 将网页发布到 Internet 和/或光盘。
- 复习并使用学到的方法，继续开发和改进网站。

Internet 服务提供商为你的网站设置托管计算机可能需要一两天，这将在第 4 章讨论。如果要立刻将网页发布到网上，现在就阅读第 4 章，这样创建第一个网页后就可以将其发布到



Internet。

注意，不管采取哪种方法，都能从本书的特色中获益，正是这些特色使本书成为学习 HTML 的最快途径。

## 示例

本书的示例都通过两部分来演示。首先列出为创建 HTML 页面需要输入的文本，以演示 HTML 和 CSS 代码；然后列出网页在大多数流行的网页浏览器中的显示结果：Microsoft Internet Explorer、Mozilla Firefox 或 Opera。

本书所有的示例都能在 Mozilla Firefox、Microsoft Internet Explorer、Opera 和 Safari 上正常显示。在本书中，我不断更换用于显示网页的浏览器，旨在说明设计良好的网页在所有浏览器上的显示都应一致。如果页面使用了某个主要浏览器不支持的特性，我将指出该浏览器的局限性。

## 特色

在每章的课程中，“练习”部分指导读者立刻将刚学到的知识应用到自己的网页中。

<b>Did you know?</b>	<b>提示：</b> 可节省宝贵时间的技巧放在“提示”中，让读者能够一眼认出它们。
----------------------	---

<b>By the Way</b>	<b>注意：</b> 不应忽略的重要信息放在“注意”中，以凸现它们。
-------------------	------------------------------------

<b>Coffee Break</b>	<b>休息一下：</b> 让读者稍微休息一下，探索一些有趣的在线示例。
---------------------	-------------------------------------

<b>Watch Out!</b>	<b>警告：</b> 需要特别注意的内容放在“警告”中。
-------------------	------------------------------

## 问与答、测验和练习

每章末尾都有简短的“问与答”小节，其中列出很多人没有勇气问的“傻瓜问题”；简单但全面的“测验”让读者能够检查自己是否理解了本章介绍的所有知识；最后给出了一两个选做的练习，让读者阅读后面的内容前有机会应用刚学到的技能。

## 本书示例下载

在本书网站可下载书中所有示例文件，方法如下：

打开 <http://www.sampublishing.com/bookstore/product.asp?isbn=0672328410&rl=#info2> 页面，(请注意，网址最后“rl=1#info2”中的“1”为字母。)单击 downloads 链接，再单击 Examples.zip 链接，选择“保存”即可。

# 计算机精品学习资料大放送

软考官方指定教材及同步辅导书下载 | 软考历年真题解析与答案

软考视频 | 考试机构 | 考试时间安排

Java 一览无余: Java 视频教程 | Java SE | Java EE

.Net 技术精品资料下载汇总: ASP.NET 篇

.Net 技术精品资料下载汇总: C#语言篇

.Net 技术精品资料下载汇总: VB.NET 篇

撼世出击: C/C++编程语言学习资料尽收眼底 电子书+视频教程

Visual C++(VC/MFC)学习电子书及开发工具下载

Perl/CGI 脚本语言编程学习资源下载地址大全

Python 语言编程学习资料(电子书+视频教程)下载汇总

最新最全 Ruby、Ruby on Rails 精品电子书等学习资料下载

数据库精品学习资源汇总: MySQL 篇 | SQL Server 篇 | Oracle 篇

最强 HTML/xHTML、CSS 精品资料下载汇总

最新 JavaScript、Ajax 典藏级资料下载分类汇总

网络最强 PHP 开发工具+电子书+视频教程等资料下载汇总

UML 学习电子书下载汇总 软件设计与开发人员必备

经典 LinuxCBT 视频教程系列 Linux 快速学习视频教程一帖通

天罗地网: 精品 Linux 学习资料大收集(电子书+视频教程) Linux 参考资源大系

Linux 系统管理员必备参考资料下载汇总

Linux shell、内核及系统编程精品资料下载汇总

UNIX 操作系统精品学习资料<电子书+视频>分类总汇

FreeBSD/OpenBSD/NetBSD 精品学习资源索引 含书籍+视频

Solaris/OpenSolaris 电子书、视频等精华资料下载索引



# 目录

## 第一部分 第一个网页

第 1 章 理解 HTML 和 XHTML .....	3
1.1 什么是网页 .....	4
1.2 网页的工作原理 .....	5
1.3 网页的组成部分 .....	9
1.4 HTML 的多面性 .....	10
1.5 HTML、XML 与 XHTML .....	12
1.6 总结 .....	13
1.7 问与答 .....	14
1.8 作业 .....	14
1.8.1 测验 .....	14
1.8.2 答案 .....	15
1.9 练习 .....	15
第 2 章 开始创建网页 .....	16
2.1 从简单网页开始 .....	17
2.2 每个网页都必须包含的 HTML 标签 .....	19
2.3 使用分段和分行来组织页面 .....	20
2.4 使用标题突出文本 .....	23
2.5 学习他人的网页 .....	25
2.6 验证网页 .....	26
2.7 总结 .....	27
2.8 问与答 .....	28

2.9 作业 .....	28
2.9.1 测验 .....	28
2.9.2 答案 .....	29
2.10 练习 .....	30
第 3 章 链接到其他网页 .....	31
3.1 链接到另一个网页 .....	31
3.2 在自己的网页间链接 .....	33
3.3 指定网页的地址 .....	36
3.4 在新浏览器窗口中打开链接 .....	37
3.5 链接到 Google Maps .....	38
3.6 添加到实用示例的链接 .....	39
3.7 总结 .....	42
3.8 问与答 .....	42
3.9 作业 .....	43
3.9.1 测验 .....	43
3.9.2 答案 .....	43
3.10 练习 .....	44
第 4 章 发布 HTML 网页 .....	45
4.1 在 Internet 建立 Web 空间 .....	46
4.2 将网页上传到 Web 服务器 .....	47
4.2.1 使用 Microsoft Internet Explorer .....	47

4.2.2	使用 Mozilla Firefox .....	49
4.2.3	使用 Microsoft FrontPage .....	51
4.2.4	使用 FTP 软件 .....	51
4.2.5	使用 Windows Web 发布向导 .....	54
4.3	使文件可下载 .....	55
4.4	其他发布 HTML 的途径 .....	56
4.4.1	将网页放在内部网上 .....	56
4.4.2	将网页发布到磁盘或存储卡上 .....	56
4.5	测试网页 .....	57
4.6	总结 .....	58
4.7	问与答 .....	58
4.8	作业 .....	59
4.8.1	测验 .....	59
4.8.2	答案 .....	59
4.9	练习 .....	60

## 第二部分 使用 HTML 创建实用网页

### 第 5 章 基本文本对齐和格式化 .....63

5.1	新方法和旧方法 .....	64
5.2	对齐网页中的文本 .....	64
5.2.1	段落对齐 .....	65
5.2.2	对齐网页的整个部分 .....	65
5.3	三种 HTML 列表 .....	67
5.4	将列表放在列表中 .....	69
5.5	粗体、斜体和特殊的文本格式化 .....	74
5.6	调整字体 .....	77
5.7	使用特殊字符 .....	81
5.8	总结 .....	83
5.9	问与答 .....	84
5.10	作业 .....	85
5.10.1	测验 .....	85
5.10.2	答案 .....	85
5.11	练习 .....	86

### 第 6 章 创建文本链接 .....87

6.1	使用锚在网页内部链接 .....	87
6.1.1	使用锚标识网页中的位置 .....	87
6.1.2	链接到锚点位置 .....	88
6.2	链接到另一个网页的特定部分 .....	91
6.3	在网页中链接 Email 地址 .....	93
6.4	链接到 Wikipedia .....	97
6.5	总结 .....	99
6.6	问与答 .....	99
6.7	作业 .....	99
6.7.1	测验 .....	99
6.7.2	答案 .....	100
6.8	练习 .....	101

### 第 7 章 创建网页图像 ..... 102

7.1	选择图像软件 .....	102
7.2	必须知道的图像基础知识 .....	104
7.3	准备照片图像 .....	104
7.3.1	裁剪图像 .....	105
7.3.2	调整图像的大小 .....	105
7.3.3	改变图像颜色 .....	106
7.3.4	控制 JPEG 压缩 .....	107
7.4	创建横幅和按钮 .....	108
7.5	减少图像的颜色数 .....	111
7.6	交错 GIF 和渐进 JPEG .....	112
7.7	创建 Web 动画 .....	113
7.8	将图像组装成动画 .....	113
7.9	生成过渡和文本效果 .....	116
7.10	总结 .....	117
7.11	问与答 .....	117
7.12	作业 .....	118
7.12.1	测验 .....	118
7.12.2	答案 .....	118
7.13	练习 .....	118

### 第 8 章 将图像放到网页中 ..... 119

8.1	将图像放到网页中 .....	119
8.2	使用文本描述图像 .....	121

8.3 将图像变为链接.....	123	11.1 创建简单表格.....	159
8.4 对齐图像.....	125	11.2 控制表格的大小.....	162
8.4.1 水平对齐图像.....	125	11.3 表格内部的对齐和跨度.....	163
8.4.2 垂直对齐图像.....	126	11.4 表格背景和间距.....	165
8.5 指定图像的宽度和高度.....	128	11.5 使用表格设计有创意的网页 布局.....	166
8.6 总结.....	129	11.6 总结.....	170
8.7 问与答.....	129	11.7 问与答.....	171
8.8 作业.....	130	11.8 作业.....	171
8.8.1 测验.....	130	11.8.1 测验.....	172
8.8.2 答案.....	131	11.8.2 答案.....	172
8.9 练习.....	131	11.9 练习.....	173
<b>第 9 章 自定义背景和颜色.....</b>	<b>132</b>	<b>第 12 章 使用 CSS 格式化网页.....</b>	<b>174</b>
9.1 背景颜色和文本颜色.....	132	12.1 理解 CSS.....	174
9.2 指定自定义颜色.....	134	12.2 一个基本样式表.....	176
9.3 使用背景图像拼贴.....	135	12.3 CSS 样式初步.....	180
9.4 使用透明的图像.....	137	12.3.1 布局属性.....	181
9.5 创建自己的平铺背景.....	139	12.3.2 格式化属性.....	181
9.6 总结.....	141	12.4 使用样式类.....	184
9.7 问与答.....	141	12.5 内部样式表和内联样式.....	186
9.8 作业.....	142	12.6 样式表快速复习.....	187
9.8.1 测验.....	142	12.7 总结.....	188
9.8.2 答案.....	142	12.8 问与答.....	188
9.9 练习.....	143	12.9 作业.....	189
<b>第 10 章 图像链接和图像映射.....</b>	<b>144</b>	12.9.1 测验.....	189
10.1 为什么不一定需要图像映射 ..	144	12.9.2 答案.....	189
10.2 映射图像中的区域.....	149	12.10 练习.....	189
10.3 创建客户端图像映射.....	151	<b>第 13 章 深入探讨样式表格式化.....</b>	<b>190</b>
10.4 总结.....	154	13.1 使用 CSS 格式化文本的高级 技巧.....	190
10.5 问与答.....	155	13.2 修改链接的外观.....	192
10.6 作业.....	155	13.3 CSS 代码注释.....	193
10.6.1 测验.....	155	13.4 CSS 技巧.....	195
10.6.2 答案.....	156	13.4.1 使用通用类.....	195
10.7 练习.....	156	13.4.2 指定多个选择符.....	196
		13.4.3 混合样式类.....	196
		13.5 重新设计曲棍球队员示例.....	197
<b>第三部分 充满创意的网页设计</b>			
<b>第 11 章 使用表格组织和排列网页.....</b>	<b>159</b>		

13.6 总结 .....	202	16.2.2 添加框架 .....	236
13.7 问与答 .....	203	16.3 在框架和窗口之间链接 .....	237
13.8 作业 .....	203	16.4 在框架中嵌套框架 .....	238
13.8.1 测验 .....	203	16.5 控制框架的边距、边框和 滚动 .....	239
13.8.2 答案 .....	203	16.6 总结 .....	243
13.9 练习 .....	204	16.7 问与答 .....	244
<b>第 14 章 将样式表用于页面布局 .....</b>	<b>205</b>	16.8 作业 .....	244
14.1 定位 .....	205	16.8.1 测验 .....	244
14.2 控制元素的堆叠顺序 .....	209	16.8.2 答案 .....	245
14.3 使用边距 (margin) .....	212	16.9 练习 .....	246
14.4 为安全起见而使用填充 (padding) .....	213		
14.5 确保内容对齐 .....	214	<b>第四部分 动态网页</b>	
14.6 管理文本流 .....	215	<b>第 17 章 面向非程序员的网页脚本         编程 .....</b>	<b>249</b>
14.7 利用样式表模板 .....	215	17.1 必须了解的脚本知识 .....	249
14.8 总结 .....	220	17.2 使用交互式加亮使网页更 有趣 .....	251
14.9 问与答 .....	220	17.3 预加载图像以提高速度 .....	254
14.10 作业 .....	220	17.4 创建动画条幅广告 .....	256
14.10.1 测验 .....	220	17.5 显示随机格言 .....	258
14.10.2 答案 .....	221	17.6 创建虚拟鱼塘 .....	261
14.11 练习 .....	221	17.7 在网站上使用 Google 广告 赚钱 .....	263
<b>第 15 章 创建打印友好的网页 .....</b>	<b>222</b>	17.8 JavaScript 的广阔世界 .....	266
15.1 如何成为打印友好的网页 .....	222	17.9 总结 .....	267
15.2 应用与介质相关的样式表 .....	224	17.10 问与答 .....	267
15.3 设计用于打印网页的样式表 .....	226	17.11 作业 .....	267
15.4 预览网页的打印结果 .....	229	17.11.1 测验 .....	267
15.5 总结 .....	230	17.11.2 答案 .....	268
15.6 问与答 .....	230	17.12 练习 .....	268
15.7 作业 .....	230		
15.7.1 测验 .....	230	<b>第 18 章 使用 HTML 表单收集信息 .....</b>	<b>269</b>
15.7.2 答案 .....	231	18.1 HTML 表单的工作原理 .....	269
15.8 练习 .....	231	18.2 创建表单 .....	270
<b>第 16 章 使用框架创建多页面布局 .....</b>	<b>232</b>	18.3 接受文本输入 .....	273
16.1 什么是框架 .....	232	18.4 标识每项表单数据 .....	274
16.2 创建框架集 .....	233	18.5 在表单中包含隐藏数据 .....	274
16.2.1 创建框架集文档 .....	234		



18.6 表单输入控件.....275	20.6 创建商品清单幻灯片..... 311
18.6.1 复选框.....275	20.7 总结..... 314
18.6.2 单选按钮.....276	20.8 问与答..... 314
18.6.3 选择列表.....276	20.9 作业..... 315
18.6.4 文本区域.....277	20.9.1 测验..... 315
18.7 提交表单数据.....277	20.9.2 答案..... 315
18.8 使用 PayPal Buy Now 按钮.....279	20.10 练习..... 315
18.9 总结.....283	<b>第 21 章 创建自己的博客..... 316</b>
18.10 问与答.....284	21.1 什么是博客..... 316
18.11 作业.....284	21.2 在 Blogger.com 上创建 博客..... 318
18.11.1 测验.....284	21.3 创建第一篇博客文章..... 320
18.11.2 答案.....285	21.4 使用 CSS 修饰博客..... 321
18.12 练习.....285	21.5 在网站中链接到博客..... 325
<b>第 19 章 在网页中嵌入多媒体.....286</b>	21.6 更新博客..... 326
19.1 将多媒体内容放到网页中.....287	21.7 总结..... 327
19.2 链接到多媒体文件.....287	21.8 问与答..... 327
19.3 使用 Windows Media Player 嵌入视频.....290	21.9 作业..... 327
19.4 使用 RealPlayer 嵌入音频.....293	21.9.1 测验..... 327
19.5 自动嵌入合适的播放器.....295	21.9.2 答案..... 328
19.6 总结.....298	21.10 练习..... 328
19.7 问与答.....299	
19.8 作业.....300	<b>第五部分 建立网站</b>
19.8.1 测验.....300	<b>第 22 章 组织和管理网站..... 331</b>
19.8.2 答案.....300	22.1 一个网页就足够时..... 332
19.9 练习.....300	22.2 组织简单的网站..... 333
<b>第 20 章 使用 HTML 和 CSS 使 eBay     拍卖更有趣.....302</b>	22.3 组织较大型网站..... 335
20.1 为什么要在 eBay 上使用 HTML 和 CSS.....303	22.4 编写易于维护的 HTML 代码..... 337
20.2 理解如何将 HTML 应用于 eBay.....304	22.4.1 使用注释说明代码..... 337
20.3 用 HTML 创建 eBay 商品 清单.....305	22.4.2 缩进使代码更清晰..... 338
20.4 提供到 About Me 网页的 链接.....307	22.5 总结..... 339
20.5 修改商品网页的背景.....309	22.6 问与答..... 340
	22.7 作业..... 340
	22.7.1 测验..... 340
	22.7.2 答案..... 341
	22.8 练习..... 341

<b>第 23 章 帮助人们找到你的网页</b>	342
23.1 宣传网站	342
23.2 在主要的搜索网站列出你的网页	343
23.3 为搜索引擎提供提示	344
23.4 自动加载另一个网页	351
23.5 指出网页的完整地址	353
23.6 总结	354
23.7 问与答	354
23.8 作业	355
23.8.1 测验	355
23.8.2 答案	355
23.9 练习	356
<b>第 24 章 超越传统网站</b>	357
24.1 超越 Web 的 HTML	357
24.2 从计算器到无线通信设备	358
24.3 HTML 作为新的用户界面	358
24.4 数字媒体革命不会出现在电视上	359
24.5 为未来做准备	360
24.6 未来的 HTML 应用	361
24.6.1 RSS 和网页聚合	363
24.6.2 使用内容管理系统	364
24.6.3 使用 XHTML Mobile 编写网页	365
24.7 总结	367
24.8 作业	368
24.9 练习	368

## 第六部分 附录

<b>附录 A 读者最常问的问题</b>	371
----------------------	-----

<b>附录 B 完整的 XHTML 1.1 和 CSS 2 快速参考</b>	376
B.1 XHTML 结构	377
B.2 XHTML 文本短语和段落	379
B.3 XHTML 文本格式化元素	382
B.4 XHTML 列表	383
B.5 XHTML 链接	384
B.6 XHTML 表格	385
B.7 XHTML 框架	387
B.8 XHTML 嵌入内容	389
B.9 XHTML 样式	390
B.10 XHTML 表单	390
B.11 XHTML 脚本	393
B.12 XHTML 常用属性和事件	394
B.13 CSS 尺寸样式属性	394
B.14 CSS 文本和字体样式属性	395
B.15 CSS 背景样式属性	397
B.16 CSS 边框样式属性	397
B.17 CSS 边距样式属性	399
B.18 CSS 填充样式属性	400
B.19 CSS 布局和显示样式属性	401
B.20 CSS 列表和项目符号样式属性	402
B.21 CSS 表格样式属性	403
<b>附录 C 从 HTML 迁移到 XHTML</b>	404
C.1 XHTML 1.1 和 HTML 4 之间的差别	404
C.2 XHTML 和文档有效性	405
C.3 声明 XHTML 名称空间	405
C.4 将 HTML 文档转换为 XHTML	406

# 第一部分

## 第一个网页

第 1 章 理解 HTML 和 XHTML

第 2 章 开始创建网页

第 3 章 链接到其他网页

第 4 章 发布 HTML 网页



# 计算机精品学习资料大放送

软考官方指定教材及同步辅导书下载 | 软考历年真题解析与答案

软考视频 | 考试机构 | 考试时间安排

Java 一览无余: Java 视频教程 | Java SE | Java EE

.Net 技术精品资料下载汇总: ASP.NET 篇

.Net 技术精品资料下载汇总: C#语言篇

.Net 技术精品资料下载汇总: VB.NET 篇

撼世出击: C/C++编程语言学习资料尽收眼底 电子书+视频教程

Visual C++(VC/MFC)学习电子书及开发工具下载

Perl/CGI 脚本语言编程学习资源下载地址大全

Python 语言编程学习资料(电子书+视频教程)下载汇总

最新最全 Ruby、Ruby on Rails 精品电子书等资料下载

数据库精品学习资源汇总: MySQL 篇 | SQL Server 篇 | Oracle 篇

最强 HTML/xHTML、CSS 精品资料下载汇总

最新 JavaScript、Ajax 典藏级资料下载分类汇总

网络最强 PHP 开发工具+电子书+视频教程等资料下载汇总

UML 学习电子书下载汇总 软件设计与开发人员必备

经典 LinuxCBT 视频教程系列 Linux 快速学习视频教程一帖通

天罗地网: 精品 Linux 学习资料大收集(电子书+视频教程) Linux 参考资源大系

Linux 系统管理员必备参考资料下载汇总

Linux shell、内核及系统编程精品资料下载汇总

UNIX 操作系统精品学习资料<电子书+视频>分类总汇

FreeBSD/OpenBSD/NetBSD 精品学习资源索引 含书籍+视频

Solaris/OpenSolaris 电子书、视频等精华资料下载索引



## 第 1 章

# 理解 HTML 和 XHTML



在使用 HTML 创建网页前，需要了解一下什么是网页，如何浏览和编辑网页以及使用网页可以做什么等背景知识。对 HTML 和 XHTML 之间的区别以及这两种不同的语言用于实现相同的功能——创建网页——的原因等，有个基本的理解也将有所帮助。本章将快速地介绍 HTML 和 XHTML 的基础知识，为网页制作者和发布者提出一些实用技巧。但是不要止于纸上谈兵，需要看看真正的网页以及网页的 HTML 代码。

### 练习

下面列出了开始阅读本书其余内容前需要做的事情。

1. 获得一台计算机。我使用安装了 Windows XP 的计算机来测试网页和截取本书的图片，但读者可以使用任何版本的 Windows、Macintosh 或 UNIX 机器来创建和浏览网页。计算机的速度本身并不会很影响访问网页的速度；但是计算机的调制解调器（modem）或网卡（NIC）必须至少是 56kbit/s，且越快越好。如果能够通过电缆 modem、DSL modem 或无线访问点，那就更好了。

2. 获得到 Internet 的连接。可以使用计算机的 modem（调制解调器）利用 Internet 服务提供商（ISP）的拨号服务，或者连接学校或公司的局域网。现在大多数 ISP 提供的 Internet 拨号的价格都在每月 20 美元以下。正确设置的细节可从提供 Internet 服务的 ISP、学校或公司获得。现在，许多公共场所，如机场、书店和图书馆也提供免费的无线 Internet 服务，如果有带有 Wi-Fi 网络支持的笔记本就可以利用这些服务。

**注意：**不知道怎么找到 ISP？最好的方法就是到网上找（使用朋友已经可以连接到 Internet 的计算机）。

**By the Way**

3. 获得网页浏览器软件。计算机需要这样的软件来获取和显示 HTML 网页。当前最流行的网页浏览器程序是 Microsoft Internet Explorer、Mozilla Firefox、Opera 和 Safari（只能用于 Macintosh）。安装几个浏览器是不错的主意，这样就可以试验一下，从而保证网页在这些

浏览器上看起来都是一致的；不能假设其他人使用的浏览器类型。网页浏览器可从软件零售商那里获得，也可从 Internet 上免费下载：[www.microsoft.com/](http://www.microsoft.com/)、[browser.netscape.com/](http://browser.netscape.com/)、[www.mozilla.com/firefox/](http://www.mozilla.com/firefox/)、[www.opera.com/](http://www.opera.com/)和 [www.apple.com/safari/](http://www.apple.com/safari/)。

### By the Way

**注意：**虽然所有网页浏览器在信息的处理上基本是一致的，但它们之间存在着一些区别，使得网页在不同的浏览器上看起来不一定一致。应在多个浏览器上测试网页，保证它们在不同浏览器的显示是基本一致的。

4. 探索！使用网页浏览器看看 Internet 上那些与你要创建的网页的内容或外观类似的网页。看看什么样的网页使你感到困惑，什么样的网页吸引你继续看其他网页，以及什么样的网页使你不断地返回一些网页。如果对某个特定的话题感兴趣，可以在流行的搜索引擎如 Google ([www.google.com](http://www.google.com)) 上搜索。

### By the Way

**注意：**如果计划将 HTML 页面放到 Internet 上（而不是将它们发布到光盘或局域网上），需要将它们传输到一天 24 小时都与 Internet 连接的计算机上。提供 Internet 访问的公司或学校也可能允许将网页放在他们的计算机上；如果不允许，可能需要找其他公司购买托管服务来放置网页。

现在可以开始使用本书学习 HTML，然后等所有网页完成后再找 Internet 托管服务商。然而，如果希望在第一个网页完成后就可以发布到 Internet 上，可以先阅读第 4 章。

## 1.1 什么是网页

在月球上还没有任何人的足迹时，一些有远见的人就决定探索是否能够将几个主要的计算机网络连接在一起，其中涉及到许多人和故事，但最终的结果是“所有网络之母”，即 Internet。

到 1990 年，通过 Internet 访问信息需要的技术相当高。实际上，它是如此困难，以至于具有博士学位的物理学家在交换数据时也经常受困。其中有一位物理学家，也就是现在著名的 Tim Berners-Lee，发明了一种通过“超文本”链接交叉引用 Internet 上文本的简单方法。这并不是一种新的想法，但他的超文本标记语言（HTML）开始流行起来，而有些野心勃勃的超文本项目反而处境艰难。

“超文本”的原义是指以页面之间交叉引用的电子形式存储的文本。现在这个术语有了更广泛的含义，它可以指可被其他对象链接的任何对象（文本、图像和文件等）。

超文本标记语言描述包含文本、图像和其他信息的网页是如何组织和相互链接的。

到 1993 年，全世界大约有 100 台计算机提供 HTML 页面的访问服务。这些互相连接的页面称为“万维网”（World Wide Web, WWW）。同时也出现了一些允许用户浏览网页的网页浏览器程序。由于 Web 的流行，一些程序员很快开始编写既可以查看网页中的文本也可以查看图形图像的网页浏览器。Marc Andressen 就是其中的一位程序员，他后来因为销售世界上最流行的网页浏览器 Netscape Navigator 而变得富有、出名。

现在，HTML 页面是 Internet 的标准界面。它们可以包含动画、声音、视频、完全交互



性的程序以及原始文本。每天都有成千上万来自全世界各地的 Web 服务器计算机上的数百万网页被检索和浏览。顺便提一下,“Web”这个词指网页互相链接形成巨大的信息网,很像蜘蛛网。这也就是为什么有时候我们会听到“crawler”(爬虫)这个词,它指在 Web 上收集网页信息的程序。

**休息一下:** 为防止读者造成 Web 服务器必须是巨大的计算机、在提供网页服务时消耗大量功耗的误解,这里将介绍世界上最小的 Web 服务器。麻省大学的研究生以小于 1 美元的成本搭建了一个世界上最小的 Web 服务器,大概只有火柴头大小。详情请参见 <http://www-ccs.cs.umass.edu/~shri/iPic.html>。

**Coffee**  
**Break**

因为电视电缆、现代化的电话线路、卫星直接传输以及公共的、私有的无线网络等高速 Internet 连接方式日益普遍,Web 很快成为大众媒体。现在直接在电视机上附加一个小盒子就可以浏览 Web,而不必使用计算机,这种设备的成本在未来几年内很可能急剧下降。换句话说,在不久的将来,Web 浏览将不必依赖于计算机。

Internet 不再是使用 HTML 的惟一场合。大多数企业私有网络(称为内部网)现在也使用 HTML 来提供商业信息给员工和客户。HTML 现在也是在 CD-ROM 和高密度的 DVD 上发布展示可以选择的界面。Microsoft 甚至将 HTML 直接集成到 Windows 操作系统中,使计算机中的每个存储文件夹都与一个 HTML 页面以及指向其他文件夹与页面的超文本链接相关联。

简单地说,HTML 无处不在。您很幸运找到正确的方式来探索 HTML 网页的工作原理以及如何创建网页。

**注意:** 实际上有两种 HTML。一种称为 HTML 4,另一种称为 XHTML 1.1。X 表示 eXtensible(可扩展的),在这一课程后面读者将了解到它为什么不叫做 HTML 5。你还将了解到它与另一种新语言 XML 之间的关系。

**By the**  
**Way**

从一开始就应该了解的是,本书中的所有示例与 HTML 4 和 XHTML(以及 XML)都是兼容的,并且应该也能够与所有解释网页语言的软件的未来版本兼容。

如果你有介绍创建网页的其他书,在那些书中的示例 HTML 看起来可能和本书中的示例有点不同。即使那些书中旧的格式在当前的网页浏览器中也能够正常显示,但强烈推荐使用本书介绍的更现代的方法,这将保证网页在将来的生命周期尽可能地长久。

## 1.2 网页的工作原理

浏览网页时,网页看起来就像纸上的页面。乍一看,显示网页的过程似乎很简单:你告诉计算机要看哪个网页,然后这个网页就出现在你的屏幕上。如果网页存储在你的计算机硬盘上,这一过程是瞬时完成的。如果网页在其他计算机上,可能需要等待网页的获取。

诚然,网页可以完成许多纸页无法实现的便利功能。例如,不能在纸上指向“转第 57 页”然后第 57 页就自动出现在你的眼前。又如,不能在纸印的订单表单的底部单击,5 秒之后就能进入公司的订单执行系统。在纸上你也不能看到动画或听到与你交谈的声音(新发明

的贺卡除外)。而所有这些在网页中都很常见。

除此之外，作为网页开发人员，还需要知道网页与纸张之间的一些更深层次的区别。首先，显示在计算机上的页面实际上可能是位于许多不同计算机的文件中的元素集合。实际上，创建包含来自澳大利亚的文本、来自俄罗斯的图片以及来自加拿大的声音的网页是可能的(虽然并不常见)。

图 1.1 是在 Microsoft Internet Explorer (当前世界上最流行的浏览网页的软件) 中显示的一个典型页面。图 1.1 中的页面在 Netscape、Firefox 或其他网页浏览器上看起来将是大致相同的。我说“大致”是因为网页浏览器对网页的解释不总是完全相同，即使理论上应该这样。为简单起见，现在假设所有主要的网页浏览器对网页的显示没有大的区别。

图 1.1  
网页浏览器将分开的  
文本和图像文件  
组合起来，显示为  
集成的页面



By the  
Way

**注意:**虽然 Internet Explorer 是当前用户数量最多的网页浏览器,但是 Mozilla Firefox 也很受用户喜爱,有可能某一天会取代 Internet Explorer 的位置。Firefox 在最近的浏览器大战中排行第二,远远超过 Opera 和 Apple 的 Safari 浏览器。后来的浏览器开发商在 Web 标准的遵守和安全问题的解决这两个方面做得比 Microsoft 好很多,这是 Firefox 和 Opera 的流行程度在快速提高的原因。

网页浏览器(如 Internet Explorer)执行的功能不只是检索文件,然后显示在屏幕上。它们实际上将页面的各个组成部分组合起来,然后根据网页制作者隐藏在文本中的命令排列这些部分。这些命令是使用 HTML 编写的。

网页浏览器是解释 HTML 命令并收集、排列和显示网页组成部分的计算机软件。

程序清单 1.1 列出了用于创建图 1.1 所示页面的文本,包含 HTML 命令。可用任何字处理软件或文本编辑器来读取和编辑这个文本文件。这个文件包含一些怪异的符号和代码文字,看起来有点奇怪,但是文本文件本身并不包含任何嵌入的图像、粗体文本或其他特殊的格式。符号“<”和“>”之间的文本是 HTML 标签。

程序清单 1.1 用于创建图 1.1 所示页面的文本

```
<?xml version="1.0" encoding="UTF-8"?>
```



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Tall Tales - The Game of Legends and Creative One-Upmanship</title>
  </head>

  <body>
    <p>
      Tall Tales&reg; Pocket Edition
    </p>
    <p>
      Tall Tales&reg; Pocket Edition is the first game in the Tall Tales line,
      and is a compact game designed for the ultimate in flexible game play.
      Tall Tales Pocket Edition includes 120 trivia cards, and is for curious
      and crafty storytellers ages 10 and up. No board, writing, or dice
      rolling makes Tall Tales Pocket Edition a great casual or travel game.
      We like to call it a "lounge game."
    </p>
    <p>
      <a href="tpe_box_pic.jpg">
      </a><br />
      Tall Tales&reg; Pocket Edition<br />
      Suggested retail: $14.95<br />
      Dimensions: 4 1/2" x 4 1/2" x 1 3/4"<br />
      Made in the U.S.A.
    </p>
    <p>
      Each type of card in Tall Tales corresponds to a certain type of question
      or topic, and is themed to a legendary creature (Loch Ness Monster,
      Bigfoot, and Alien). The Tall Tales&reg; and Zinger&reg; cards are trivia
      cards that include questions and answers, while the Topper&reg; cards
      provide topics that you use as the basis for sharing a whopper of a story.
      Playing the game simply involves drawing and playing these cards:
    </p>
    <p>
      <a href="pg_games_ttcards_lg.gif"></a>
      <a href="pg_games_topcard_lg.gif"></a>
      <a href="pg_games_zngcard_lg.gif"></a>
    </p>
    <p>
      To learn more about Tall Tales, visit the Tall Tales Web site at <a
      href="http://www.talltalesgame.com/">www.talltalesgame.com</a>.
    </p>
```

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

Internet Explorer 读取程序清单 1.1 所列 HTML 代码中的命令，然后显示出你在图 1.1 中看到的所有图像和文本。在文本中包含的 HTML 命令告诉浏览器，查找独立的图像文件并将它们显示在文本中。其他的命令告诉浏览器如何将页面上的文本分行。虽然该程序清单中有些代码对 HTML 新手来说有点奇怪，但从代码中仍可以辨识出显示在页面中的内容。

**Did you Know?**

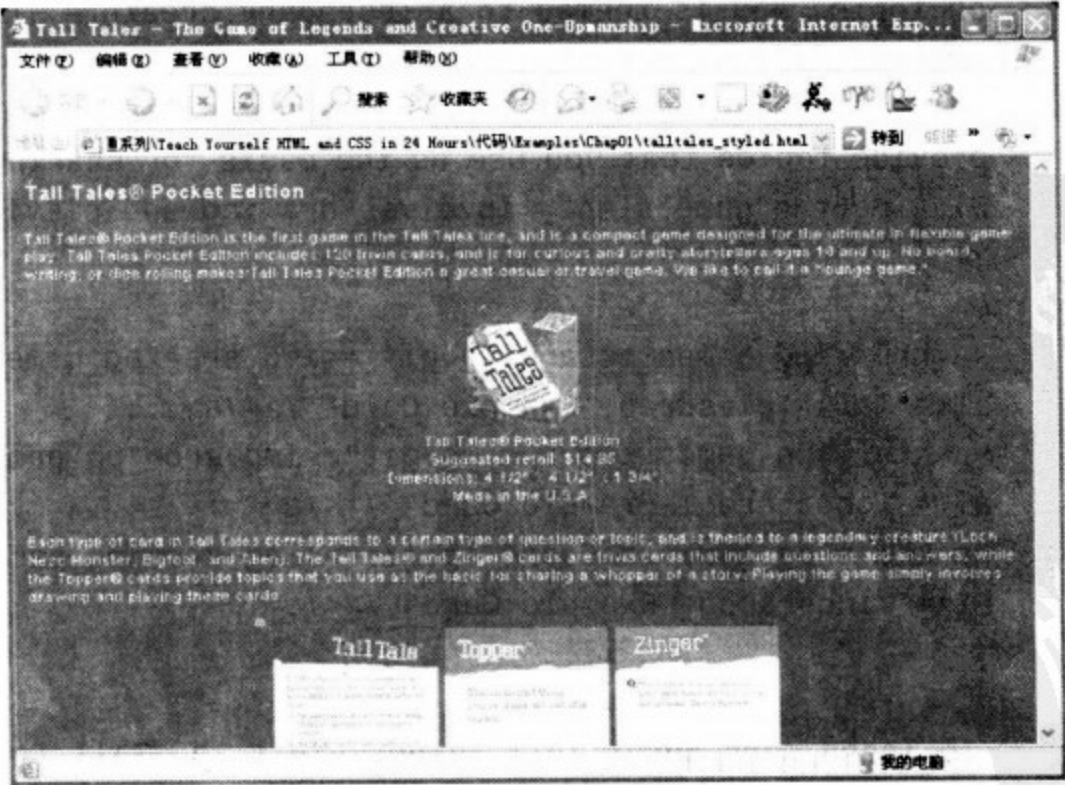
**提示：**要查看网页的 HTML 命令，只要右击鼠标（如果使用的是 Macintosh 计算机，则按住 Ctrl 键然后单击），然后，如果是 Internet Explorer 则在弹出菜单中选择“查看源文件”，如果是 Firefox 则在弹出菜单中选择“查看页面源文件”。这是了解 HTML 工作原理与学习他人示例的好方法。有些网站可能禁止单击鼠标右键，这种情况下可以使用“查看”菜单中的选项“源文件”（Internet Explorer）或“页面源文件”（Firefox）。

有些网页使用一种称为“框架”的高级特性，同时可以显示多个 HTML 页面。在大多数网页浏览器中，可以在任何框架上右击，然后选择“查看框架源文件”，来查看帧的 HTML 命令。

无论采用何种方式查看网页的源文件，需要牢记的是，很多商业网页使用复杂的 HTML 代码，它们可能难以阅读和理解。因此，如果查看流行网站的源文件，发现其复杂程度超出想象，也不要气馁。

图 1.1 显示和程序清单 1.1 列出的网页的格式化比较少，这是有意设计的，为使代码较少。HTML 显然可以包含更多的代码，下面将举例说明应用程序运用很少的格式也能显著地影响网页的外观。图 1.2 包含同样的网页，只是增加了一些格式，包括一个背景图像和一些字体调整。

图 1.2  
为网页添加样式而不改变网页内容就能大大改变网页的外观



这个包含样式的网页版本与没有样式的版本包含的内容是相同的，但最终的结果看起来很不一样。在本书中，读者将学习如何将样式运用到 HTML 代码中来取得生动的视觉效果。

### 1.3 网页的组成部分

读者马上就要学习如何理解和编写 HTML 命令。现在要注意一点：创建网页只需要输入一些文本。可以用任何字处理软件或文本编辑器来输入和保存文本，然后用任何 HTML 兼容的软件（网页浏览器）打开文本文件，它就是一个网页。

**注意：**建议不要使用 Microsoft Word 或其他 HTML 兼容的字处理软件来编辑第一个 HTML 网页——这些程序大多会试图以奇怪的方式重新编写 HTML 页面，最后你可能完全无法理解。可以使用诸如 Microsoft FrontPage 或 Macromedia Dreamweaver 的 Web 开发工具，但是要切换视图保证能够编辑原始的 HTML 代码。

By the Way

如果希望网页中包含图像、声音、动画、视频和交互式程序，这些内容并不是直接插入到文本文件中，就像在大多数面向打印的排版程序，如 Microsoft Word 和 Adobe Acrobat 中创建文档那样。相反，需要输入 HTML 文本命令，告诉网页浏览器到哪里查找这些媒体文件。媒体文件本身是独立的，虽然网页浏览器在显示页面时使它们看起来像是同一文档的一部分。

例如，程序清单 1.1 中的 HTML 文档直接引用了 4 个独立的图形图像。图 1.3 显示了在 Microsoft Photo Editor 中编辑的这 4 个图像文件，Microsoft Photo Editor 是在随 Microsoft Office 发布的。

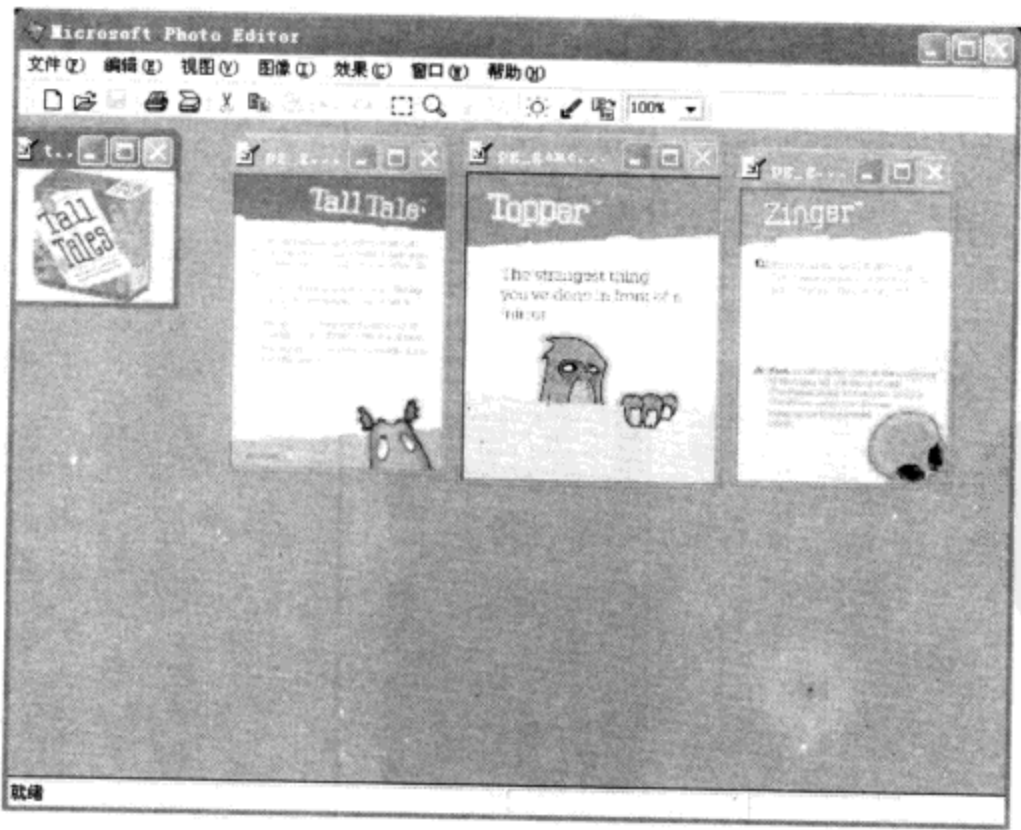


图 1.3  
虽然文本和图像在图 1.1 和图 1.2 中的显示是集成的，但图像文件实际上是独立存储、可独立编辑的

可以使用喜欢的任何图形软件来修改或替换这些图像，包括所有 Windows 版本自带的简单画图程序。对图像的修改可能对网页的显示有很大影响，即使不对 HTML 文本文件做任何修改。也可只存储一个图像文件而在任意多的网页上使用同一个图像。在第三部分将学到更

多关于如何在网页中包含图像的知识。

**Did you know?**

**提示:**创建 HTML 页面有两种方法:可以使用文本编辑器输入文本和 HTML 命令,也可以使用图形软件来生成 HTML 命令。

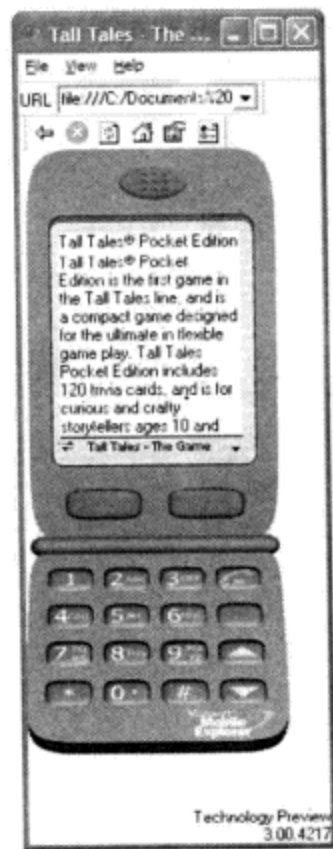
如果使用显示实际 HTML 文本的编辑器,将能够更好地进行本书的学习,也更容易学习 HTML。任何字处理软件或文本编辑器——包括 Windows 的记事本和 Macintosh 的 TextEdit 编辑器都可以。实际上,鼓励开始使用简单的文本编辑器,这样更简单——字处理软件可能包含格式化信息,如果不注意很难删除。

现在,建议不要使用所见即所得的图形 Web 编辑器,如 Microsoft FrontPage 或 Macromedia Dreamweaver,除非在这些工具中使用可以直接编辑源代码的视图。你会发现在刚开始学习 HTML 时,使用简单的文本编辑器将更容易,学习效果更好。对网页的工作原理有了进一步的理解后,可以使用可视化工具。然而,大部分可视化工具也提供代码视图作为可视化工具的文本编辑器,因此开始时也可以使用可视化工具。

1.4 HTML 的多面性

有一点很重要:网页可能有许多不同的外观,这取决于谁在浏览它以及使用什么来浏览它。图 1.4 与图 1.1 是同一网页,图 1.4 是在 Microsoft 的 Mobile Explorer 仿真器的显示结果——Mobile Explorer 仿真器用于测试移动设备中的网页显示。大多数手机屏幕的宽度和高度都不超过 200 像素,当它们显示网页时将受到很大的限制。

图 1.4  
图 1.1 所示的网页在 Pocket PC 的 Internet Explorer 上看起来很不一样



可以看到,手机的网页视图与在计算机上的显示截然不同。实际上,页面只显示了文本而不显示图像,因为没有足够的屏幕空间来显示图像。了解网页的这点区别对于早期的 HTML



学习很重要，这样就能记住你无法完全控制网页是如何被显示的，或者更确切地说，你无法控制页面显示时的区域大小。除此之外，浏览器在渲染网页的实际内容时是基本一致的，包括决定哪些内容是最重要的（文本）以及屏幕大小有限制时哪些内容（图像）可以忽略。

**注意：**虽然手机屏幕通常都很小，但也有一些大一点的“智能电话”，如 Pocket PC、Treo 设备和 BlackBerry 设备。这些手持设备的屏幕大小通常在 240×320 的范围内，这比普通手机屏幕要大很多，但仍比现在大多数台式机使用的 640×480、800×600 及更高屏幕分辨率小很多。

By the  
Way

这里要说明的一点是，大多数网页在主要的台式机网页浏览器中的显示都是一样的，它们在 PC、Macintosh 和 UNIX 机器上看起来也是一样的。例如，图 1.1 所示的页面在这些机器上的显示都是一样的，只要所有机器显示的窗口大小、字体和程序设置是一样的。

下面要说的是不好的方面。即使用户使用同一网页浏览器的同一版本，仍可以通过选择不同显示选项或改变浏览窗口的大小来改变网页显示的结果。所有主要网页浏览器都允许用户根据自己的需要替换网页制作者指定的背景和字体。屏幕分辨率、窗口大小和可选的工具栏也可以改变用户第一次查看网页时看到的内容。图 1.4 所示的手机示例很清楚地说明了这一点。

**休息一下：**在本课前面，介绍了世界上最小的 Web 服务器。因为现在我们正在讨论的是屏幕大小和网页，现在就介绍世界上最小的网站。可以访问 <http://www.guimp.com/>，看看这个功能完整的网站只占了 18×18 像素的屏幕大小。真是令人印象深刻！

Coffee  
Break

继续学习网页如何随浏览器和显示设置的不同而改变前，先看看图 1.5 所示的页面，它在 Firefox 浏览器中以 800×600 的分辨率大字体显示。将该图与图 1.2 比较，图 1.2 是同一网页在 Internet Explorer 中以 800×600 的分辨率默认字体设置显示。不幸的是，作为网页制作者，无法直接控制某台计算机的显示设置；浏览网页的人总是可以选择他们喜欢的浏览器配置。你需要做的是将网页设计成在主流浏览器默认配置下的显示尽可能一致。

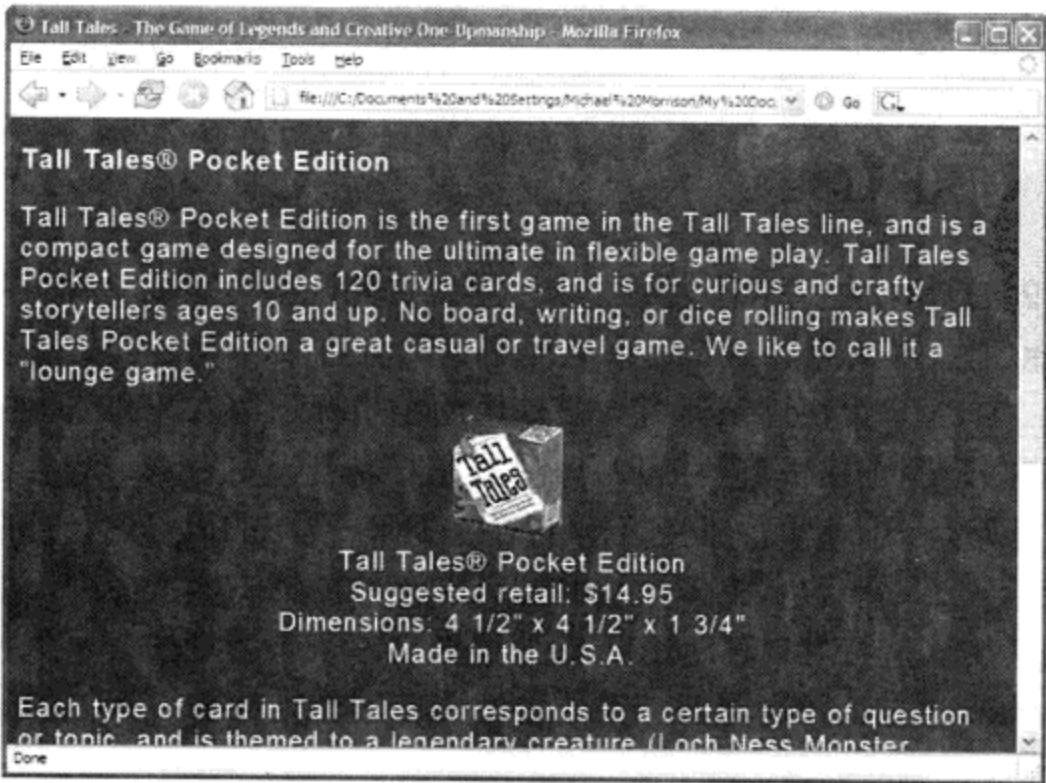
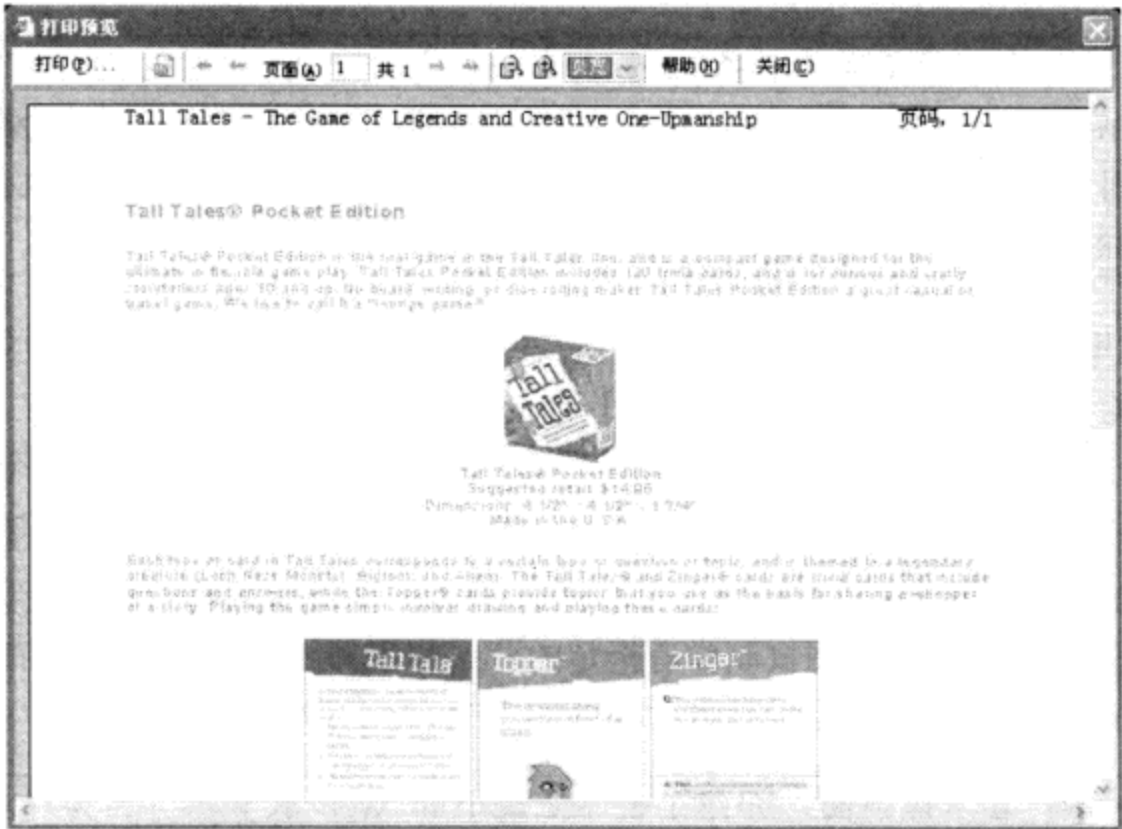


图 1.5  
与图 1.2 相同的页面，只是由 Mozilla Firefox 以大字体显示

甚至不能假设用户会在计算机显示器上显示网页。图 1.1、图 1.2 和图 1.5 也可能显示在低分辨率的电视屏幕或高分辨率的打印预览（见图 1.6）中。

图 1.6  
将网页发送给打印机时，网页浏览器通常将网页的背景改为白色



学习创建网页时，要记住网页被浏览时可能有许多种不同的形式。有些网页制作者可能只尽力使网页在自己的计算机上看起来很完美，而当他们在其他人的计算机上发现网页看起来不一样时就会感到失望。更糟糕的是，有些制作者傻傻地在网页上放置一些信息，要求每个人将浏览窗口大小和字体设置改为与制作者的计算机相同，或者声明：“本网页最好在这样的设置下浏览”。如果你遇到这样的信息，相信你和其他人一样只是忽略它。

在第三部分，读者将发现许多技巧可以保证网页在大多数情况下都能保持美观。

**By the Way**

**注意：**在本书中你会遇到许多示例网页。在本书的网站——Sams Publishing 网站（<http://www.sampublishing.com/>）可以找到这些示例，同时还有更多的示例可以研究。

## 1.5 HTML、XML 与 XHTML

本课之前提到，最初的 HTML 是非常伟大的技术，因为它允许科学家以一种有效的、相对结构化的方式在 Internet 上共享信息。直到后来出现网页浏览器，HTML 开始更多地应用于编码而不是学术论文中。HTML 很快地从研究人员使用的小小标记语言成为完善的在线发布语言。当 HTML 因为可以图形浏览而变得生动后，网页开发人员开始为这一语言添加各种特性。虽然这些新特性一开始是很不错的，但它们破坏了 HTML 设计的简单性，且在浏览器如何显示网页这一问题上带来了不一致。大多数新特性只能在一种浏览器上使用，如果使用了不合适的浏览器就无法使用这些特性。HTML 开始变得像对本来不应动的房子进行修理。结果便是，这个时期出现的一些与具体浏览器相关的特性现在被作为标准，其他的一些则被完全遗弃。

与大多数改革一样，Web 的诞生也很混乱，对 HTML 的修改正反映了这种混乱。在过去的几年中，已经做了大量工作来解决 HTML 的不一致性，并试图恢复这一语言的秩序。HTML 的混乱导致了网页浏览器必须猜测如何显示网页，这并不是件好事。理想情况下，网页设计者应能够确切地定义网页如何显示，并使网页的显示不管在哪种浏览器或操作系统上都是一样的。设计者甚至应该能够定义网页指的是什么，并使这些内容显示在不同的浏览器平台上都是一样的。这样的目标还比较遥远，但现在 XML (eXtensible Markup Language, 可扩展标记语言) 在这一目标的实现发挥着重要作用。

XML 用于创建标记语言，标记语言用于描述结构化信息，如 HTML。

XML 是用于创建特定语言如 HTML 的一种通用语言。我知道这听起来可能有点奇怪，它指的是 XML 提供一种基本结构以及标记语言必须遵守的一组规则。使用 XML 可以创建一种独特的标记语言来描述任何一种信息，包括网页。知道 XML 是创建其他标记语言的语言后，可以使用 XML 创建自己的 HTML 版本。例如，甚至可以创建一种名为 BCCML (Bootle Cap Collection Markup Language, 杯瓶收藏标记语言) 的标记语言，用于创建和管理稀有杯子和瓶子。XML 提供了基础规则，以统一的方式来组织信息，且信息可以是来自网页到杯子和瓶子的任何信息。

你可能会想，杯子和瓶子与 Web 没有任何关系，为什么要提它们呢？原因是 XML 不全是关于网页的。XML 实际上比 Web 更广，它可用于表示任何计算机上的任何信息。想象一下全球计算机、手机、手持计算机、电视和广播上的信息，就能理解为什么 XML 不仅用于网页。然而，XML 的第一个应用确实是用于恢复 Web 的一些秩序，也就是读者学习 HTML 时将学习的 XML。

如果 XML 比 HTML 能够更好地描述数据，这意味着 XML 将取代 HTML 成为 Web 的标记语言吗？不。XML 并不是 HTML 的替代品，HTML 是相对不那么结构化的语言，它可以从 XML 的规则得到改进。这两种技术自然结合，使 HTML 遵守 XML 的规则和结构。为实现这种结合，出现了一种新版本的 HTML，即 XHTML。读者在本书中学习的就是 XHTML，因为它是更清晰的 HTML 版本。

XHTML (Extended Hypertext Markup Language, 扩展超文本标记语言) 是 HTML 的一个版本，它与 XML 严格的规则兼容。

**注意：**在本课程前面，提到的 HTML 是指 HTML 4，这是 HTML 的最新版本。XML 和 XHTML 也有几个版本，XML 1 是 XML 的最新版本，XHTML 2 是 XHTML 的最新版本。由于 XHTML 2 很新，尚未在主要的网页浏览器中得到支持，因此本书将坚持使用 XHTML 1.1。在继续阅读本书和创建网页时，读者将学习到更多关于 XHTML 1.1、HTML 4 以及它们之间的关系的细节。

By the  
Way

## 1.6 总结

这一章介绍了什么是网页以及它们如何工作的基础知识。读者学习了 HTML 命令包含在网页的文本中，但图像和其他媒体信息存储在独立的文件中。还学习了创建 HTML 命令时为

什么自己输入 HTML 文本通常比使用图形编辑器更好，特别是刚开始学习 HTML 时。读者看到了一个网页在不同的地方看起来可能很不一样，这取决于用于显示它的软件和硬件。最后，还学习了 XML 和 XHTML 以及它们为什么与 HTML 有关。

## 1.7 问与答

问：我在第一个网页上遇到了问题，它不能工作，我做错什么了吗？

答：第一个网页总是最难的。要了解如何对可能出错的地方进行逐步的分析以及如何修正错误，请参考附录 A（当遇到问题而在书中其他地方找不到时，将发现这一附录很方便）。

问：我还不是很清楚“网页”与“HTML 页面”之间的区别是什么。还有，它们与“主页”或“网站”的区别又是什么？

答：从技术上说，“网页”是指位于 Internet 而不是在自己计算机硬盘上的页面。但实际上，“网页”和“HTML 页面”这两个术语是可以通用的。“网站”是创建在一起的、内容上相关的一个或多个网页，就像一本书的书页。“主页”通常指访问网站时看到的第一个网页，不过也有人将“主页”用于任意网页。“主页”的另一个用法是指个人网页，而不是企业网站。

问：我看了 Internet 上一些网页的 HTML 源文件，它看上去很难学。要学习这一语言，是否必须像计算机编程人员那样思考？

答：虽然复杂的 HTML 页面看上去真的会让人望而却步，但学习 HTML 比学习其他计算机语言如 BASIC、C++ 和 Java 等要容易好几个数量级。要成为成功的 HTML 制作人员，并不需要有任何计算机编程人员的经验或技能。许多商业网页的 HTML 代码看起来很复杂是因为它们可能是由可视化 Web 设计工具创建的，而不是手工编码；可视化工具会使代码难于阅读！还要记住一点，有些大型网页文件看起来就很复杂，也可能是长期修改和改进的自然结果，每个网页都是每次创建一点的。

问：创建 HTML 页面时是否需要一直与 Internet 保持连接？

答：不需要。实际上，如果只要将网页发布到光盘、软盘或局域网上的话，根本不需要连接 Internet。如果还没有准备好要发布到 Internet 的话，在开发网页的过程中也不需要 Internet 连接。实际上，我花几个月脱机在硬盘上创建整个网站，然后将它们放到 Internet 上。在第 2 章将介绍更多关于脱机创建网页的细节。

## 1.8 作业

作业包含一些测试题和练习，帮助巩固对这一章内介绍的内容的理解。试着回答所有问题然后再看后面的“答案”小节。

### 1.8.1 测验

1. 给出 Internet、网页和万维网的定义。
2. 要在计算机上创建一个包含一些文本和两个图像的网页，需要保存多少个文件？



3. 能否用 Microsoft Word 或类似的字处理软件来创建网页？

### 1.8.2 答案

1. Internet 是“网络的网络”，它将全球不计其数的计算机连接起来。

网页是文本文档，它使用以 HTML 这一特殊语言编写的命令来添加格式信息、图像和其他媒体信息，并可与其他网页链接。

万维网是 Internet 上所有网页的统称。

2. 至少 3 个文件：一个用于保存网页本身，包含文本和 HTML 命令，每个图像各用一个文件。在有些情况下，可能需要更多文件来存储背景图案、声音或网页的可交互特性。

3. 可以。实际上，可用任何计算机上可保存纯文本文件或 ASCII 文件的任何应用程序来创建网页。只要记住，字处理软件通常默认地使用一些额外的格式信息。使用纯文本编辑器如 Windows 记事本或 Macintosh 的文本编辑器会更简单些。

## 1.9 练习

在本书每章结束后，都有一些可选的练习，帮助巩固和扩展本章所学知识。然而，读者可能急于开始学习 HTML，因此跳过第 1 章的热身运动，直接进入第 2 章的学习。



## 第 2 章

# 开始创建网页

这一章将教你创建第一个网页。学习这一章的最好方法是，一边阅读一边动手创建网页，模仿这一章所示的网页创建自己的网页。如果不太有把握，可以先浏览一遍本章的内容，然后在创建自己的网页时再回过头来重新阅读一遍。不过建议你不必担心，直接开始吧！

第 1 章中提到，可以使用任何文本编辑器或字处理软件来创建 HTML 网页。虽然最后你可能想使用专门的 HTML 编辑器，但在本章中推荐你使用 Windows 记事本或 Macintosh 的文本编辑器，这样就不用在学习 HTML 的同时学习一种新软件。

### 练习

在开始学习本章的课程前，应准备一些要放在网页中的文字：

1. 找出（或编写）关于自己、家庭、公司、垒球队或其他你感兴趣的主题的几段文字。
2. 将这些文字保存在一个纯文本的、标准的 ASCII 文本文件中。记事本和大多数简单文本编辑器总是将文件保存为纯文本文件，但如果使用其他软件则可能需要选择文件类型（选择菜单“文件/另存为”后）。
3. 学习本章课程时，将添加 HTML 代码（称为标签）到文本文件中，使它成为一个网页。使用记事本或其他简单文本编辑器来做，不要使用 Word 或写字板！
4. 在保存包含 HTML 标签的文件时，将这些文件加上扩展名.html。这很重要：如果保存文件时忘记在文件名后输入.html，大多数文本编辑器将赋予文件其他的扩展名（如.txt或.doc）。如果是这样，在网页浏览器中查看这些文件时可能就无法找到。换句话说，网页浏览器要求网页文件的扩展名为.html；你也会见到文件扩展为.htm 的网页，这也是可以接受的。Web 上还有其他的一些文件扩展名，如.asp（Microsoft Active Server Pages）和.php（PHP：Hypertext Preprocess），这些扩展名通常是与超出 HTML 范围外的技术相关的。

**注意：**如果使用 Macintosh 计算机上的文本编辑器，创建网页的步骤与使用 Windows 记事本稍微有些不同。首先必须在“格式”菜单中选择“使用纯文本”，然后在“保存”标签下不选中“为纯文本文件附加.txt 扩展名”复选框。另外，默认设置将在浏览器中显示.html 文档，这样就不能对文档进行编辑。要修改这一点，在“格式化文件处理”选项卡中选择“忽略 HTML 文件的格式化文本命令”。

By the  
Way



## 2.1 从简单网页开始

程序清单 2.1 列出了要创建简单的 HTML 页面需要输入和保存的文本。如果在网页浏览器如 Internet Explorer 中打开这个文件，将看到如图 2.1 所示的页面。创建的每个网页都必须包含<html>、<head>、<title>和<body>等标签。

程序清单 2.1 标签<html>、<head>、<title>和<body>

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The First Web Page</title>
  </head>

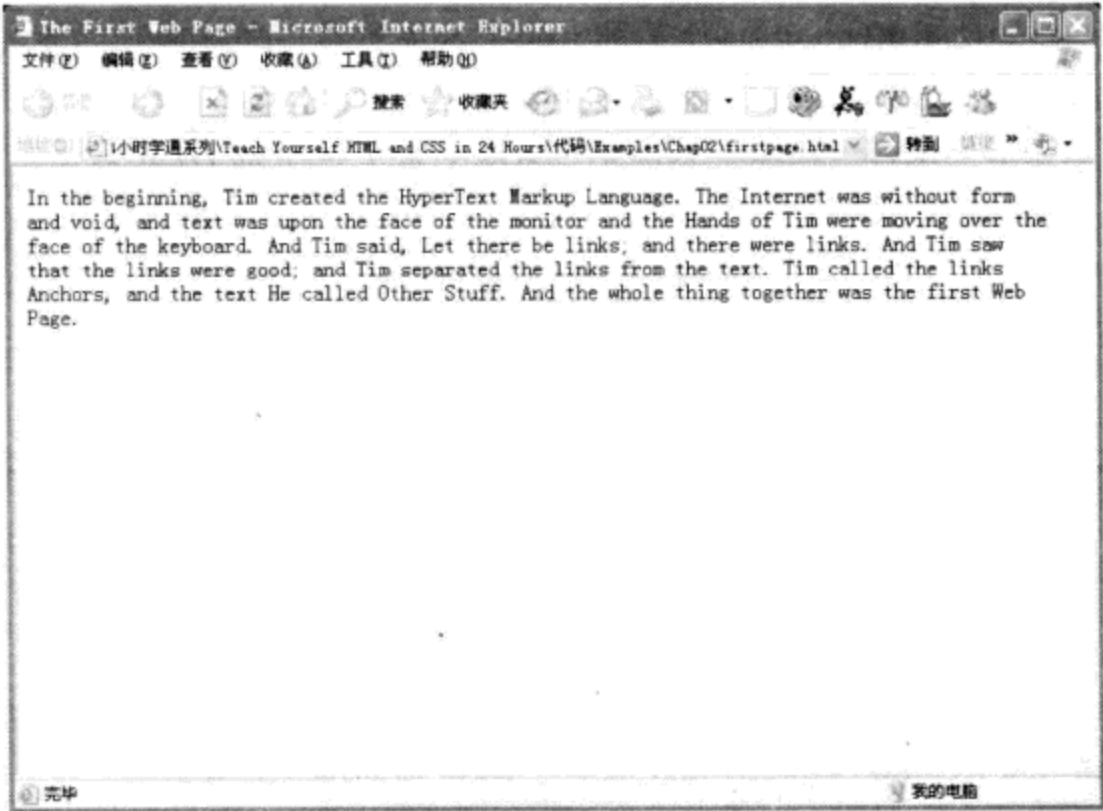
  <body>
    <p>
      In the beginning, Tim created the HyperText Markup Language. The Internet
      was without form and void, and text was upon the face of the monitor and
      the Hands of Tim were moving over the face of the keyboard. And Tim said,
      Let there be links; and there were links. And Tim saw that the links were
      good; and Tim separated the links from the text. Tim called the links
      Anchors, and the text He called Other Stuff. And the whole thing together
      was the first Web Page.
    </p>
  </body>
</html>
```

在程序清单 2.1 中，与所有 HTML 页面一样，以“<”开始、以“>”结束的单词实际上都是编码命令。这些编码命令成为 HTML 标签，因为它们“标记”了一些文本，使网页浏览器能够识别这是哪种文本，这样浏览器就能够正确地显示文本。

HTML 标签是用于指示如何显示网页部分的编码命令。

图 2.1

使用网页浏览器浏览程序清单 2.1 所示的网页时，只能看到实际的标题和正文文本



网页代码中的前几行是标准的样本代码，所有页面都将包含这些内容。这些代码实际上将页面标识为 XHTML 1.1 文档，也就是从技术上说这个网页是一个 XHTML 页面。本书的所有页面都是 XHTML 1.1 页面。因为 XHTML 是 HTML 的一个更结构化的版本，因此本书所有页面也可以称为 HTML 页面。在代码中使用 XHTML 1.1，开发出来的网页将遵守最新的 Web 标准。

开始学习程序清单 2.1 中的 HTML 标签的含义之前，可能希望看看要创建的是什么并看看文档本身。

1. 在 Windows 记事本（或 Macintosh 的文本编辑器）中输入程序清单 2.1 中的所有文本，包括 HTML 标签。
2. 选择“文件/另存为...”，选择“文本文档”作为文件类型。
3. 将文件命名为 firstpage.html。
4. 在硬盘上选择一个要保存网页的文件夹——记住选择的文件夹！单击“保存”或“确定”按钮保存文件。
5. 现在启动最喜欢的网页浏览器（同时让记事本继续运行，这样便于在浏览网页和编辑代码之间切换）。

By the Way

**注意：**查看保存在本地计算机中的网页不需要连接 Internet。在默认情况下，网页浏览器每次启动时都试图自动连接 Internet，这通常是合理的。然而，如果在本地硬盘上（脱机）开发网页时，这样将造成干扰，可能不断地得到无法找到网页的错误消息。如果任何时候都可以通过局域网、电缆 Modem 或 DSL 连接 Web 的话，将不存在这种问题，因为浏览器从来不会脱机。否则，就需要根据使用的浏览器类型采取适当的措施。

➤ 在 Windows 的 Microsoft Internet Explorer 中，选择“工具/Internet 选项”，单击“常规”选项卡，然后在“主页”下单击“使用空白页”。如果是在



Macintosh OS X 下的 Internet Explorer, 查看“浏览器显示”设置下的应用程序首选项, 在“主页”群中单击“不使用任何页”按钮。

- 在 Mozilla Firefox 中, 选择“工具/选项”; 单击“常规”设置页下的“使用空白页”单选按钮。
- 在其他浏览器中应该也有类似的设置来修改默认主页, 使浏览器不会试图访问 Internet。

这里只是教你不要让浏览器每次启动都从 Internet 上获取页面。当然, 当完成网页的创建后, 可能希望恢复原来的默认设置。

6. 在 Internet Explorer 中, 选择“文件/打开”, 然后单击“浏览”。如果使用的是 Firefox, 选择“文件/打开文件”。找到正确的文件夹并选择 firstpage.html 文件。也可以将 firstpage.html 文件拖放到浏览器窗口中。

现在应该看到如图 2.1 所示的页面。

## 2.2 每个网页都必须包含的 HTML 标签

现在是揭下 HTML 标签的神秘面纱的时候了。理解这种语言后, 将具有超出他人的创造力。不要告诉其他人, 但它真的很容易。

开始学习 HTML 标签前, 先来解释程序清单 2.1 中看起来有点混乱的代码。第一行代码表明该 HTML 文档实际上是一个 XML 文档。

```
<?xml version="1.0" encoding="UTF-8"?>
```

XML 版本设置为 1.0, 这是相当标准的, 另外还指定了字符编码类型 (UTF-8)。

**注意:** 现在理解字符编码对你来说并不是很重要。重要的是必须在网页中包含适当的样本代码, 使它们遵守最新的 Web 标准。

**By the Way**

程序清单 2.1 的第二行和第三行看起来更复杂:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

同样, 这段代码的意义并不很重要, 只要记住在网页开头包含这些代码。这段代码指出该文档是 XHTML 1.1 文档, 这样允许网页浏览器检查文档, 保证代码满足 XHTML 1.1 的所有要求。

**注意:** XML/XHTML 样本代码在创建网页时并不是一定要包含。在这个示例中, 可以删除开头的几行代码, 使页面以 <html> 标签开始, 这样也能在网页浏览器中打开这个文件。包含这些额外的代码的原因是, 为保证网页在 Web 发展时能够移植到改进后的新 Web 上。另外, 这些额外代码有助于验证网页的准确性——在本章稍后将学到如何实现这一点。

**By the Way**

大多数 HTML 标签都由两部分组成: 开始标签, 表明本标签由文本的哪个地方开始; 结

束标签，表明标签在哪个地方结束。结束标签以“/”开始，位于“<”符号后。还有一种标签是空标签，其独特之处在于不需要一对开始标签和结束标签。为保证读者理解每种标签的作用，下面列出了对这三种标签的总结。

- 开始标签是指明 HTML 命令开始的 HTML 标签；命令影响的文本位于开始标签后。开始标签总是以“<”开始、以“>”结束，如<html>。
- 结束标签是指明 HTML 命令结束的 HTML 标签；命令影响的文本在结束标签之前。结束标签总是以“</”开始、以“>”结束，如</html>。
- 空标签是在页面中发出一个 HTML 命令而不包含任何文本的 HTML 标签。空标签总是以“<”开始、以“/>”结束，如<br/>。

例如，在程序清单 2.1 中的<body>标签告诉网页浏览器真正的页面正文文本从哪里开始，而</body>指明在哪里结束。位于<body>和</body>这两个标签之间的内容将显示在网页浏览器窗口中主要的显示区域，如图 2.1 所示。

网页浏览器将在浏览器窗口的顶部显示<title>和</title>之间的文本，如图 2.1 所示。将网页加到浏览器的书签或收藏夹中时，标题文本也用于标识网页。为网页提供标题很重要，这样便于访问网页的人将网页加到书签中供以后使用。

在每个 HTML 页面中都将使用<body>和<title>标签，因为每个网页都需要标题和一些正文。还将使用程序清单 2.1 所示的其他两个标签：<html>和<head>。将<html>放在文档的开头，表明这是一个网页；将</html>放在最后，表明网页已结束。

在页面中有一个标题部分和正文部分，分别由标签<head>和<body>标识。页面标题部分中的消息对页面进行描述，但并不显示在网页浏览器中。放在正文中的信息才被网页浏览器显示。<head>标签的位置总是靠近页面的 HTML 代码的开头，在<html>标签之后。

### By the Way

**注意：**你肯定注意到了，在上面的示例中还有一些代码与<html>相关联。这些代码由两个属性组成（xmlns 和 xml:lang），用于指定与该标签相关的其他信息。这两个属性是所有 XHTML 网页的标准要求。

用于指定页面标题的<title>标签显示在页面的标题部分，也就是说它位于开始标签<head>之后、结束标签</head>之前。接下来将介绍其他一些可放在<head>和</head>之间的高级标题信息，如用于格式化网页的样式表（style sheet）规则。

### Did you know?

**提示：**创建并保存与程序清单 2.1 所示文档类似，但只包含<html>、<head>、<title>和<body>的框架页面很方便。这样，需要创建新网页时，可以打开该文档开始编辑，从而省去每次都要输入必需标签的麻烦。

程序清单 2.1 中的<p>标签用于包含一段文本。在这个示例中，这个标签并不是必需的，因为只有一个段落。但如果网页有多个段落，这个标签就很重要。

## 2.3 使用分段和分行来组织页面

网页浏览器显示 HTML 页面时，并不会考虑行尾和文本间的空格。例如，在图 2.2 中开

头的诗中，每个单词间都只有一个空格，而不管它在程序清单 2.2 是如何输入的。这是因为在 HTML 代码中多余的空格都自动地减为一个。另外，当文本到了浏览器窗口的边缘时，它自动地滚动到下一行，而不管原来的 HTML 文件中换行在哪里；在这个示例中，正好所有文本都在一行中。



图 2.2  
浏览 HTML 代码如程序清单 2.2 所示的页面时，只有包含<br />和<p>标签的地方才出现分行和分段

程序清单 2.2 图 2.2 所示页面的 HTML 代码

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Advertising Agency Song</title>
  </head>

  <body>
    <p>
      When your client's hopping mad,
      put his picture in the ad.

      If he still should prove refractory,
      add a picture of his factory.
    </p>

    <hr />

    <p>
      When your client's hopping mad,<br />
```

```

        put his picture in the ad.
    </p>
    <p>
        If he still should prove refractory,<br />
        add a picture of his factory.
    </p>
</body>
</html>

```

如果要控制分行和分段的位置，必须使用 HTML 标签。要使段落间隔一行，可在每段的开始加上<p>标签，在每段结尾加上</p>标签。也就是说，将每段的文本放在一对<p>与</p>标签之间。

<br/>标签在段落内强行分行。与前面介绍的大多数标签不同，<br/>并不要求结束标签</br>——这是之前提到过的空标签示例之一。这也是 XHTML 出现在网页中的示例之一，因为普通的 HTML 空标签不要求“/”符号，但新的 XHTML 标准要求。因此，应遵守最新的版本，创建编码正确的网页——总是在空标签中包含“/”。

### By the Way

**注意：**现在在 Internet 上看到的大多数网页都使用<br>而不是<br/>，目前大多数网页浏览器都把它们当成一样的。然而，如果现在就养成使用<br/>这一新格式，以后就可以省下许多修改工作。

同样，结束标签</p>在 HTML 4 中是可选的，现在的网页开发人员通常也不使用它。新的 XHTML 标准要求使用结束标签</p>，因此推荐你总是使用它们。养成编写清晰 HTML 代码的习惯，对于成为成功的网页设计人员来说很重要。

程序清单 2.2 和图 2.2 中的诗显示了标签<br/>和<p>用于分开押韵的广告代理歌的行和段。你可能还注意到了清单中的<hr/>标签，它使页面包含一条水平的标尺线（见图 2.2）。使用<hr/>标签插入水平标尺线也会引起分行，即使并没有使用<br/>标签。要使水平标尺线上下有更多的空间，可以在<hr/>标签前后分别添加标签<p>和</p>。这相当于将水平标尺线单独放在一个段落中。

与<br />一样，水平标尺线标签<hr />也是一个空标签，因此不需要结束标签</hr>。

使用一段文本，将它格式化为适当的 HTML。

1. 将<html><head><title>My Title</title></head><body>添加到文本的开头（使用自己的网页标题代替“My Title”。同时在页面标题中包含满足 XHTML 要求的样本代码）。
2. 在文本的末尾添加</body></html>。
3. 在每段的开头添加<p>标签，结尾添加</p>标签。
4. 在需要分行的地方使用<br />标签。
5. 使用<hr />标签绘制水平标尺线，将文本的主要部分分开，或在页面需要线条的地方使用。



6. 将文件保存为 `mypage.html` (使用自己的文件名代替 “`mypage`”)。

**注意：**如果使用字处理软件来创建网页，要确保将 HTML 文件保存为纯文本格式或 ASCII 格式。

**By the way**

7. 在网页浏览器中打开文件来浏览网页。

8. 如果什么地方出错，回到文本编辑器中进行修改，然后再次保存文件。单击浏览器中的“刷新”可以看到对网页的修改结果。



## 2.4 使用标题突出文本

在 Internet 上浏览网页时，将注意到许多网页开头都有一个字体比其他文本更大更粗的标题。程序清单 2.3 是一个简单的示例，这个网页包含了一个标题以及一些正常的段落文本作为比较。在标签 `<h1>` 和 `</h1>` 之间的文本将作为大标题文本显示。另外，`<h2>` 和 `<h3>` 表示字体较小的标题，依次类推，还有字体更小的标题标签。

程序清单 2.3 标题标签

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Music City Mafia - Terry Lancaster</title>
  </head>
  <body>
    <h1>16 - Terry Lancaster</h1>
    <p>
      <br />
      Nickname: Big T<br />
      Position: RW<br />
      Height: 6'3"<br />
      Weight: 195<br />
      Shoots: Left<br />
      Age: 40<br />
      Birthplace: Nashville, TN
    </p>
    <hr />
    <p>
      Favorite NHL Player: Brett Hull<br />
      Favorite NHL Team: Nashville Predators<br />
    </p>
  </body>
</html>
```

```
Favorite Southern Fixin: Skillet Fried Potatoes<br />
Favorite Meat and Three: Swett's<br />
Favorite Country Star: Patsy Cline<br />
Favorite Mafia Moment: "Chet finishing a game with his eyelid completely
slashed through."
</p>
</body>
</html>
```

By the Way

**注意：**继续阅读本书时，读者将创建一个更大的网页，程序清单 2.3 中的网页只是这个网页的一部分。我认为实际的示例更能使你感受到 HTML 的实用价值，因此我在本书中以一个业余曲棍球队的网站为例。实际上，它正是我参加的曲棍球队 Music City Mafia。我的曲棍球伙伴很慷慨地允许我在本书中使用他们那些难看的鬼脸作为示例。

By the Way

**注意：**现在读者可能注意到了，HTML 代码被缩进以表示 HTML 文档中不同部分之间的关系。这种缩进完全是可选的，可以将所有标签都放在一起，之间没有空格或分行，网页在浏览器中仍可正常显示。这里的缩进旨在让读者很快理解充满代码的网页以及它们之间的关系。代码缩进是一种良好的 Web 开发习惯，最终使网页更易于维护。

在图 2.3 中可以看到，创建标题的 HTML 简单得不能再简单。在这个示例中，曲棍球队员的球衣号码和姓名使用<h1>标签突显在页面中。要创建一级标题，只要在文本前面使用<h1>标签，在文本后使用</h1>标签。至于较小的二级标题，使用<h2>和</h2>，三级标题使用<h3>和</h3>。

图 2.3

程序清单 2.3 中的<h1>标签使曲棍球队员的球衣号码和姓名比其他信息更突出



**注意：**这个简单的示例中包含了一个图像，可能有点操之过急了，然而，有时候为了达到想要的结果，在可承受的范围之内再进一步是值得的。你将在第 7 章和第 8 章学习创建图像以及在网页中使用图像，现在需要知道的是标签 `<img>` 负责将图像放在网页上。

**By the  
Way**

**休息一下：**既然在讨论曲棍球和 HTML，现在就去 <http://www.azhockey.com/unusual.html> 看一些关于曲棍球运动的具有娱乐性的、不寻常的故事。这个网页同时也很好地说明了简单的网页也能够很好地将信息呈现给访问者。我并不是说这个网页应得一个设计奖，但它确实是 Web 简单性的一个很好的示例，这可能成为 Web 发展中令人耳目一新的改变。

**Coffee  
Break**

从理论上说，也可以使用 `<h4>`、`<h5>` 和 `<h6>` 来表示比较不重要的标题，但通常没有人使用这些标签——毕竟，如果标题不是大字体或粗体，那标题还有什么意义呢？此外，大多数网页浏览器对这些标题的显示与小的 `<h3>` 标题的区别几乎看不出来。至于图 2.3 所示的示例，只需使用 `<h1>` 标签和其他普通的段落文本就足够了。

**注意：**在现在的许多网页中，通常用包含艺术字体和图标的图形图像来取代本章讨论的普通标题文本。你将在第二部分中学习如何创建图像并将图像放在网页上。然而，文本标题仍被广泛使用，它有一个优点，就是传输和显示的速度快，不管读者与 Internet 的连接速度是快是慢。

**By the  
Way**

要记住 `title` 与 `heading` 之间的区别。这两个单词在英语中是可以通用的，但在 HTML 中，`<title>` 指定整个网页的标题，但并不显示在网页上，而是显示在浏览器窗口的标题栏中。而 `heading` 标签使一些文本突显在网页上。一个网页只有一个 `<title>`，且只能出现在 `<head>` 和 `</head>` 标签之间，而 `<h1>`、`<h2>` 和 `<h3>` 可以有任何多个，顺序也可以根据需要任意安排。另外，与 `title` 不同，`heading` 总是放在页面中。

**注意：**记住，放在网页标题中的信息是不准备显示在页面中的，而页面主体中的内容是要供人浏览的。

**By the  
Way**

在第二部分，你将学习对网页中文本的外观进行全面控制。由于不必对标题进行字体大小、字体和字体颜色的设置，因此标题是突出重要文本最简单、最流行的方法。

## 2.5 学习他人的网页

看到许多流行的网页所呈现的视觉和声音效果，你可能认识到这一章介绍的简单网页只是 HTML 的冰山一角。由于你已经掌握了基础知识，看看 Internet 上别人的网页可能让你有意想不到的收获。正如在第 1 章提到的，只要在网页浏览器中右击然后选择“查看源文件”，就能看到任何网页的 HTML 源代码。

如果还看不懂有些 HTML 标签或不知道如何使用，不要担心。在下面几章将学到这些东西。然而，现在提前看看这些标签，将能够知道已掌握的知识，了解将来能够在网页中实现

的功能。

**By the Way** **注意：**Sams Publishing 网站上的 HTML 是特别设计的，很直观，容易理解。

## 2.6 验证网页

下面打破常规，推荐一件现在几乎没有 Web 设计人员愿意做的事情：对所有网页进行验证。很少有 Web 设计人员愿意验证，因为我们才刚刚从包含混杂代码的 Web 形态转入改造为更结构化的、包含一致化页面的 Web 形态。设计和绘制漂亮的房屋蓝图是一回事，建筑师设计出安全的、适于建造的结构则是另一回事。网页验证是类似的过程，只是这种情况下建筑师是应用程序而不是人。

验证是指在特定应用程序中运行网页、查找错误并保证网页遵循严格的 XHTML 标准的过程。

网页验证好的一点是它非常容易。实际上，负责开发 Web 标准的标准机构——万维网联盟（World Wide Web Consortium，简称 W3C）——提供了在线验证工具。访问下面的网址：<http://validator.w3.org/>，并试着进行网页验证。W3C 的标记验证服务如图 2.4 所示。

图 2.4  
W3C 的标记验证服务让你能够验证 HTML(或 XHTML)文档，保证文档编码正确



如果已经在网上发布网页，可以使用“Validate by URL”选项进行验证；否则，使用“Validate by File Upload”选项，它让你能够验证存储在本地计算机文件系统中的文件。单击“浏览”按钮，浏览找到要验证的文件，然后单击“Check”按钮。如果一切顺利的话，网页将会得到如图 2.5 所示的通过报告。

如果 W3C 的标记验证服务在网页上发现错误，它将提供一些具体的细节，包括出错代码的行号。这不失为跟踪问题、调试网页的好办法。因此，文档验证不仅让你能够知道网页是否正确，还能帮助你在网页发布之前发现问题、解决问题。



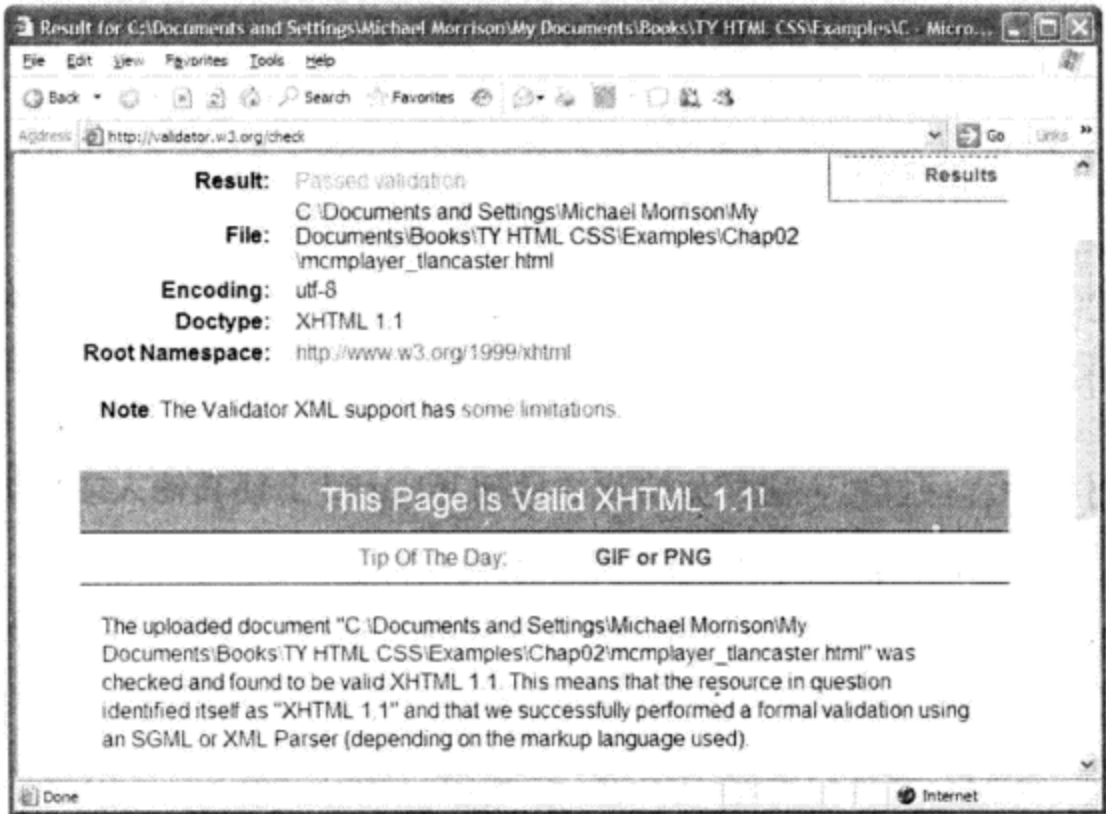


图 2.5  
如果网页通过了 W3C 的标记验证服务，就表明这个网页已经准备好了

**提示：**有些 Web 开发工具包含内置的验证功能，可以代替 W3C 的标记验证服务。

**Did you know?**

## 2.7 总结

这一章介绍了最基本、最重要的 HTML 标签。将这些编码命令添加到纯文本文档中，可以将文件转换为真正的网页。

创建网页的第一步是在文档开头和结尾加入一些必要的标签，包括页面的标题。接下来可以根据需要标记行和段落结束的位置，或添加水平标尺线和文本标题。表 2.1 总结了本章介绍的所有标签。

表 2.1                      第 2 章介绍的标签

标签	功能
<html>...</html>	包含整个 HTML 文档。
<head>...</head>	包含 HTML 文档的标题。
<title>...</title>	指明文档的标题，用于<head>内。
<body>...</body>	包含 HTML 文档的正文。
<p>...</p>	包含一个段落；段落间隔一行。
 	分行。
<hr />	水平标尺线。
<h1>...</h1>	一级标题。
<h2>...</h2>	二级标题。
<h3>...</h3>	三级标题。
<h4>...</h4>	四级标题（很少使用）。
<h5>...</h5>	五级标题（很少使用）。
<h6>...</h6>	六级标题（很少使用）。

## 2.8 问与答

**问：**我创建了一个网页，但在网页浏览器中打开它时，看到了所有文本，包括 HTML 标签。有时候甚至看到一些奇怪的字符出现在页面开头！什么地方做错了？

**答：**可能没有将文件保存为纯文本文件。试着重新保存文件，一定要将它保存为纯文本文件或 ASCII 文件。如果不太清楚如何使用字处理器将文件保存为纯文本文件，不要勉强。使用记事本或文本编辑器。另外，还要保证网页的文件名以 .html 或 .htm 结束。

**问：**将 HTML 网页保存在计算机中后，如何将它放到 Internet 上使其他人能够访问？

**答：**第 4 章将解释如何将网页放到 Internet、局域网和光盘上。

**问：**我想让“Fred's Fresh Fish”显示在页面开头，同时让网页被加入书签中时该内容显示在书签中。如何在这两个地方显示？

**答：**指定页面标题，并在页面开头添加一个文本标题，如下所示：

```
<html>
  <head>
    <title>Fred's Fresh Fish</title>
  </head>

  <body>
    <h1>Fred's Fresh Fish</h1>
    ...the rest of the page goes here...
  </body>
</html>
```

**问：**Internet 上的有些网页开头并没有使用<html>标签，而你说过网页总是以<html>开始。这是怎么回事？

**答：**如果网页中没有<html>标签，许多网页浏览器也会正常地显示。但在网页中包含这个标签是一个好习惯，因为有些软件需要这个标签来识别有效的 HTML 页面。另外，网页也应该是真正的 XHTML 页面才能符合最新的 Web 标准。

## 2.9 作业

作业包含一些测试题和练习，帮助巩固对这一章介绍的内容的理解。试着回答所有问题，然后再看后面的“答案”小节。

### 2.9.1 测验

1. 所有 HTML 页面都必须包含哪四个标签？
2. 在下面的文本中插入适当的分行和分段标签，使之成为两行诗，中间有一空行：

Good night, God bless you,

Go to bed and undress you.  
 Good night, sweet repose,  
 Half the bed and all the clothes.

3. 编写 HTML，使下列内容按顺序出现在网页中：

- 包含 “We are Proud to Present” 的小标题；
- 横穿页面的水平标尺线；
- 包含单词 “Orbit” 的大标题；
- 包含 “The Geometric Juggler” 的中等标题；
- 另一条水平标尺线。

4. 编写一个完整的网页，标题为 “Foo Bar”，页面开头有一个 “Happy Hour at the Foo Bar” 标题，下面是正常字体的 “Come on down!”。

## 2.9.2 答案

1. `<html>`、`<head>`、`<title>`和`<body>`（及其对应的结束标签`</html>`、`</head>`、`</title>`和`</body>`）。

2.

```
<p>Good night, God bless you,<br />
Go to bed and undress you.</p>
<p>Good night, sweet repose,<br />
Half the bed and all the clothes.</p>
```

3.

```
<h3>We are Proud to Present</h3>
<hr />
<h1>Orbit</h1>
<h2>The Geometric Juggler</h2>
<hr />
```

4.

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Foo Bar</title>
  </head>
  <body>
```

```
<h1>Happy Hour at the Foo Bar</h1>  
Come on Down!  
</body>  
</html>
```

## 2.10 练习

1. 即使阅读本书旨在创建商业网页，可能也希望创建个人网页作为练习。输入几段自我介绍，并使用本章学到的 HTML 标签使它成为一个网页。
2. 本章介绍的 HTML 标签很常用，因此你可能想记住这些标签。记住这些标签的最好方法是花些时间创建几个网页。可以试着创建一些基本网页，包含要发布到 Internet 上的信息；也可以发挥想象力，创建一些有趣的网页。不要忘记每次修改后都要进行测试。





## 第 3 章

# 链接到其他网页

在前面两章，读者学习了如何使用 HTML 标签来创建包含文本的网页。然而，这时候的网页还是个孤岛，没有与其他信息连接。要使其成为“真正”的网页，需要将它与万维网连接——或者至少要与自己的个人网页或公司网页连接。

本章介绍如何创建超文本链接。读者将学习如何创建到第 6 章中同一网页的另一部分的链接。

超文本链接是当浏览者单击鼠标时，将浏览者从一个网页导航到另一个网页的文本。

虽然在本章学习的 HTML 标签也可使图形图像成为可单击的链接，但这里不专门介绍图形链接。读者将在第 8 章学习有关图形链接的内容。现在将精力集中在通过文本而不是图像链接到其他网页。

### 3.1 链接到另一个网页

创建链接的标签是[<a>](#)，它表示“anchor”（锚）。虽然“anchor”这个词用于描述链接可能不是很清楚，但可以这样理解，<a>标签用于标识网页中一个特定的点——锚点（anchor point）。当然，肯定有更好的单词能够更好地表达这个意思，但我们已经用惯了“anchor”这个词，因此习惯它吧！要链接的网页的地址放在<a>标签内的“href=”后，如下所示：

```
<a href="http://www.stalefishlabs.com/products.html">click here!</a>
```

该链接以带下划线的蓝色字体显示“Click here!”。用户在这些单词上单击时，位于www.stalefishlabs.com的Web服务器中的products.html网页将显示在网页浏览器中，就像用户手工输入了地址一样。顺便说一下，Internet地址也称为统一资源定位符（Uniform Resource Locators，URL）。

回到<a>标签，href表示超文本引用（hypertext reference），它是<a>标签的一个属性。你将在第 5 章学到更多关于属性的知识。

属性是与标签相关的额外信息，提供关于标签的更多细节。例如，<a>标签的 href 属性指定要链接的网页地址。

### By the Way

**注意：**读者可能知道，在大多数网页浏览器上输入地址时可以省略前面的 http://。然而，在网页上的<a href>链接中输入地址时不能省略。

### Did you Know?

**提示：**通常，地址中可以省略 HTML 页面的名称。Internet 上的大多数计算机都自动地为特定地址或目录文件夹返回主页。例如，可用 http://www.stalefishlabs.com 来引用 http://www.stalefishlabs.com/index.html，因为服务器知道用户应先浏览 index.html（参见第 4 章）。当然，这只能发生在主页存在的情况下；否则，还是要输入完整的网页名称。

程序清单 3.1 包含了一些<a>标签，它们显示了如图 3.1 所示的下划线链接。链接的地址在 href 属性中指定。例如，单击图 3.1 中的“Times Square”，将进入如图 3.2 所示的网页：<http://www.earthcam.com/usa/newyork/timesquare/>。

程序清单 3.1 <a>标签可将网页同有趣的网页连接

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>You Aren' t There</title>
  </head>

  <body>
    <h1>Wonders of the World</h1>
    <p>
      Vacations aren' t cheap. But who needs them anymore, with so many live
      cameras connected to the World Wide Web? Pack a picnic, and you can
      visit just about any attraction you want without ever spending any gas
      money or boarding a plane. Stop off at
      <a href="http://www.earthcam.com/usa/newyork/timesquare/">Times
      Square</a> in New York City to see a live view of the busy spot. Then
      head south to Atlanta, Georgia and watch the
      <a href="http://www.atlantafalcons.com/fans/article.jsp?id=6947">Atlanta
      Falcons training camp</a> live online. When you' re finished watching
      Michael Vick, head out west to one of my favorite destinations and take
      in the incredible red rocks of
      <a href="http://www.earthcam.com/cams/arizona/sedona/">Sedona,
      Arizona</a>.
    </p>
  </body>
</html>
```

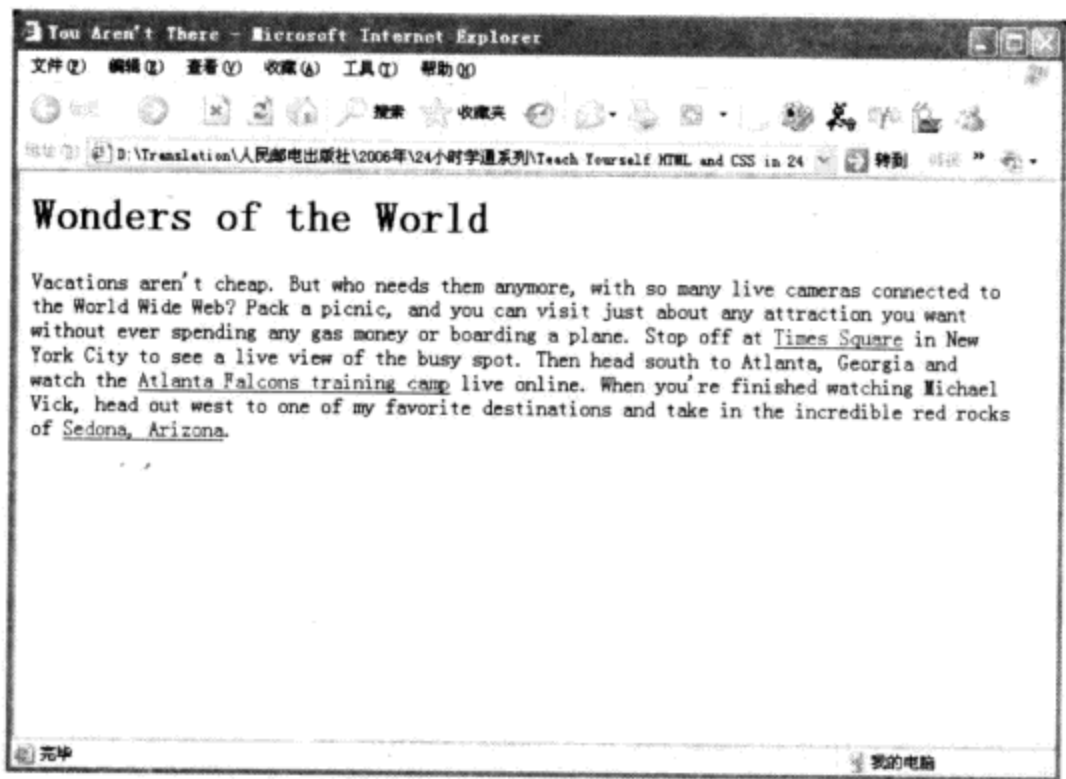


图 3.1

程序清单 3.1 中的 HTML 生成这个网页，链接显示为带下划线的蓝色或紫色文本

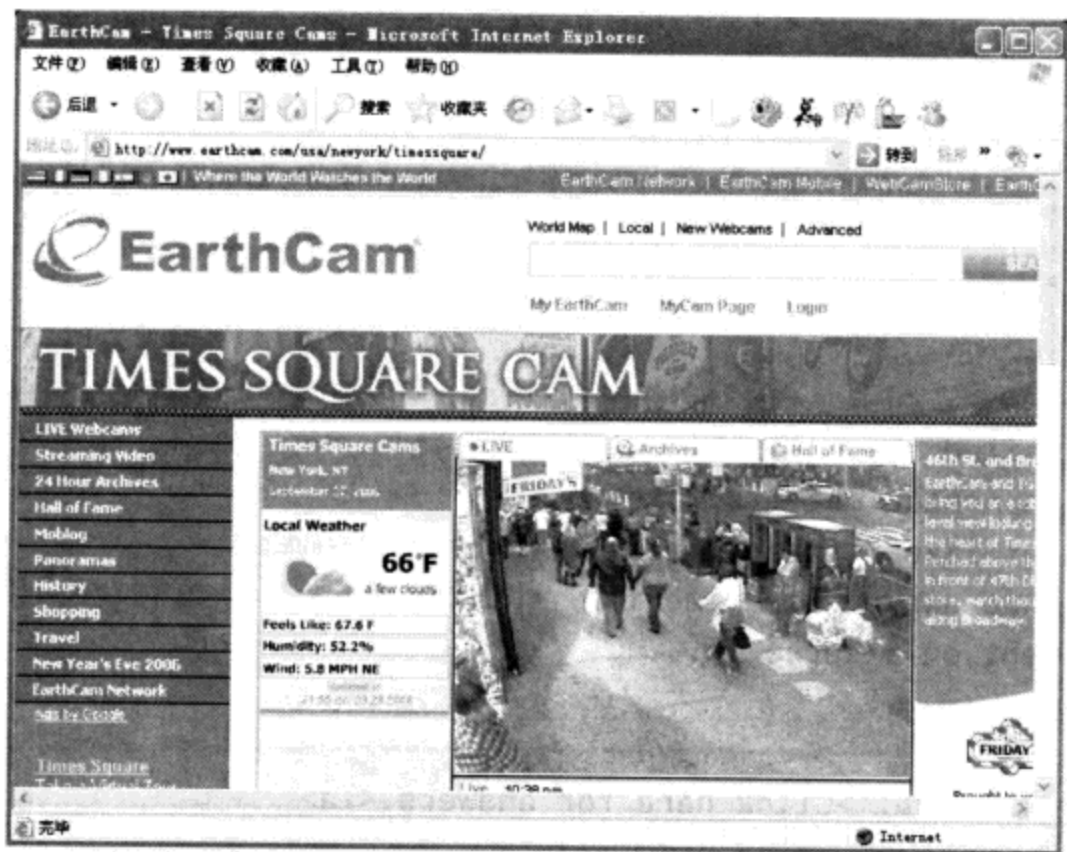


图 3.2

单击图 3.1 中的 Times Square 链接，将进入 Internet 上的 Times Square Earth Cams 页面

**提示：**可以很容易地将网页浏览器中的地址复制到自己的 HTML 页面中。只要在 Web 浏览器中的地址栏、“书签属性”对话框或“编辑收藏夹”对话框将地址选择显示，接着选择“编辑/复制”（或按 Ctrl+C，在 Mac 上是 Command + C），然后在 HTML（文本）编辑器中输入<a href=“，再选择“编辑/粘贴”（CTRL+V 或 Command+V）。

**Did you Know?**

### 3.2 在自己的网页间链接

关于之前提到的一条规则——在 href 属性中指定的每个地址前面都需要包含 http://，有一个例外是，在处理同一计算机或 Web 服务器上的文件时。创建位于同一计算机上的网页之

间的链接时，不一定要指定完整的 Internet 地址。实际上，如果两个页面保存在同一文件夹中，可以只用 HTML 文件本身的文件名：

```
<a href="pagetwo.html">click here to go to page 2.</a>
```

举个例子，程序清单 3.2 和图 3.3 是一个测试页面，包含到程序清单 3.3 和图 3.4 的答案页面的链接。答案页面包含一个链接返回测试页面。由于程序清单 3.2 中的页面链接到同一目录下的其他页面，因此可使用文件名而不必使用完整的地址。

#### 程序清单 3.2 链接到同一目录下的另一个网页

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>History Quiz</title>
  </head>
  <body>
    <h1>History Quiz</h1>
    <p>
      Complete the following rhymes. (Example: William the Conqueror played
      cruel tricks on the Saxons in... ten sixty-six.)
    </p>
    <p>
      1. Columbus sailed the ocean blue in...<br />
      2. The Spanish Armada met its fate in...<br />
      3. London burnt like rotten sticks in...<br />
      4. Tricky Dickie served his time in...<br />
      5. Billy C. went on a spree in...
    </p>
    <p>
      <a href="answers.html">Click here for answers.</a>
    </p>
  </body>
</html>
```

---

#### 程序清单 3.3 这是 answers.html 文件；程序清单 3.2 是 quizzer.html，answers.html 链接到 quizzer.html

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>History Quiz Answers</title>
```



```
</head>

<body>
  <h1>History Quiz Answers</h1>
  <p>
    1. ...fourteen hundred and ninety-two.<br />
    2. ...fifteen hundred and eighty eight.<br />
    3. ...sixteen hundred and sixty-six.<br />
    4. ...nineteen hundred and sixty-nine.<br />
    5. ...nineteen hundred and ninety-three.
  </p>
  <p>
    <a href="quizzier.html">Click here for the questions.</a>
  </p>
</body>
</html>
```

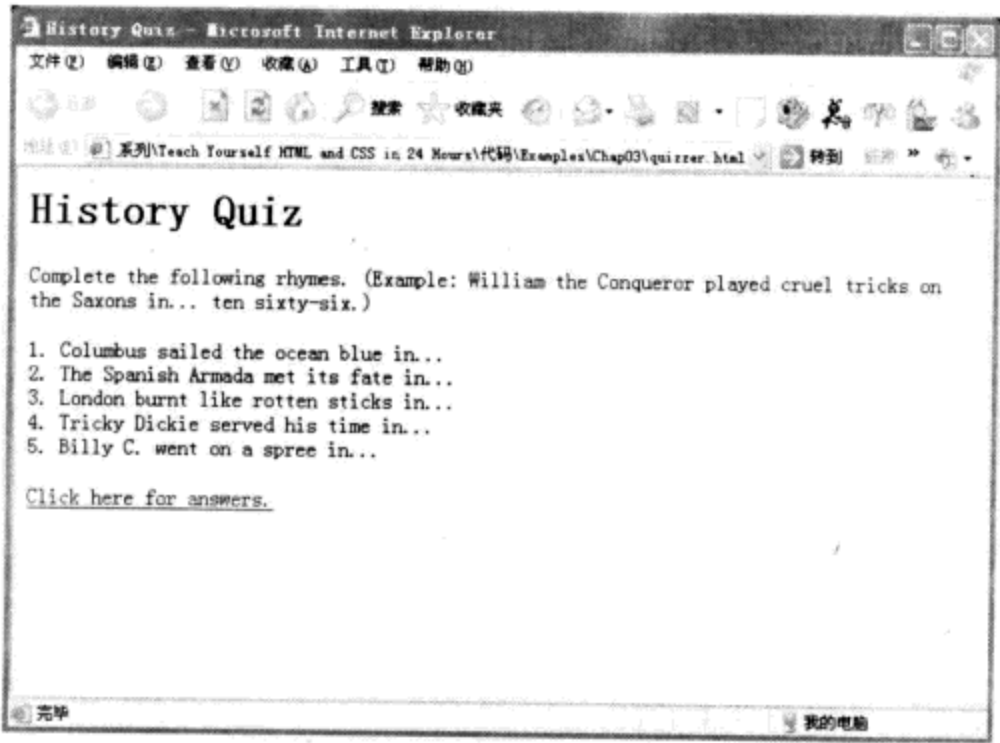


图 3.3  
这是程序清单 3.2 列出的 quizzier.html，程序清单 3.3 通过链接引用该页面

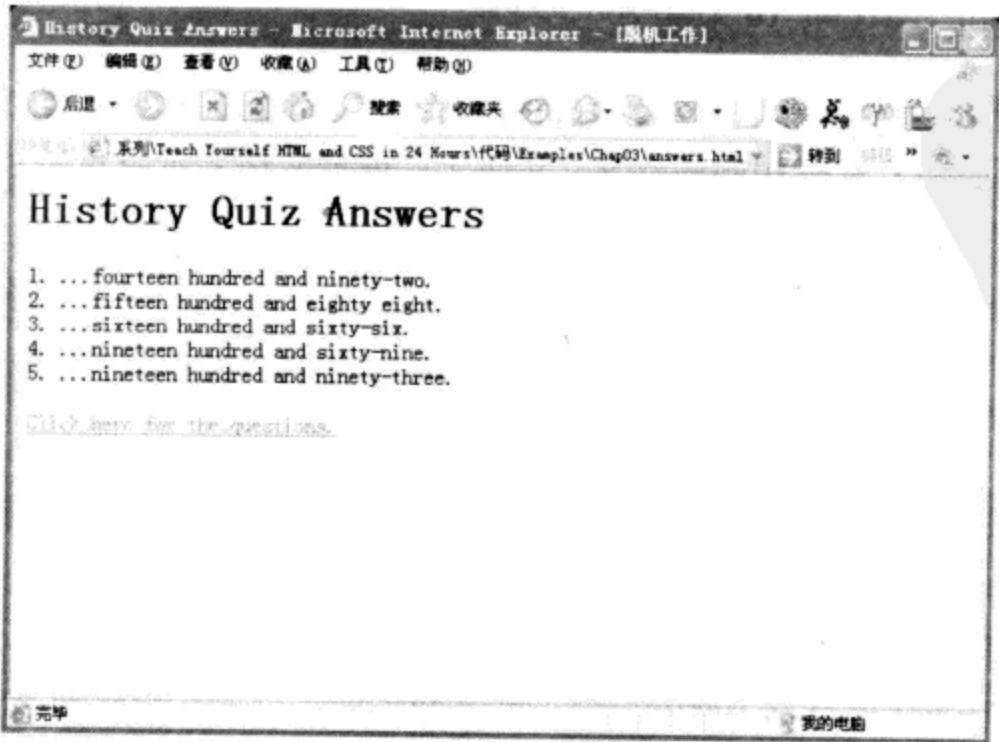


图 3.4  
图 3.3 的 Click here for answers 链接将进入该页面。单击链接 Click here for questions 将返回图 3.3 所示的页面

使用文件名而不是完整的 Internet 地址可以节省很多打字时间。更重要的是，网页之间的链接不管整个页面集合如何存储都可以正常工作。文件还保存在计算机硬盘上时，可以测试这些链接。当这些文件存储到 Internet、CD-ROM、DVD 或存储卡上时，所有链接仍能正常工作。这种识别网页的简单方法并没有什么神奇的地方——它仍与网页地址有关，下面你将了解到这一点。

### By the Way

**注意：**在 Sams Publishing 网站上，可以找到许多演示超文本链接的有趣示例网页，包括 Internet 上的 Indigestible Ingestibles Research 网站之旅以及轻松的文学故事测验。

## 3.3 指定网页的地址

为组织网站的网页，最简单方法是将所有网页放在同一文件夹下。当网页以这种方式存储时，可以只在<a>标签的 href 属性中指定文件名来链接网页。如果有许多网页，可能要将这些网页放在多个文件夹下，以便更好地组织。在这种情况下，链接网页时仍不应使用完整的 Internet 地址。可以使用相对地址，相对地址只包含要找到另一个网页的足够信息。

相对地址说明了一个网页到另一个网页的路径，而不是完整（或绝对）的 Internet 地址。

例如，假设要在硬盘的 Webpages 目录下创建一个 zoo.html 页面。这个文件包含一个链接，该链接指向 african.html 页面，它位于 Webpages 下的 elephants 子文件夹中。这个链接如下所示：

```
<a href="elephants/african.html">learn about african elephants.</a>
```

### Did you Know?

**提示：**HTML 总是使用斜杠 “/” 来分隔目录。不要使用 Windows 或 DOS 使用的反斜杠 “\”！你可能对提到 DOS 感到恐怖，但它确实有一些东西遗留在 Windows 操作系统中，如反斜杠。

african.html 页面可能包含一个链接返回到 zoo.html 页：

```
<a href="../zoo.html">return to the zoo.</a>
```

双点号 (“..”) 是一个特殊符号，表示包含当前文件夹的文件夹——也就是父文件夹。“..”在 Windows、Macintosh 和 UNIX 都有相同的含义。实际上，指定文件名本身也是使用相对地址的一种形式，因为它表明文件位于当前文件夹下。

如果在网页中统一使用相对地址，将页面移到另一个文件夹、硬盘或 Web 服务器时，就不必修改链接。在这个例子中，只要 african.html 页面位于 elephants 子文件夹下，链接就可以正常使用。

### Did you Know?

**提示：**关于使用相对地址（elephants/african.html）还是绝对地址（http://www.takeme2thetoo.com/elephants/african.html），有一个通用规则，那就是：在链接存储在一起的文件时应用相对地址；链接到其他地方——其他计算机、其他硬盘或 Internet 的其他网站上的文件时，应使用绝对地址。

必要时相对地址也可以用于表示复杂的目录结构；第 22 章提供了关于组织和链接大量网页的更多建议。

### 练习

在第 2 章中，你可能创建几个网页。现在再添加一些网页，并将它们链接起来。

- 使用一个主页作为进入其他网页的入口和枢纽。如果在第 2 章创建了关于自己或公司的网页，将它作为主页。也可以现在创建一个新网页，将其作为主页。
- 在主页上，添加一些[<a href>](#)链接，指向已创建（或计划创建）的其他 HTML 文件。确保每个链接的文件名都拼写正确，包括大小写。
- 可能还想在主页或单独的热点链接页面中包含一些 Internet 网站的链接。人们通常在自己的主页上包含朋友的个人主页链接。而商业网站应注意不要使潜在的用户太快离开本网站到其他网站去——没人能保证它们会回来！

**注意：**在自己网页的链接中使用完整地址，也有一个理由。如果用户将你的网页保存到自己的硬盘中，那么，网页上的所有链接都将无效，除非他下载了所有网页或你使用了完整地址。

对于地址问题，一种中庸的解决方法是：在每个页面底部都包含一个到主页的链接，并使用完整地址；而其他链接只使用文件名或相对地址。

**By the Way**

记住，在自己网页的链接中只用文件名（或相对地址），到其他网站的链接使用完整的 Internet 地址。

## 3.4 在新浏览器窗口中打开链接

现在你已经知道如何创建链接地址，下面将介绍一个略有争议的链接问题。你肯定听说过弹出窗口，它是一个不经过用户同意就打开和显示的新浏览器窗口。虽然弹出窗口被一些激进的广告商滥用，然而，当你希望在一个单独的网页浏览器窗口中显示信息时，弹出窗口仍是有效的方式。

被链接的网页在新浏览器窗口中打开时，原来的页面还在原来的窗口中。要实现这一效果，可在[<a>](#)标签中包含另一个属性 `target`，并将 `target` 属性设置为 `blank`，如下面的例子所示：

```
<a href="../zoo.html" target="_blank">return to the zoo.</a>
```

当用户单击这个例子中的链接文本，`zoo.html` 网页将显示在一个全新的浏览器窗口中。不鼓励过多地使用这种技巧，因为大多数人不喜欢四处弹出新浏览器窗口。一个有用的场景是链接到其他网站的页面时。这样你的网站仍活动在背景中，就不会忘记链接是从哪里开始的。

我已经介绍了在新浏览器窗口中打开网页的最简单方法，下面要说一个坏消息。为限制

Web 开发人员使用弹出窗口，W3C 在 XHTML 中删除了 `target="_blank"`。因此，如果要创建有效的 XHTML 页面就不能使用这种代码。但有一种方法可以解决这个问题。

### By the Way

**注意：**W3C 完全摒弃 `target="_blank"` 属性是有点过了。关于 Web 设计人员是否能够强行在弹出窗口中打开链接还有许多争论。我觉得只要用户在单击链接时有节制地使用弹出窗口，是可以接受的；没有提示用户，就不能使用弹出窗口。

对弹出窗口限制的一个解决方法是，使用 JavaScript 脚本偷偷地设置页面上所有链接的 `target` 属性。你可以觉得这样的方法有悖于 W3C 的初衷，你是对的。但实际上弹出窗口在有些情况下可有效地使用。在本课稍后，新版本的 Music City Mafia 曲棍球队网站就是一个例子。

如果你不知道 JavaScript，不必担心，因为你不需要知道 JavaScript 或使用现有的脚本进行编程。这个例子中的脚本存储在文件 `external.js` 中。这里也不会解释这些脚本代码，因为本书不是关于 JavaScript 的。我要介绍的是如何在网页中使用脚本。第一步是将 `external.js` 导入到网页中，这可以通过页面开头的一行代码来实现：

```
<script type="text/javascript" src="external.js"></script>
```

将脚本导入到页面中后，为创建一个在新窗口中打开的链接，要做的是在该链接的 `<a>` 标签中添加属性 `rel="external"`。脚本将查询这个属性，然后适当地修改链接，使它在新窗口中打开。最终的结果是，链接在新浏览器窗口中打开，而它仍遵循了 XHTML。

### By the Way

**注意：**代码 `rel="external"` 看起来有点奇怪，它指链接与当前页面之间的关系（relationship）是外部的（external）。也就是说，链接应在外部打开，如在独立的浏览器窗口中。

## 3.5 链接到 Google Maps

如果你没有用过 Google Maps，现在停止阅读，在网页浏览器中打开 <http://maps.google.com>。在搜索框中输入家庭或公司地址，然后单击“Search”按钮。在地图上单击并拖曳，地图将实时地移动。单击页面左边的按钮“+”或“-”可放大或缩小地图。另外，单击“Satellite”按钮可切换到地图的卫星视图，而在“Hybrid”模式下则可以看到街道的卫星图像。我为什么在有关 HTML 的书跑题，讨论 Google Maps 呢？因为它是 Web 上最酷的工具之一，对你自己的网页可能非常有用。图 3.5 显示了 Google Maps。

### By the Way

**注意：**说明一下，我和 Google 没有任何关系，推荐 Google Maps 也不会拿到任何报酬。我个人认为它是最神奇的在线工具之一。

介绍如何使用 Google Maps 后，下面介绍如何在网页中链接到地图。在提供公司的地图位置、重要事件的发展方向或其他事物上，这可能很有帮助。提供 Google Maps 链接的关键在于地图右边的“Link to this page”选项（见图 3.5）。单击这个选项，可以在浏览器的地址



栏中看到当前地图的链接。然后可以将该链接复制并粘贴到 HTML 代码中，作为<a>标签的 href 属性。这样就创建了一个指向 Google Maps 上确切位置的链接。

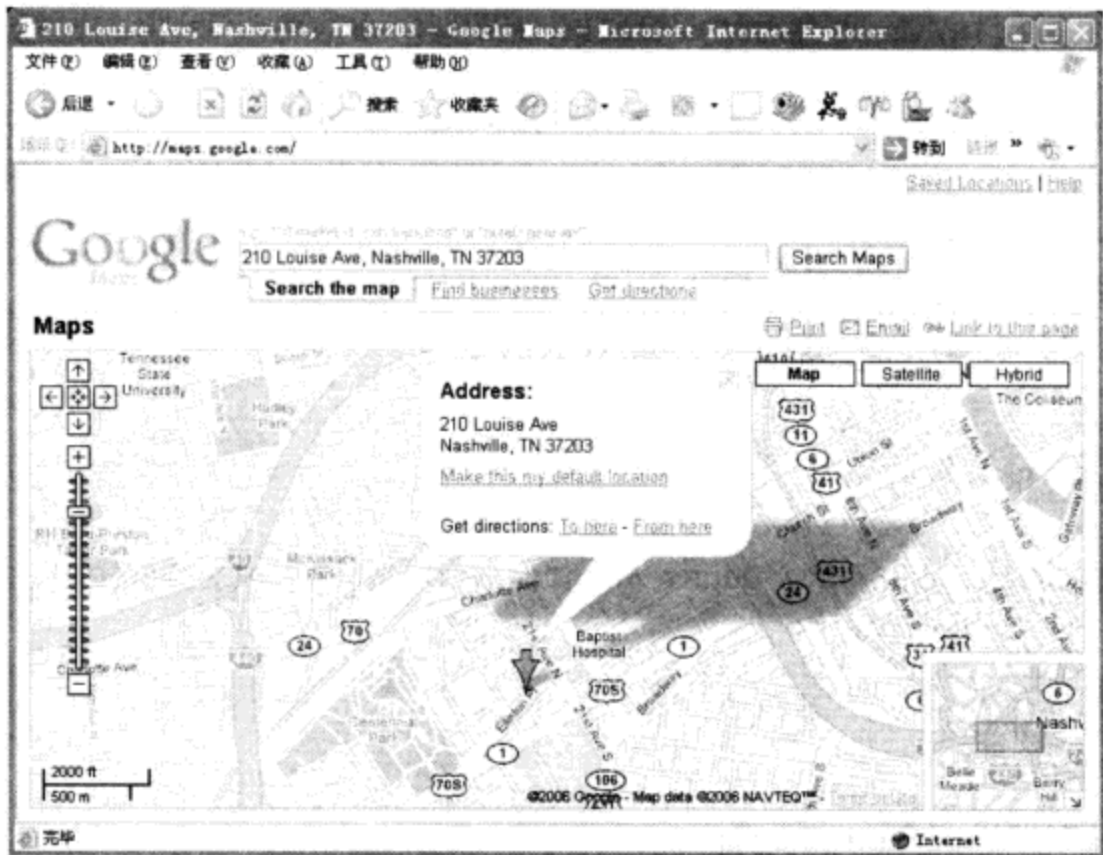


图 3.5  
Google Maps 让你能够创建到任何地理位置的交互性地图的链接

**休息一下：**如果你认为 Google Maps 是一个有趣的工具，可能也会对 <http://earth.google.com> 的 Google Earth 感兴趣。与 Google Maps 不同的是，Google Earth 是可以下载到本地计算机的应用程序。Google Earth 在成像上要强大得多，有效地提供了一种可交互的全球照片视图。

Coffee  
Break

举个例子，图 3.5 中的地图显示了我一个朋友开的一家餐馆（Café Coco）的地理位置。如果我的朋友要在他的网站上包含餐馆位置的地图链接，他需要做的只是在网页中包含下列 HTML 代码：

```
<a href="http://maps.google.com/maps?q=210+Louise+Ave,+Nashville,+TN+37203"> View map</a>
```

这些代码使用文本“View Map”作为到 Café Coco 的 Google 地图的链接。这个例子很好地表明，有时候只要很少的工作就可以在网页中增加很多有趣的功能。

### 3.6 添加到实用示例的链接

在结束本课前，修改一下曲棍球队网页的例子，在其中加入一些链接。实际上，我已经对这个例子中的各个队员的网页进行了扩展。这个例子很好地说明了如何将同一网站的网页链接在一起，以及如何插入到其他网站的有用链接，包括 Google Maps。

程序清单 3.4 列出了我的业余曲棍球队中，另一个队友的网页的 HTML 代码。在这个例子中，该队友的兴趣爱好链接到其他相关的网页。

## 程序清单 3.4 链接到其他提供额外信息的网页

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Music City Mafia - Donny Sowers</title>
    <script type="text/javascript" src="external.js"></script>
  </head>

  <body>
    <h1>21 - Donny Sowers</h1>
    <p>
      <br />
      Nickname: The Donger<br />
      Position: LW<br />
      Height: 5'9"<br />
      Weight: 175<br />
      Shoots: Rarely<br />
      Age: 42<br />
      Birthplace: God's Country (Texas)
    </p>
    <hr />
    <p>
      Favorite NHL Player: <a
      href="http://www.nhl.com/lineups/player/8458520.html">
      Peter Forsberg</a><br />
      Favorite NHL Team: <a href="http://www.nashvillepredators.com/">Nashville
      Predators </a><br />
      Favorite Southern Fixin: <a href="http://www.jackdaniels.com/">Jack
      Daniels</a><br />
      Favorite Meat and Three: <a href="http://www.judgebeans.com/">Judge Beans
      Bar-B-Q & Steakhouse</a> (<a
      href="http://maps.google.com/maps?q=123+12th+ave+n,+nashville,+tn"
      rel="external">map</a>)
    <br />
      Favorite Country Star: None<br />
      Favorite Mafia Moment: "Any time <a href="mcmplayer_dfirlus.html">Duke</a>
      scores a game-winning goal but we still somehow lose the game."
    </p>
  </body>
</html>

```

大多数有趣的链接代码都在文档的后半部分。在解读这些代码前，先看看图 3.6 中的结果。

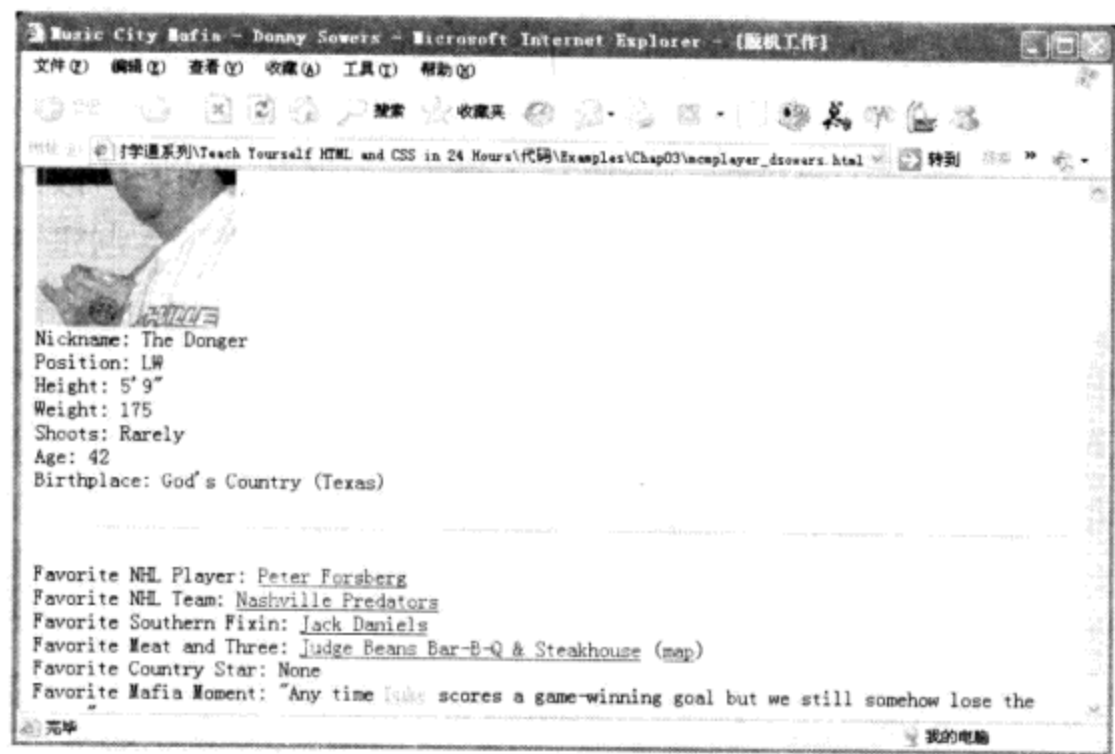


图 3.6  
几个链接用于提供关于这位曲棍球队员的兴趣爱好等信息

**注意：**在图中我特意向下滚动，使所有链接都显示出来。

By the Way

这个网页和你在前一章看到的曲棍球队员的网页并没有太大的不同，只是在这个网页中有这位队员兴趣爱好的一些链接。单击这些链接将进入对应的页面。然而，这些链接不是相同的。如果仔细看一下程序清单 3.4 中的代码，将注意到最后一个链接“Duke”指向同一网站的另一名队员的网页。将该链接与 map 链接比较，map 链接是指向 Google 地图的外部链接（弹出窗口），在这个例子中，map 链接提供了一个到该队员最喜欢的宾馆 Judge Beans Bar-B-Q & Steakhouse 的地图链接。其他所有链接都是普通的指向其他网站的链接。图 3.7 显示了单击页面上的 Duke 链接将打开的页面。



图 3.7  
链接到其他队员的网页，只要在 <a>标签中指定网页的文件名

引用上图所示的网页时，只在 HTML 代码中将<a>标签的 href 属性设置为该 HTML 文件的文件名。在本章前面提到过，只指定网页的文件名或部分路径（相对路径）时，假设网页与链接它的网页在同一台计算机上。

### 3.7 总结

<a>标签实现了“超文本”。使用这个标签，可以创建可单击的页面间链接，以及指向页面上特定锚点的链接。本章集中介绍了使用相对路径或绝对路径来标识网页、创建到其他页面的简单链接。

创建指向其他人的网页的链接时，应在<a href>标签中包含每个网页的完整 Internet 地址。对于指向自己网页的链接，只要包含文件名或足够的目录信息。

表 3.1 总结了本章介绍的<a>标签。

表 3.1 第 3 章介绍的 HTML 标签及其属性	
标签/属性	功能
<a>...</a>	使用 href 属性可创建指向另一个文档或锚点的链接。
属性	
href="address"	要链接的文档或锚点的地址。
Target="_blank"	在新浏览器窗口中打开链接的页面，但该属性不符合 XHTML 的要求（可以使用 JavaScript 来避开这种限制）。

### 3.8 问与答

问：我创建链接时，有些链接是蓝色的，有些是紫色的，这是为什么？为什么在 Internet 上见到的大多数链接既不是蓝色也不是紫色的？

答：默认情况下，蓝色链接表示用户最近还没有访问的页面。访问页面后，链接就变为紫色的。这些默认颜色可被网页开发人员修改，以匹配他希望的颜色主题（且通常都被修改），因此在 Web 上看到的许多链接既不是蓝色也不是紫色。还可以使用级联样式表（CSS）来修改链接中字体的大小、粗细和其他字体属性。第 9 章和第 13 章将介绍如何修改链接的颜色和外观。

问：如果链接到 Internet 上的一个网页，然后该网页的拥有者将这个网页移走或删除了，将发生什么情况？

答：这取决于这个人如何设置其服务器。通常，当人们单击这个链接时，将看到“无法显示网页”等信息；他们仍可单击“返回”按钮返回你的网页。

问：链接在我的计算机上工作得好好的，但当将网页放到网上后，链接就不能正常使用了。这是为什么？

答：下面是最有可能的一些原因：

- 大小写错误。在 Windows 计算机上，使用<a href="freddy.html">链接文件名为 Freddy.html 的页面是可行的。在大多数 Web 服务器上（通常是 UNIX 机器），链接必须为<a href="Freddy.html">（或将文件名改为 feddy.html）。更糟糕的是，有些文本编辑器和文件转换程序在用户不知情的情况下修改了文件名的大小写。最好的解决方案是所有网页的文件名都采用小写。



- 文件名包含空格。大多数 Web 服务器不允许文件名包含空格。例如，不应将网页命名为 my page.html，而应命名为 mypage.html 或 my\_page.html。
- 本地绝对地址。如果在链接中使用了本地绝对地址，如 C:\myWebsite\news.html，将网页放到 Internet 上时，链接将无效。不应使用本地绝对地址；本地绝对地址通常是因创建临时链接来测试网页的一部分而产生的。因此，将网页发布到 Web 前，务必删除这样的测试链接。

下一章将介绍如何上传文件到网站，以及上传后如何重命名文件以保证拼写和大小写是正确的。

## 3.9 作业

作业包含一些测试题及练习，帮助巩固对这一章介绍的内容的理解。试着回答所有问题，然后再看后面的“答案”小节。

### 3.9.1 测验

1. 你小学时最好的朋友通过 Internet 找到了你，并告诉你他想要和你交换主页链接。你如何在网页上添加一条链接，指向他的网页 [www.cheapsuits.com/~billybob/](http://www.cheapsuits.com/~billybob/)？
2. 将网页发布到 Internet 上时，主页将放到 <http://www.mysite.com/home.html> 上。在这个页面中编写 HTML 代码，当访问者单击 all about me 时，将显示 <http://www.mysite.com/mylife.html> 页面。
3. 你计划发布一张包含 HTML 的光盘。如何在\guide 目录中的一个页面创建一个指向\guide\maine\katahdin.html 页面的链接？
4. 如何在\guide\maine\katahdin.html 页面中创建指向\guide\arizona\superstitions.html 页面的链接？

### 3.9.2 答案

1. 将下列代码加入网页：

```
<a href="http://www.cheapsuits.com/~billybob/">
my buddy billy bob's page of inexpensive businesswear</a>
```

2. `<a href="mylife.html">all about me</a>`

下面的代码也可行，但它将比较难于在本地硬盘上测试：

```
<a href="http://www.mysite.com/mylife.html">all about me</a>
```

- 3.

```
<a href="maine/katahdin.html">mount katahdin</a>
```

4.

```
<a href="../../../arizona/superstitions.html">  
the superstition range</a>
```

### 3.10 练习

创建一个网页，网页包含你喜欢的网站列表。你可能已经将这些网站收藏在网页浏览器中，如果是这样，可以访问这些网站，在浏览器的地址栏中找到这些网站的地址。



## 第 4 章

# 发布 HTML 网页

现在到了你一直在等待的这一章！现在，网页可以发布了，这一章将解释如何将网页显示在目标观众眼前，这些观众可能是你的朋友、同事或网上用户。

发布网页最明显的途径当然是 Internet，但你可能希望将网页的发布限制在组织内部的局域网，而不对外公开。也可能选择通过 CD-ROM、DVD-ROM、USB 存储卡或软盘发布网页。这一章将介绍所有这些方式，并提供关于如何设计网页使之适合于选定发布方式的一些建议。

内部网是私有网络，其访问仅限于一个组织，但使用的技术标准和协议与全球公开的 Internet 相同。

练习

学习发布网页前，应先考虑要使用哪种发布方法。

- 如果希望网页在全世界有尽可能多的人看到，就必须使用 Internet。但不要忘记其他发布方法；基于 Internet 的网页也容易用于光盘或局域网发布。
- 如果要面向组织内部的员工发布，就只在局域网中发布。
- 如果提供很大的图像、多媒体或即使通过电缆或 DSL Modem 连接传输起来也很慢的其他内容，可考虑使用 CD-ROM 或 DVD-ROM。可以在网站上提供 CD-ROM 给那些通过 Internet 找到你并希望获得更多体验的用户。当然，那时必须设计一个很大的网站才能让用户向你索取光盘，特别是当你看看现在的一些网站使用的多媒体有多少。
- 如果准备会议展示，并可能将相关材料发布在 Internet 或局域网上，为什么不使用 HTML 而使用 PowerPoint 呢？还可以将这些文件放在存储卡中，必要时也刻光盘，让需要的人拿走。

## 4.1 在Internet建立Web空间

要使HTML网页成为可公共访问的万维网的一部分，需要将网页放在Web服务器上。如果自己运行Web服务器，只需将文件复制到正确的目录下。然而，你如果没有Web服务器用于存放网站，就需要使用Internet服务提供商（ISP）的Web服务器来存放。

Web服务器是长期与Internet连接的计算机，使用特殊软件应Web浏览器的请求传输网页。

几乎所有提供Internet接入的ISP现在也都提供存放个人网页的空间，这些空间是免费的或费用很低，然而，如果希望网站能够吸引大量的访问者或包含很大的多媒体文件，可能必须支持更多的费用。

先检查你获得Internet连接的公司是否为标准用户提供Web空间。这通常是开始创建网站的一个好途径，因为空间是免费的，且立即可以使用。

### Did you know?

**提示：**即使空间开始是免费的，也不一定要选择提供Internet连接的公司来存放网页，特别是当他们不提供高级功能，如购物车和在线商务服务时。你可能还希望拥有自己的域名（[www.yourname.com](http://www.yourname.com)），这通常都要使用付费的托管服务。最后，如果计划运行高流量的商业网站，从专门的Web托管服务提供商那里可以获得更多的功能和更可靠的服务。可以在<http://thelist.internet.com/>上查看ISP列表，比较各个Internet服务提供商提供的托管服务。

创建网站的Web托管服务价格低至每月3美元，然而，如果有许多人开始浏览你的网页，以后价格可能上涨。对于有一百个网页左右的网站，当有几千访问者访问网页时，支付的费用是每月3美元，但当有几十万人浏览时价格就是每月2000美元。价格最终是由网站的访问量决定的，虽然一开始你可能希望访问量很高。还要记住，Web托管服务越来越便宜，因此你可能在我提到的低端范围内找到合适的托管服务。

一些免费的Web托管服务，如Geocities（<http://geocities.yahoo.com/>）、Tripod（<http://www.tripod.lycos.com/>）和Angelfire（<http://www.angelfire.lycos.com/>），很受网页制作者的青睐，它们确实是免费的，但大多数这样的服务都要求在网页中包含广告。免费服务的另一个问题是它们经常显示弹出广告，这很烦人。我知道Angelfire提供了一种选择，如果每月付费5美元左右就没有广告，因此，如果预算很紧，这是个不错的选择。

### By the way

**注意：**创建网站时最重要的选择之一是作为网站地址的名字。如果没有付费注册一个域名——每年大概需要30美元，网站地址将包含Internet服务提供商的名称（如<http://www.shore.net/~smith/>），如果愿意付费并注册一个域名，可以选择其他公司还没有用过的名字（如<http://www.mister-smith.com/>）。如果Web托管服务不提供域名注册，可以使用域名服务，如<http://www.godaddy.com/>、<http://www.easydns.com/>或<http://www.register.com/>提供的服务。在注册新名字前，首先必须搜索一下名字是否已经有人在使用；每个域名服务商都提供免费的查询服务。记住，不准备发布网站也可以注册域名；可以先注册域名，这样就不用担心在创建网站时域名被别人注册了。

## 4.2 将网页上传到 Web 服务器

Web 服务器通过 Internet 将网页发布给用户时,它使用超文本传输协议(Hypertext Transfer Protocol, HTTP)这种信息交换标准。然而,要将网页上传到网站,通常需要使用另一种通信标准——文件传输协议(File Transfer Protocol, FTP)的软件。

上传文件到 Web 服务器时,文件传输软件必须遵守 FTP 标准。使用 FTP 将文件上传到 Web 服务器后,有用户使用 HTTP 请求这些文件时,服务器将这些文件传输给用户。FTP 用于上传 HTML 文件到 Web 服务器,而 HTTP 是网页浏览器用于下载和浏览网页的协议。

将网页上传到网站前,很可能还需要从 Web 托管服务商那里获得四项重要的信息。

- 账号(有时称为用户名或用户 ID)。如果提供托管服务和提供 Internet 接入的是同一家公司,则网站账号可能是电子邮箱地址。
- 密码。这可能也是电子邮箱的密码。
- 服务器站点的 FTP 地址,如 ftp.yousite.com。注意,这个地址可能与用户在网页浏览器中访问你的网页的地址相同,也可能不同。
- 放网页的文件夹。有时可以将网页放在 Web 服务器的根目录下,但通常需要放在 www 或 public 这样的子目录下,或与你的域名相同的目录下。

接下来需要决定要使用哪种软件将网页上传到 Web 服务器以及维护网站。本章将介绍五种选择:

- Microsoft Internet Explorer;
- Mozilla Firefox (fireFTP 扩展);
- Microsoft FrontPage (或类似的 Web 开发软件);
- FTP 软件,如 SmartFTP、CuteFTP 或 FileZilla;
- Windows Web 发布向导。

选择哪种工具比较好呢?这取决于具体情况。如果要开发复杂的网站,Microsoft FrontPage 或 Macromedia Dreamweaver 这样的软件能够帮助管理页面间的链接,自动跟踪哪些网页已经被修改因此需要更新,从而节省大量的时间。然而,对于网页制作初学者和只想创建简单的个人或商业网站的人来说,学习和使用简单的 FTP 程序,如 SmartFTP 或 CuteFTP 是最简单的方法。如果只想将网页上传到网上而不想安装新软件,可使用已安装的网页浏览器或标准的 Windows Web 发布向导。

接下来的几章将详细介绍使用上述五种方法发布网页。在确定哪种方法最适合之前,可能需要尝试多种方法。

### 4.2.1 使用 Microsoft Internet Explorer

你可能已经从名字猜出来,Windows 的 Microsoft Internet Explorer 是随 Windows 提供的,



可以像文件管理器 Windows 资源管理器那样工作。在 Internet Explorer 的地址栏输入 FTP 地址时，可以对 Web 服务器上的文件或目录进行剪切、粘贴、删除或重命名，就像本地计算机硬盘上一样。

如果使用的是 Internet Explorer 的 Windows 版本，执行下列步骤将硬盘上已创建好的网页上传到在线网站。

1. 打开 Microsoft Internet Explorer。单击地址栏，输入“我的电脑”并按回车键，将可以看到计算机上所有的硬盘驱动器和文件夹。找到创建好的网页所在的文件夹。注意进入这个文件夹不能选择“文件/打开”——必须在资源管理器中浏览网页。

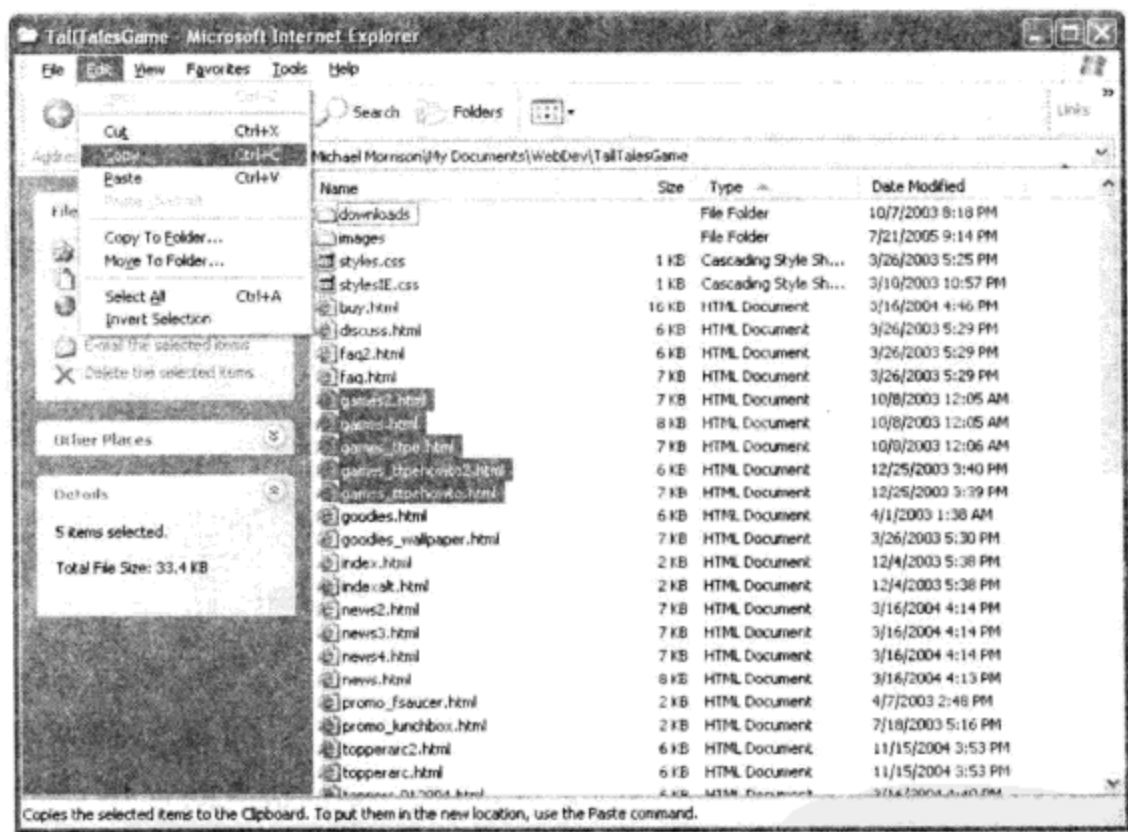
**Did you Know?**

**提示：**如果页面位于“我的文档”中，可以在地址栏中直接输入“我的文档”进入该文件夹。

2. 单击要上传到网站的文件选中它。与 Windows 程序一样，单击文件时可以按 Shift 或 Ctrl 选择多个文件。别忘了 HTML 文件包含的所有图像文件（见第 7 章和第 8 章）。

3. 选择“编辑/复制”（见图 4.1）。

图 4.1  
要使用 Microsoft Internet Explorer 上传网页，首先导航到网页存放的文件夹，然后选择“编辑/复制”



4. 现在再在 Internet Explorer 中单击地址栏，按照下面的格式输入网站账号、密码和网站地址：

```
ftp://myname:mypassword@mywebaddress.com/home/web/wherever/
```

用访问网站的账号和密码代替 myname 和 mypassword，用网站 FTP 地址代替 myWebaddress.com，用要放网页的顶级目录代替/home/Web/wherever/。

5. 等到 Web 服务器的文件夹在 Internet Explorer 中打开后，选择“编辑/粘贴”（见图 4.2）。文件将从硬盘传输到 Web 服务器。

6. 返回地址栏并输入网页的 URL，对网页进行测试。现在网页已经发布到网上了！

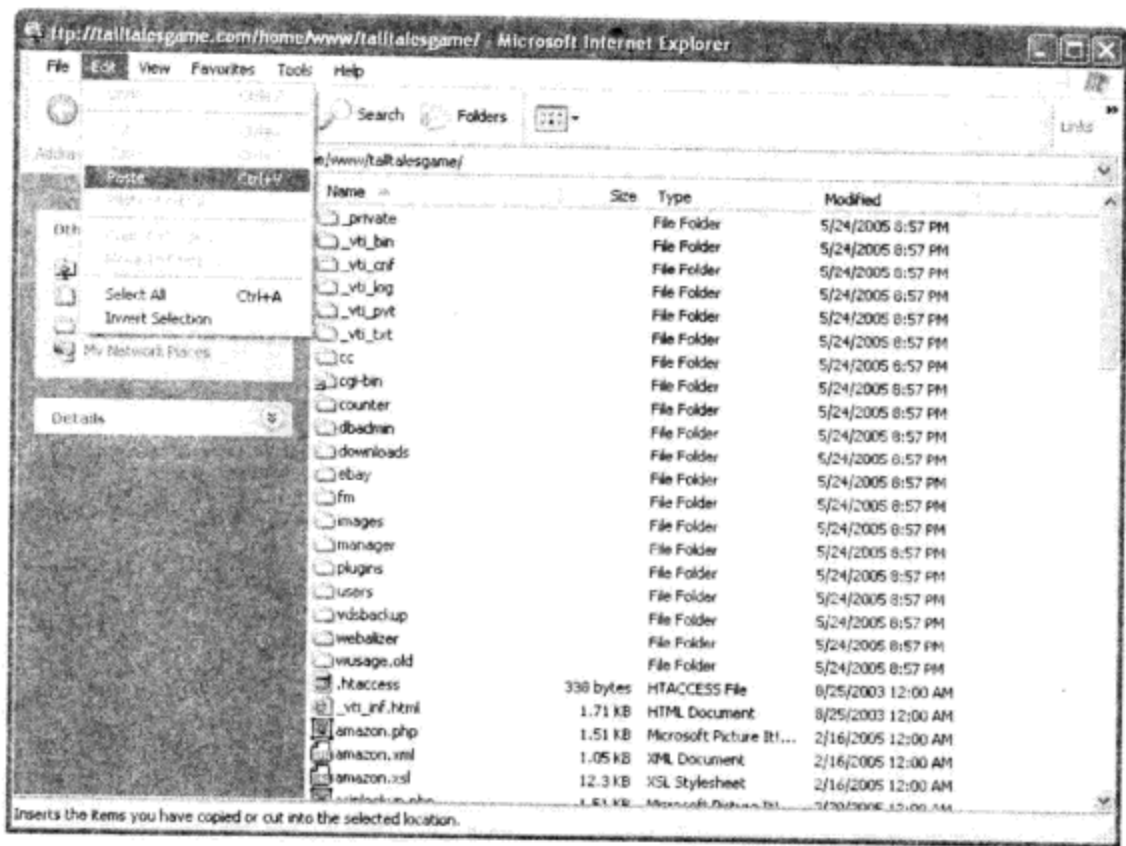


图 4.2  
Microsoft Internet Explorer 使用熟悉的 Windows 资源管理器界面，将文件粘贴到远程 Web 服务器

要删除或重命名 Web 服务器上的文件或文件夹，右击它然后选择“删除”或“重命名”，就像在 Windows 资源管理中对本地文件进行操作一样。

**提示：**在前三个步骤中，也可以使用 Windows 资源管理器而不是 Internet Explorer。在 Windows 中传输文件的另一个技巧是，将文件从 Windows 资源管理器中拖放到 Internet Explorer 中，而不必使用复制和粘贴的方法。

**Did you Know?**

将文件成功地发布到 Web 服务器上后，要在 Internet Explorer 的地址栏中输入网页的地址，然后按回车键，对网页进行测试。如果为网站注册了域名，可以在域名后输入一个斜杠 (“/”)，然后输入网页的文件名。否则，就需要输入 Web 托管服务提供商提供的 Web 服务器的名字，然后再加斜杠和文件名。

**警告：**通常不能直接将文件从一个 FTP 站点传输到另一个 FTP 站点。如果要这样做，需要先将文件传输到本地硬盘，然后再将它传输到新的 FTP 站点。这是因为大多数 FTP 程序只能在本地图算机和 FTP 站点之间传输文件，而不能在 FTP 站点之间传输文件。

**Watch Out!**

### 4.2.2 使用 Mozilla Firefox

如果更喜欢 Mozilla Firefox，也可以使用这个网页浏览器将文件上传到 Web 服务器。然而，首先需要安装一个浏览器 fireFTP。fireFTP 扩展可在 <http://fireftp.mozdev.org> 免费下载。要在 Firefox 中下载和安装这个扩展（然后执行下列步骤）；必须在 Firefox 中输入上述 URL 安装 fireFTP 扩展。

扩展是小型应用程序，类似于插件，可以下载并安装到 Firefox 中以增加新功能。

Firefox 下载软件时可能出现警告。如果出现警告，单击浏览器窗口右上角的“Edit

Option”，然后单击“Allow”，使浏览器可以下载扩展。可能出现是否安装扩展的确认，如图4.3所示。这时只要单击 Install Now 按钮，使 fireFTP 安装继续进行。安装完成后，需要关闭并重启 Firefox，才能使修改生效。这样就可以使用 fireFTP 开始上传网页了。

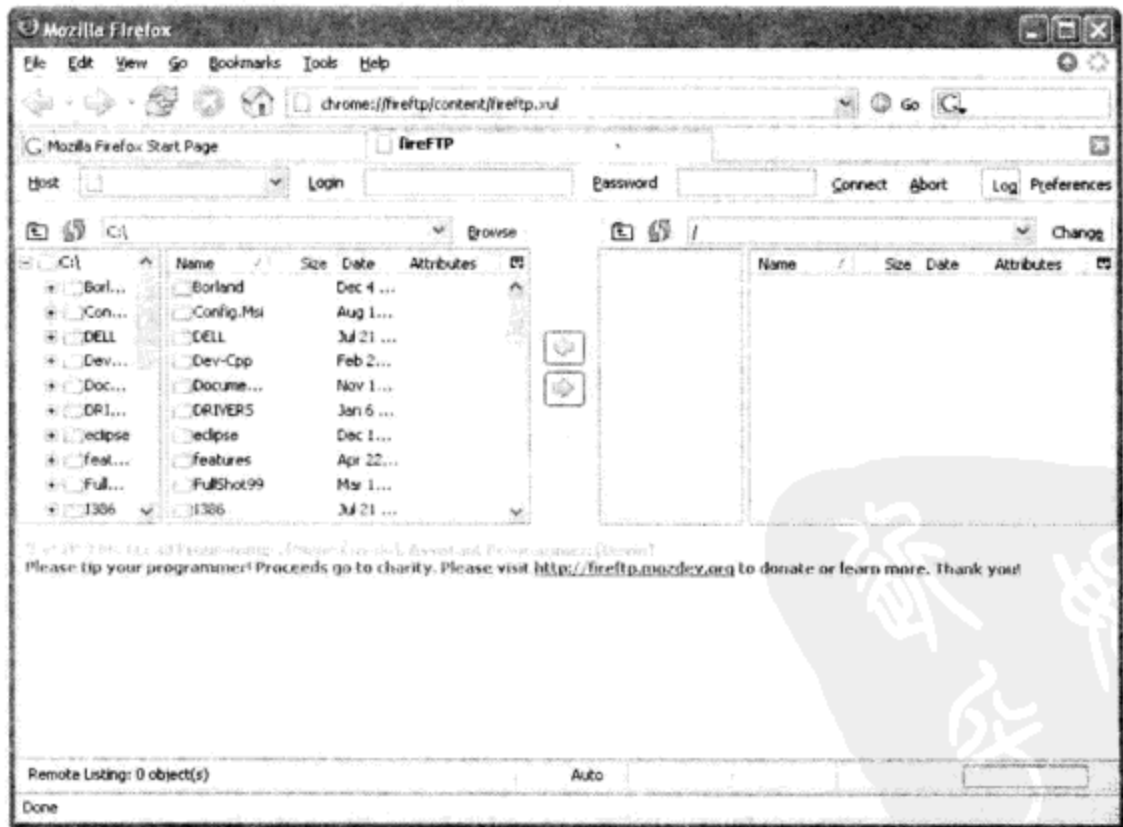
图 4.3  
提示是否在 Firefox 中安装 fireFTP 浏览器扩展时，只要单击 Install Now 按钮确认安装



要使用 Mozilla Firefox 和 fireFTP 浏览器扩展来发布网页，需要执行下列步骤。

- 1. 在安装 Mozilla Firefox 的文件夹下运行 Firefox；如果 Firefox 在桌面上有快捷方式，双击它。
- 2. 选择“Tool/fireFTP”启动 fireFTP 扩展。浏览器窗口将显示文件管理器视图，如图4.4所示。

图 4.4  
fireFTP 的用户界面有点像文件管理器，只是本地文件显示在左边的面板中，Web 服务器的文件显示在右边的面板中



- 3. 在 fireFTP 窗口顶部的文本框中输入网站地址、账号和密码。输入这些信息后，单击密码字段右边的 Connect，便可连接到 Web 服务器。如果一切顺利，将看到一个确认连接的通知框——只要单击 OK 继续就可以了。
- 4. 在左边面板中找到正确的文件夹，然后单击网页并将其拖曳到右边的面板。等待所

有文件都传输完毕。

5. 在浏览器的地址栏中输入网页的 URL，测试网页是否可访问。

虽然 Firefox 可将文件传输给 Internet 上的任何 Web 服务器，但也可以选择使用独立应用程序，如 smartFTP 或 cuteFTP。当然，如果你喜欢将 Firefox 作为主要的 Web 浏览器，完全可以使用 Firefox 和 fireFTP 来发布网页。由于有 fireFTP，Firefox 提供了比 Internet Explorer 更多的 FTP 功能。

### 4.2.3 使用 Microsoft FrontPage

虽然可以使用 Internet Explorer 或 Firefox 来发布网页，但如果使用完整的 Web 开发软件包，还有更高级的选择。一个这样的软件包是用于 Windows 的 Microsoft FrontPage，当你掌握 HTML 后，它也是创建网页的有用工具。FrontPage 将整个网站的发布变成一个只需一步的自动化过程（见图 4.5）。不管是创建了一个网页还是创建上百个网页，只要选择“文件/发布 Web”就可以将所有网页上传到网上。也有 Macintosh 兼容的 Web 开发工具可用于发布网页，你将在后面学到。



图 4.5  
使用 Microsoft FrontPage 可以一次上传许多网页或只上传修改过的网页

第一次将网页发布到网站时，需要指定 Web 服务器的地址，也就是提供 Web 托管服务的服务器的 FTP 地址（如果 Web 服务器安装了 FrontPage 扩展，也可以使用 HTTP 地址）。

接下来将要求输入账号和密码。FrontPage 自动上传最后一次更新以来修改过的所有网页，它跟踪最后一次发布后哪些网页被更新了。其他 Web 开发软件，包括 Macromedia Dreamweaver 和 Adobe GoLive，也提供类似的自动化上传和站点管理功能。

**注意：**在 FrontPage 中指定要发布网页的地址时，要注意包含正确的子目录。例如，sover.net 的 Web 托管要求网页放在目录 www 中，也就是说文件必须上传到 ftp.sove.net/www/。不要忘记最后的 www/，否则 FrontPage 将网页上传到错误的目录下。

By the Way

### 4.2.4 使用 FTP 软件

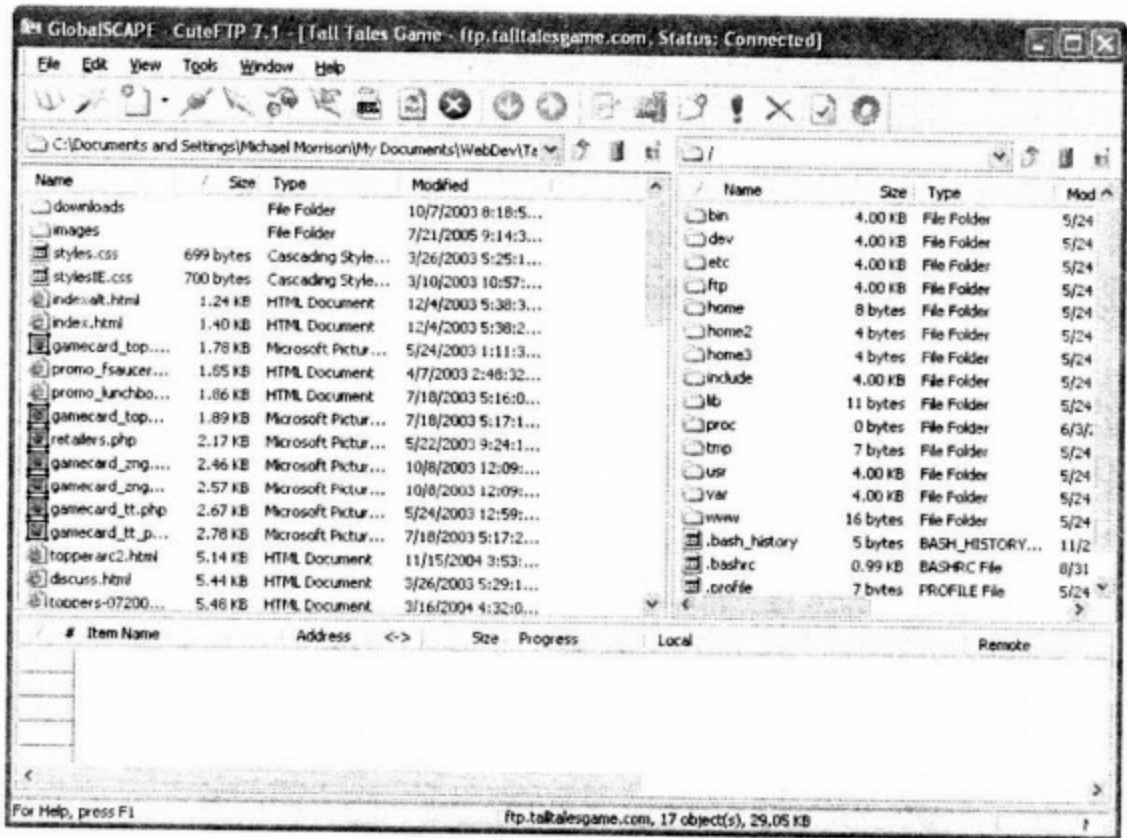
如果不喜欢上述几种发布网页的方法，或希望对如何在 Web 服务器上组织 HTML 文件及其相关的资源进行更多控制，可考虑使用 FTP 软件。FTP 软件可以在 Web 服务器上传输、删除和重命名文件，也可以创建、重命名和删除目录。这让你能够控制网站的组织结构。有许多 FTP 应用程序，大多数是免费的或只要一点注册费。CuteFTP 是流行的 Windows FTP 程



序，我有过成功的使用经验。CuteFTP 的试用版本可在 <http://www.cuteftp.com/> 免费下载。图 4.6 显示了运行中的 CuteFTP，它显示了 Web 服务器上的文件。

图 4.6

CuteFTP 是一款功能强大、用户友好的 FTP 应用程序，个人用户可免费使用



如果使用的不是 Windows 机器，不必担心，因为有很多类似于 CuteFTP 的 FTP 软件包可用于其他操作系统，如 Macintosh 和 UNIX 系统。Fetch (<http://www.fetchsoftworks.com/>) 和 Transit (<http://www.panic.com/transmit/>) 是两个流行的 Macintosh FTP 应用程序，在 UNIX 上可使用 gFTP (<http://gftp.seul.org/>)。也可以使用你最喜欢的搜索引擎来搜索“ftp 客户端”或“ftp 软件”，看看还能找到什么。

**By the Way**

**注意：**另一个流行的 Windows FTP 应用程序是 SmartFTP，可在 <http://www.smartftp.com/> 下载。从可免费下载试用版的网站下载或在线购买完整版。另外，也可以考虑 FileZilla，这是一个开源的 Windows FTP 客户端，可在 <http://filezilla.sourceforge.net/> 下载。

不管使用的是哪种计算机、哪种 FTP 软件，发布网页的过程都很简单。我将以 CuteFTP 为例演示发布网页的步骤，但实际上使用任何 FTP 软件的过程都是相似的。执行下列步骤。

1. 首先必须将 Web 服务器的地址、账号和密码告诉 FTP 程序，才能访问服务器（见图 4.7）。单击窗口顶部的标签，可以指定关于 FTP 站点的其他信息。
2. 下面是如何填写图 4.7 所示的每项。
  - Label 是为网站指定的名称。其他人不能看到这个名称，因此可以输入任何名称。
  - Host Address 是要上传网页的 Web 服务器的 FTP 地址。这一地址通常（但不总是）以 ftp 开始。注意这个地址可能是也可能不是用于访问你的网页的地址。提供 Web 服务器的 ISP 应能够告诉这里要输入的正确地址。
  - 运行 Web 服务器的公司也会提供用户名和密码。要知道，CuteFTP（以及其他大多数 FTP 程序）会自动记住密码，也就是说其他可以使用你的计算机的人也可以修改



你的网站。

- 应将 Login method 设置为 Normal，除非有人让你选择其他类型。（Anonymous 设置用于从公开的 FTP 服务下载文件，不要求用户 ID 或密码）。
- 通常，不需要修改 CuteFTP 中其他选项卡的设置，除非在连接时遇到问题。如果遇到，让 ISP 帮助做出最好的设置。

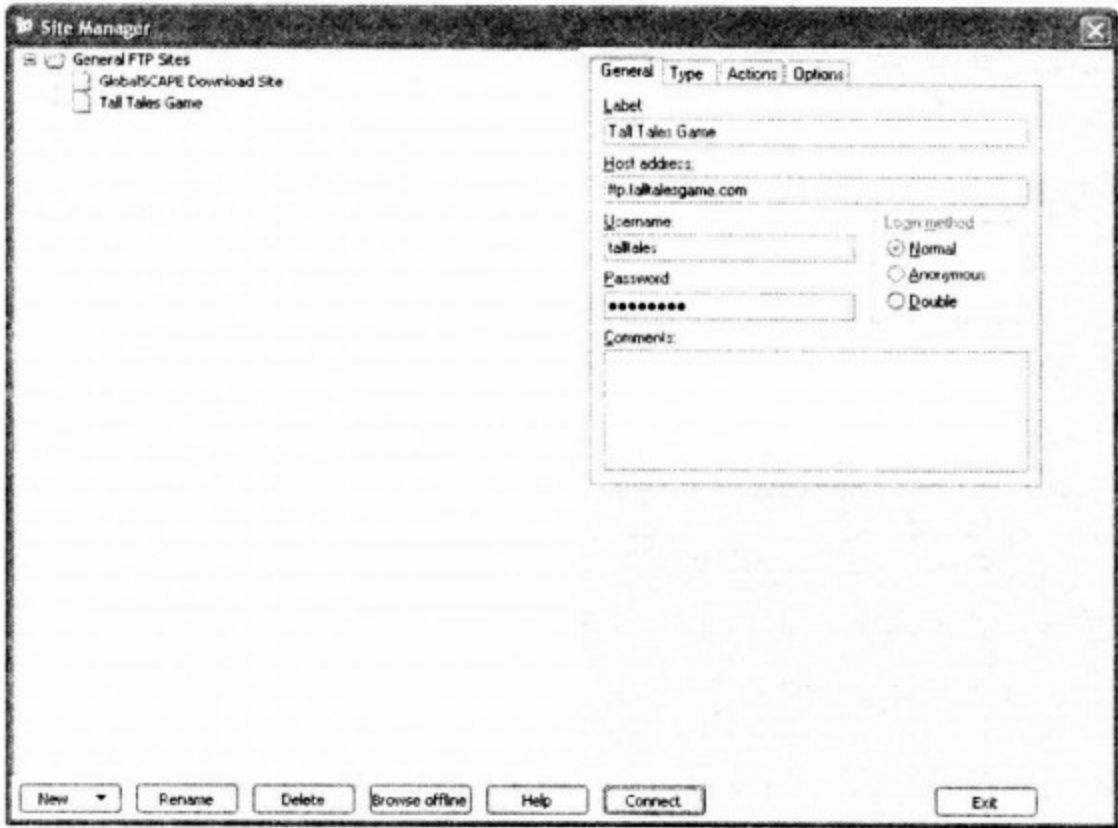


图 4.7  
CuteFTP 包含直观的 FTP 站点管理器，虽然大多数网页制作者只需要 FTP 站点登录信息

3. 设置完后，单击 Connect，建立到 Web 服务器的连接。

大多数服务器都发送一条简单的信息给与服务器连接的用户。许多 FTP 程序都忽略这条消息，但 CuteFTP 会显示这条消息。这条信息不包含什么重要的信息，因此单击 OK 即可。

4. 连接到服务器后，将看到两个文件列表，如图 4.6 所示。左边的窗口面板列出本地计算机上的文件，右边的面板列出 Web 服务器上的文件。

要将网页传输到服务器，在左边的窗口中选择 HTML 文件以及相关的图像文件（记住在 Windows 程序中，可以按住 Ctrl 并单击鼠标来选择多个文件）。然后选择 Transfer/Upload 或单击工具栏上的 Upload 按钮。也可以通过直接将文件从一个面板拖到另一个面板来传输文件。

CuteFTP 还包含（在本地计算机或服务器上）删除或重命名文件的命令以及创建和修改目录的命令。

**提示：**大多数 Web 服务器都使用一个特殊的文件名，如果用户请求页面时不指定文件名，就将这个特殊的文件传送给用户。例如，如果访问网站 <http://www.talltalesgame.com/>，该网站的 Web 服务器将自动将 index.html 发送给你。其他 Web 服务器可能使用不同的默认文件名，如 default.html 或 welcome.html。

询问 ISP 它使用的默认文件名是什么，以便将主页命名为该文件名。

**Did you know?**

- 5. 在网页浏览器中输入网页的 URL，立即就可以看到刚上传到 Web 服务器的网页。
- 6. 将文件上传到 Web 服务器并修改完后，选择 File/Disconnect，关闭连接。

大多数 FTP 应用程序都会记住每个 FTP 连接的设置，因此下次上传网页时，不必填写第 2 步中的信息。可以只单击 Connect，选择要上传的网页，然后单击 Upload 按钮。

By the Way

**注意：**在大多数 Web 服务器上，页面一旦上传，立即就可以通过万维网进行访问。然而，有些服务器要求用户手工修改文件权限设置，控制哪些人可以访问哪些文件。ISP 会告诉你如何在服务器上修改权限设置以及是否需要这么做。

### 4.2.5 使用 Windows Web 发布向导

至此，介绍了几种发布网页的方法，从最简单的到功能最强大的。现在回过头来，介绍一种非常简单但有局限性的方法。这就是与标准的 Windows XP（家庭版和专业版）一起发布的 Web 发布向导。Web 发布向导非常容易使用，但支持的 Web 托管服务有一定的限制。如果使用的服务是它不支持的，就不得不使用其他方法来发布网页。

By the Way

**注意：**Web 发布向导支持的默认 Web 服务器提供商之一是 MSN Groups，这是一个在线社区，是 MSN（Microsoft Network）的一部分。MSN Groups 的详情请参见 <http://groups.msn.com/>。

顾名思义，Web 发布向导使用的是 Windows 用户熟悉的向导界面，也就是说用户可以通过在连续的多个窗口中回答一系列问题来发布网页。要通过 Web 发布向导发布网页，必须从 Windows 资源管理器开始。下面是使用 Web 发布向导发布网页的步骤。

- 1. 启动 Windows 资源管理器，进入包含要发布的 HTML 文件的文件夹。单击资源管理窗口左边的“将该文件发布到 Web”。
- 2. 这时将打开 Web 发布向导。单击“下一步”按钮开始发布网页的过程。
- 3. Web 发布向导的第一步确认是否要发布当前文件夹下的文件，如图 4.8 所示。选择要发布的文件后，单击“下一步”。

图 4.8  
单击选中要用 Web 发布向导发布的文件



4. 在 Web 发布向导显示的列表中（见图 4.9）选择 Web 托管服务提供商。如果列表中不包含你的提供商，可以跳过，进入下一步，使用另一个方法来上传文件。

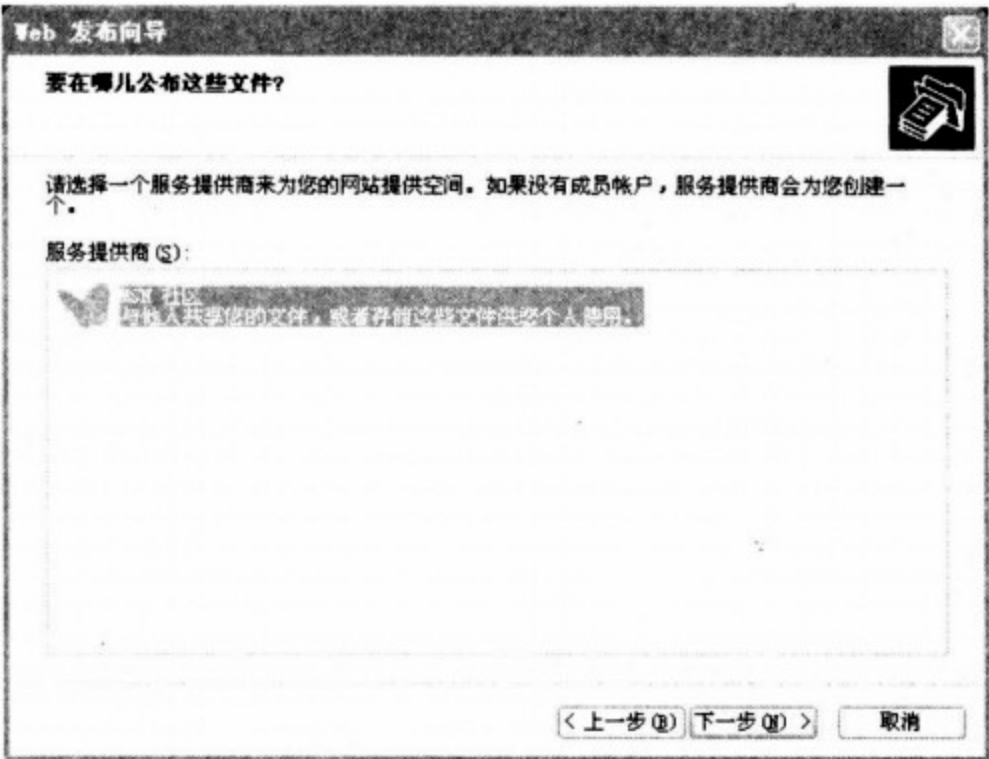


图 4.9  
只要你使用的 Web 托管服务提供商得到支持，Windows Web 发布向导就能为你提供简单的方法来发布网页

- 5. 输入选择的 Web 托管服务提供商所需的信息。这些信息随服务提供商而异，因此如果不是很清楚向导要求输入什么时，需要询问提供商。
- 6. 继续到向导的最后一步，网页将上传到 Web 服务器。如果上传的网页很多或 Internet 连接速度不太快，可能需要等待几分钟。
- 7. 在网页浏览器中输入网页的 URL，对网页进行测试。

### 4.3 使文件可下载

知道如何将网页发布到网上后，下面介绍另一个与发布 HTML 文件相关的问题。我指的是文件而不是要发布到网上的网页。不管是包含艺术作品的 ZIP 文件，还是包含销售数据的 Excel 表格，经常需要将这种不是网页的文件发布到 Internet。

要使不是网页的文件在 Web 上可以访问，只要按照前面介绍的上传文件的步骤，像 HTML 文件一样将文件上传到网站。当文件上传到 Web 服务器后，需要在网页上创建一个到该文件的链接，这在第 3 章介绍过。例如，如果文件名为 `artfolio.zip`，链接应如下所示：

```
<a href="artfolio.zip">Click here to download my art portfolio.</a>
```

记住，有些 Web 托管服务器是根据传输出去的字节数收费的，因此，如果有很多人下载你的文件，那么对上传大文件要谨慎一点。比如，如果每天有一万个人从你的网站下载 2MB 的文件，你需要支付的费用将很可观，网站负载将很大。当然，除非你创建的是下一个雅虎或 eBay，否则，按照正常的标准，这样的网站流量是很大的。另外，如果你的网站有很多人访问，就能获得许多广告收入了。



## 4.4 其他发布HTML的途径

将HTML页面发布到网上显然是学习HTML和创建网页的首要原因。然而，在某些情况下，有些发布形式无法使用。例如，在贸易展览上你可能希望将营销资料设计为HTML，放在CD-ROM或DVD-ROM。又如在培训课程上，你可能希望在软盘、存储卡或CD-ROM上包含基于HTML的指南手册，分发给学生。这只是两个例子，说明HTML页面不一定是发布到Internet。下面几章将介绍这些发布途径。

### 4.4.1 将网页放在内部网上

虽然你学到的用于发布网页到Internet的方法通常也适用于发布到公司私有的内部网，但各个公司内部网的差别相当大。在有些情况下，可能需要使用FTP程序将文件传输到内部网的服务器。而在有些情况下，可能可以使用本地计算机上的文件管理软件来传输文件。可能还需要调整权限设置或对隔离私有内部网和公共Internet的防火墙进行配置。

关于将网页发布到内部网，最好询问系统管理员。他可以帮助你将在公司的服务器上，以最好的方法保证网页的可访问性和安全性。

### 4.4.2 将网页发布到磁盘或存储卡上

除非为公司创建要放到公司内部网的文档，否则Internet是发布网页的最好途径，大部分时候这种想法是正确的。然而，由于下列三种原因，有时需要考虑使用磁盘形式的发布：

- 目前，有磁盘驱动器和USB端口的人比可以连接Internet的人多；
- 磁盘和存储卡传输信息的速度比从Internet下载的速度快；
- 通过发布磁盘和存储卡可选择目标用户，不管这些用户是否可以连接Internet或内部网。

在不久的将来，可访问Web的电视和高速网络更为普及后，上述优势可能会丧失。现在，通过磁盘发布仍是可行的，如果要选择用户的话。另外一点可能是更重要的，不能连接Internet的情况下可以使用磁盘将网页发布给用户。

#### **Coffee** Break

**休息一下：**对于各种存储卡及其特性的详细分析，可参考[http://en.wikipedia.org/wiki/Memory\\_card](http://en.wikipedia.org/wiki/Memory_card)的Wikipedia上关于存储卡的介绍。这里有大量好的信息，包括各种存储卡平均每G字节的价格。

要将网页发布到CD-ROM、DVD-ROM或存储卡上，只需用文件管理器软件将硬盘上的文件复制到这些介质上。但要记住一点，以http://开始的链接只能在连接Internet的情况下才能使用。使用CD-ROM的成本很低，但DVD-ROM稍微贵一点。存储卡要贵得多，但可以反复使用。

**警告：**不要在网页的<a href>链接标签中使用驱动盘字符（如 C:）；否则，当网页复制到其他磁盘上时，链接将失效。关于如何创建在磁盘上和 Internet 上都可使用的链接，请参见第 3 章。

**Watch Out!**

要将网页发布到 CD-ROM 或 DVD-ROM 上，要有能够刻录磁盘的驱动器（及相关软件），也可以将文件发送给母盘制作和复制公司，让他们帮你发布。CD-ROM 复制的成本很低，特别是考虑到一张 CD-ROM 可以存储很多（650MB）信息时。DVD-ROM 的价格有些波动，也比较贵，但性价比与 CD-ROM 差不多。DVD-ROM 可存储的信息要多得多，单张容量在 4.7GB（单面、单层）到 17.1GB（双面、双层）。

**提示：**读取 HTML 网页必须使用浏览器软件。然而，现在几乎人人都有网页浏览器，因此发布网页时不需要提供网页浏览器。如果要包含浏览器，可以考虑 Opera，这个浏览器包含 Mozilla Firefox 和 Microsoft Internet Explorer 的大多数基本特性，但它小到可以放在一张 1.44MB 的软盘或更小的存储卡上，且有 30 天的免费试用期。Opera 可从 <http://www.operasoftware.com/> 下载。如果得到 Microsoft 和 Netscape 的同意或付许可费，也可以将它们的浏览器存储在 CD-ROM 中。没有得到 Microsoft 的书面许可，千万不要发布它的软件拷贝，因为它会不惜一切确保你获得许可。

**Did you Know?**

存储卡比 CD-ROM 和 DVD-ROM 贵，但更方便，且价格也在下降。存储卡使用起来很灵活，因为可以任意读写且便于携带。存储卡有多种接口，包括 USB、MMC（MultiMediaCard，多媒体卡）、SD（Secure Digital，安全数码卡）和 CompactFlash。现在最大的存储卡容量是 32GB，但这个数字在不断上升。

## 4.5 测试网页

将网页传输到 Internet 站点、内部网服务器、磁盘或存储卡上后，应立即对每个网页进行测试。

下面的检查清单有助于确保网页正常工作。

1. 传输网页前，对硬盘上的网页进行下面所列的所有测试。将网页发布到母盘或 Web 服务器后，再次进行测试——如果网页发布到 Internet，可以使用 56kbit/s 的 modem 进行连接，在最小的连接速度下测试网页。
2. 使用最新版本的 Mozilla Firefox、Microsoft Internet Explorer 以及至少一个其他浏览器（如 Opera 或 Safari）执行下列步骤。使用旧版本的 Netscape Navigator 或 Internet Explorer 也是不错的方法，因此许多人仍在使用过时的版本，有些网页的显示可能会不一样。
3. 可能的话，使用 800×600 的分辨率进行测试。如果网页在这一分辨率下的显示效果还不错的话，它们在更高分辨率下的效果应该会更好。也可以用 1024×768 或 1600×1200 的分辨率再进行测试。我想不用使用 640×480 的分辨率进行测试，因为使用这样的分辨率的人应该很少。



4. 测试前关闭网页浏览器中的自动图像加载，因此这样可以看到不显示图像的网页是什么样的。检查<alt>标签显示的消息，然后启动图像加载，加载图像，再看看网页。第8章将介绍如何将图像放在网页中以及如何使用 alt 标签。

5. 使用浏览器的字体大小设置，用不同的字体大小浏览网页，确保精心设计的网页布局不会受到破坏。

6. 从主页开始，系统地检查每个连接是否可用（每次单击后使用“返回”按钮返回，然后单击下一个链接）。

7. 等到每个页面完全加载后，滚动页面，检查所有图像是否正常显示。

8. 如果网站很复杂，可以列出网站的所有网页，确保所有网页都经过测试。

9. 计算每个网页通过 56kbit/s modem 加载的时间，尤其当提供 Internet 连接的 ISP 与运行 Web 服务器的 ISP 不同时。网页上的信息是否有足够的价值，能够避免用户在页面完成加载前就离开该页面？当然，宽带连接越来越普遍了，但现在忽略拨号 Web 用户还有点为时太早。

如果网页通过了所有这些测试，就可以保证它们在每个访问者的机器上的显示效果都会很好。

## 4.6 总结

这一章介绍了发布网页的常用方法以及选择网页发布方法时所需的基础知识；还逐步教你学会通过常用的文件传输软件将网页上传到 Web 服务器；最后还提供了一个检查清单，帮助将网页放到适当的地方后彻底地测试网页。

## 4.7 问与答

**问：**从家里上传页面到网站时，一切正常；但在工作地点的计算机上尝试时，却得到错误信息，这可能是什么原因呢？

**答：**工作地点的计算机可能有防火墙，它是一个安全层，用于保护本地网络不受来自 Internet 的破坏。需要在 FTP 程序中调置一些特殊配置选项，使发送的文件可以顺利通过防火墙。至于配置细节，公司的网络管理员可以帮助你。

**问：**我不知道选择哪家 ISP——有那么多！

**答：**显然，可以对比一下 <http://thelist.internet.com> 列出的公司的价格。也可以要一些网站规模与你计划相同的用户的姓名；（通过电子邮件）询问这些用户对公司提供的服务与支持是否满意。另外，确保 ISP 最少有两个主要的 Internet 连接（T3 或更高），且连接最好是由两个不同的网络公司提供的。

**问：**你推荐的所有测试所需的时间超过了创建页面的时间！不能少进行一些测试吗？

**答：**如果创建页面的目的不是赚钱或提供重要服务，则网页对某些人来说比较可笑或偶尔产生错误，将不是大问题。对于这种情况，只需要通过不同的窗口和字体大小来测试每个页面，一天就能做完。然而，如果需要树立专业形象，那么除严格的测试外，别无

选择。

问：我想将我的网站命名为 lizardlover.com，但是已经有人比我先申请了。我该怎么办呢？

答：如果你经营一个爬虫类宠物店，店名为 Lizard Lover, Inc.，然后有人注册了这个域名，你可尝试联系他们来解决问题。也可以控告他们，但如果没有足够的预算来拿起法律武器，也可以注册 lizardlover.org、lizardlover.net 甚至 lizardlover.biz（如果这些域名尚未被抢注的话）。另外，很多域名注册公司允许以少量费用预定域名，也就是说他们将监视这个域名，并在它到期时自动为你申请这个域名。

## 4.8 作业

作业包含一些测试题及练习，帮助巩固对这一章介绍的内容的理解。试着回答所有问题，然后再看后面的“答案”小节。

### 4.8.1 测验

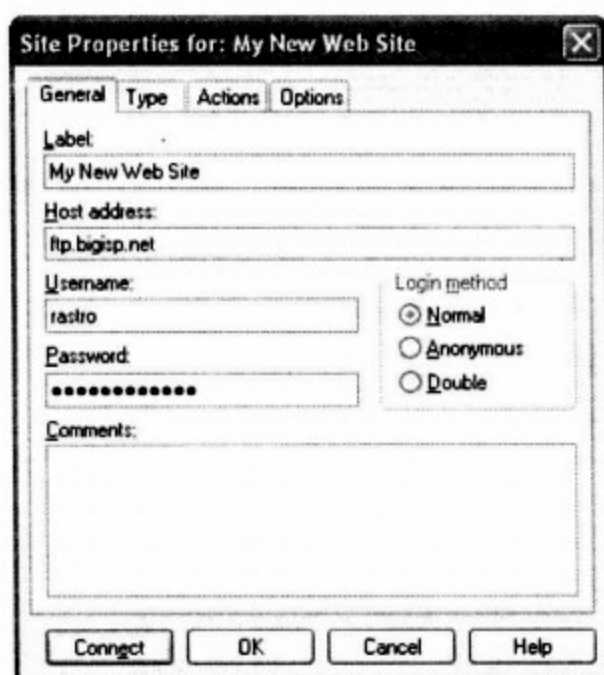
1. 如何将网页存放到闪存条、CD-ROM 或 CD-ROM 中？
2. 如果你的 ISP 要求将网页放到 ftp.bigisp.net 的默认主目录下，你的用户名是 rastro，密码是 rorry\_relroy。在 CuteFTP 中如何设置这些信息，从而将文件上传到 Internet？
3. 在 Internet Explorer 中输入什么地址，才可以看到测验 2 中上传的网页？
4. 如果下面的网页被命名为 mypage.html，要将其发布到 Internet，需要传递哪些文件到网络服务器？

```
<html>
  <head>
    <title>My Page</title>
  </head>
  <body background="joy.gif">
    
    <h1>My Web Page</h1>
    <p>Oh happy joy I have a page on the Web!</p>
    <a href="otherpage.html">Click here for my other page.</a>
  </body>
</html>
```

### 4.8.2 答案

1. 将含有 HTML 与图像文件的目录结构从硬盘复制到储存卡、CD 或 DVD 上。这样，任何人都可以将这些媒体插入计算机中，启动网络浏览器，并从储存卡或光盘中打开这些页面。
2. 单击 CuteFTP 工具栏的 New 按钮，然后输入下面的信息（如图 4.10 所示）：

图 4.10



3. 仅根据测验 2 的信息无法给出答案。可能的答案是 <http://www.bigisp.net/~elroy/>，但是也可以选择一个完全不同的域名，如 <http://elroy-and-astro.com/>。

4. 需要将下面三个文件都传递到网络服务器的同一目录下：

- mypage.html;
- joy.gif;
- me.jpg。

如果希望这个页面的链接也可以使用，就必须再传输下面这个文件及其引用的所有图像文件：otherpage.html。

## 4.9 练习

将页面放到 Internet 上！

## 第二部分

### 使用 HTML 创建实用网页

第 5 章 基本文本对齐和格式化

第 6 章 创建文本链接

第 7 章 创建网页图像

第 8 章 将图像放到网页中

第 9 章 自定义背景和颜色

第 10 章 图像链接和图像映射







## 第 5 章

# 基本文本对齐和格式化

在 Web 早期，所有文本都以相同的字体和大小显示。如果你使用计算机的时间足够长，了解在 mice 和图形操作系统出现前的历史，就知道我说的，因为那个时候的文本格式化仅限于按回车键开始一个新段落！HTML 可以控制文本的外观以及文本在网页中如何对齐与显示。本章首先介绍文本对齐和格式化的基础知识，最后介绍高级文本技巧。读者将学习在网页中使用粗体、斜体、上标、下标、下划线和删除线等各种格式的文本，以及如何选择字体和字号。

这一章将介绍的另一个重要主题是列表，列表是在 HTML 中列举多项的一种方式。由于列表在网页中很常见，因此 HTML 提供了标签，用于在每个列表项前自动缩进文本以及添加编号或项目符号。在这一章，除学习定义列表——用于在网页上缩进内容的一种简单方法，还将学习如何格式化带编号和项目符号的列表。

**注意：**在 HTML 中控制文本对齐和格式化有两种完全不同的方法。本章将介绍的是遵守 HTML 标准、采用级联样式表（Cascading Style Sheets, CSS）的方法。虽然到第 12 章才会介绍 CSS，但在这章读者将学习足够的信息，将样式应用于文本。还有一种本章一开始就将简要介绍的格式化文本的“旧”方法，由于它不符合 HTML 标准，因此没有必要花很多时间来介绍。不仅如此，CSS 的功能也比这种方法强大得多。

**By the Way**

### 练习

如果有文本需要缩进、居中或进行其他视觉处理以便更容易阅读，将可充分利用本章介绍的知识。

- 任何类型的大纲、演示文稿中的项目列表、编号的步骤、词汇表或数据库中的文本信息列表，都可作为工作对象。
- 使用任何文本都可以，但尽量使用要放到网页中的文本。公司宣传册或个人简历中的文本是不错的选择。

- 如果使用的文本来自字处理软件或数据库程序，将它存储为纯文本文件或 ASCII 文件。然后随着本章课程的学习，在文本中添加合适的 HTML 标签和样式属性，对文本进行格式化。
- 添加第 2 章介绍的标签<html>、<head>、<title>和<body>，然后使用本章介绍的代码对正文文本进行格式化。

## 5.1 新方法和旧方法

在早期的 HTML 版本中，文本的格式化使用特殊标签。例如，要将文本格式化为粗体，只要将它放在标签<b>和</b>之间。同样，<font>标签用于指定特定字体，包括字体、字号等。虽然这种格式化标签系统可用于基本的文本格式化，但这些标签散布于 HTML 语言中，使得内容和用于对内容进行格式化的代码难以区分。用标签来标记网页中的信息以及信息如何格式化的细节，是不好的设计。

解决方法是删除许多格式化标签，在网页格式化中使用独立的技术。这种独立的技术就是 CSS (Cascading Style Sheet, 级联样式表)。这种技术听起来吓人，但将其用于格式化文本并不难。实际上，CSS 让你能够进行许多比使用标签更有趣的文本格式化。然而，要澄清一点：并非所有旧标签都删除了。例如，<b>和<i>仍得到支持，被广泛用于将文本格式化为粗体和斜体。

虽然与使用许多标签相比，我更喜欢样式表，但也不能完全忽略标签。因此，这一章在进行基本的文本格式化时，将介绍使用标签和使用样式表这两种方法。本书第三部分将进一步探讨 CSS 规则。本章重点将放在标签和样式表这两种文本格式化方法上。

## 5.2 对齐网页中的文本

在 Web 上浏览信息时，很容易理所当然地认为段落都自动地左对齐。然而，有些时候可能选择将文本向右或居中对齐。HTML 允许指定 HTML 元素——如文本段落或网页的某些部分的对齐方式。在介绍这两种对齐方式前，先来复习一下属性是如何发挥作用的。

在第 3 章中，简单地介绍了如何使用属性为 HTML 标签提供额外信息。属性是 HTML 标签中的特殊代码，用于控制标签的作用。即使在最简单的网页中，属性也起着重要的作用，因此熟悉属性的用法很重要。举个例子，属性可用于决定段落的对齐方式。以<p>标签开始段落时，可以通过设置 style 属性来指定段落中的文本应与左边距、右边距对齐还是居中对齐。

**By the Way**

**注意：**在本课及其他部分中，你将学习到，style 属性实际上几乎可用于所有 HTML 格式化。

### 5.2.1 段落对齐

要使段落右对齐，只要将 `style="text-align:right"` 放在段落开头的 `<p>` 标签中；要使段落居中对齐，使用 `<p style="text-align:center">`。同样，使段落左对齐的代码是 `<p style="text-align:left">`。最后一种对齐几乎不会用到，因为当使用 `<p>` 标签时左对齐是默认设置。style 属性的 `text-align` 部分称为 style 规则，也就是说它设置 HTML 元素样式的特定方面。有许多样式规则可用于控制网页内容的格式化。

**注意：**如果没有设置属性，HTML 中的每个属性和样式规则都有默认值。对于 `<p>` 标签的 `text-align` 样式规则，默认值为 `left`，因此仅使用 `<p>` 标签相当于使用 `<p style="text-align:left">`。要成为优秀的网页开发人员，学习常用样式规则的默认值很重要。

**By the Way**

`text-align` 样式规则并非仅用于 `<p>` 标签中。实际上，几乎每个包含文本的 HTML 标签，包括 `<h1>`、`<h2>` 和其他标题标签以及下面将见到的一些标签，都可以使用该样式规则。除 `text-align` 样式规则外，还有许多其他样式规则。学习更多 HTML 标签后，将学习如何使用它们。

根据官方的 HTML 标准，标签和属性不区分大小写。然而，更精确的 XHTML 标准要求标签和属性必须小写，因此你的代码应使用小写以符合 XHTML 标准。XHTML 标准还要求使用引号将属性值括起。

例如，下面的代码从技术上说是大多数主流网页浏览器都能接受的：

```
<P STYLE=TEXT-ALIGN:CENTER>
```

但这行代码不符合网页设计的最新标准，因为它是大写的，且 `style` 属性值 `text-align:cernter` 没有用引号括起。要与将来的标准和软件保持兼容，应使用下面的代码：

```
<p style="text-align:center">
```

**注意：**有时候同样的样式规则用于不同的标签时会有不同的含义。例如，你在第 8 章将发现，`style="text-align:left"` 用于图像标签 `<img>` 与用于本章讨论的文本标签的效果是不一样的。

**By the Way**

需要指出的是，在 CSS 样式之前，HTML 内容使用特殊标签 `<align>` 进行对齐。`<align>` 标签的功能与 `text-align` 样式规则很像，但在 W3C 定义的 XHTML 中，这个标签已不再使用。大多数浏览器仍支持这个标签，但最好不要使用它。

### 5.2.2 对齐网页的整个部分

如果希望一次对齐的内容不只一个段落或标题，可使用 `<div>` 标签和 `text-align` 样式规则。

`<div>`标签和对应的结束标签`</div>`本身实际上并没有什么功能，这使它们看起来似乎是没有用的标签。

然而，如果在其中包含 `text-align` 样式规则，`<div>`就变得很有用。例如，`<div style="text-align:center">`和`</div>`之间的所有元素都居中对齐。这可以是文本行、段落、标题、图像以及其他接下来将学习如何放在网页中的元素。同样，`<div style="text-align:right">`将到下一个`</div>`标签之间的所有内容都右对齐。

**By the  
Way**

**注意：**在本书稍后你将知道，`<div>`标签也可用于使用特殊格式化样式来组织文本。

程序清单 5.1 演示了`<p>`标签和`<div>`标签的 `style` 属性及其 `text-align` 样式规则。结果如图 5.1 所示。你将在第三部分中学习`<div>`标签的许多高级用法。

#### 程序清单 5.1 style 属性的 text-align 样式规则

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Bohemia</title>
  </head>

  <body>
    <div style="text-align:center">
      <h1>Bohemia</h1>
      <h3>by Dorothy Parker</h3>
    </div>
    <p style="text-align:left">
      Authors and actors and artists and such<br />
      Never know nothing, and never know much.<br />
      Sculptors and singers and those of their kidney<br />
      Tell their affairs from Seattle to Sydney.
    </p>
    <p style="text-align:center">
      Playwrights and poets and such horses' necks<br />
      Start off from anywhere, end up at sex.<br />
      Diarists, critics, and similar roe<br />
      Never say nothing, and never say no.
    </p>
    <p style="text-align:right">
      People Who Do Things exceed my endurance;<br />
      God, for a man that solicits insurance!
    </p>
  </body>
</html>
```

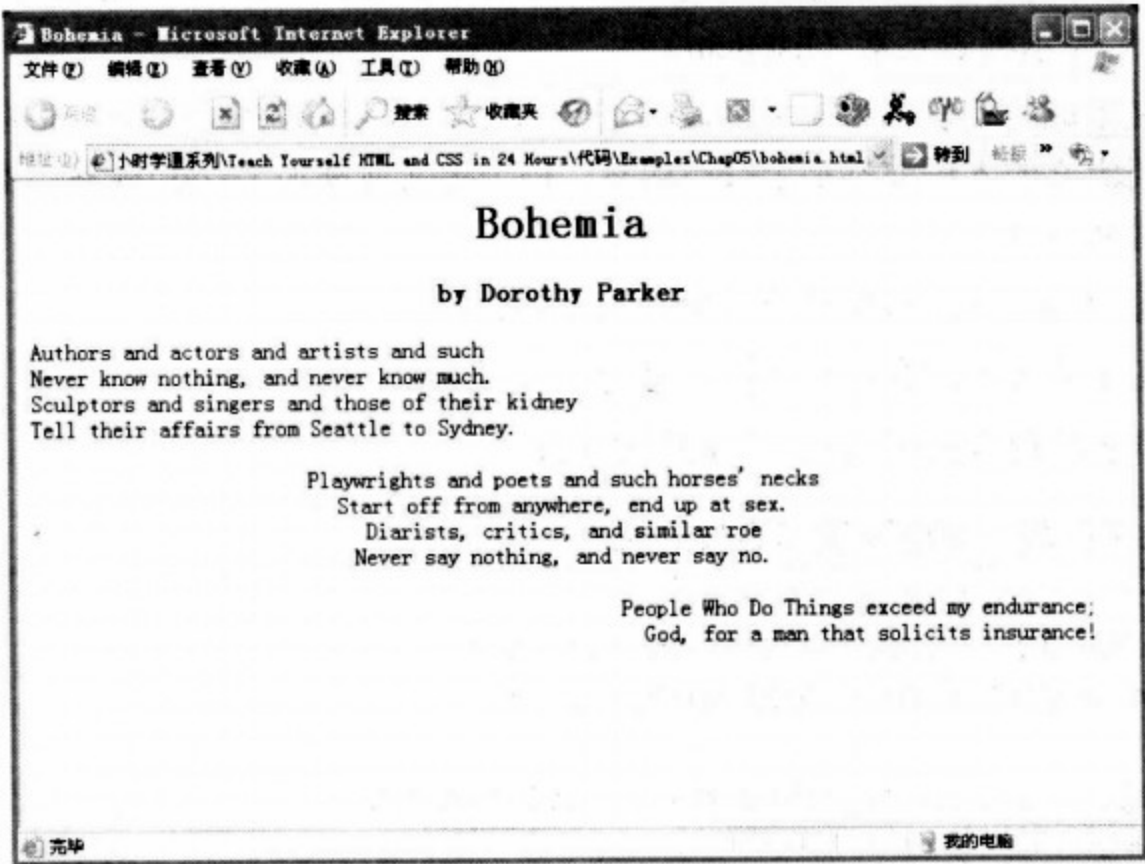


图 5.1  
程序清单 5.1 指定的对齐设置在网页浏览器中的显示结果

### 5.3 三种 HTML 列表

将网页上的信息表示为列表 1 通常很有用。有三种基本的 HTML 列表。图 5.2 显示了这三种列表，程序清单 5.2 列出了创建这些列表的 HTML。

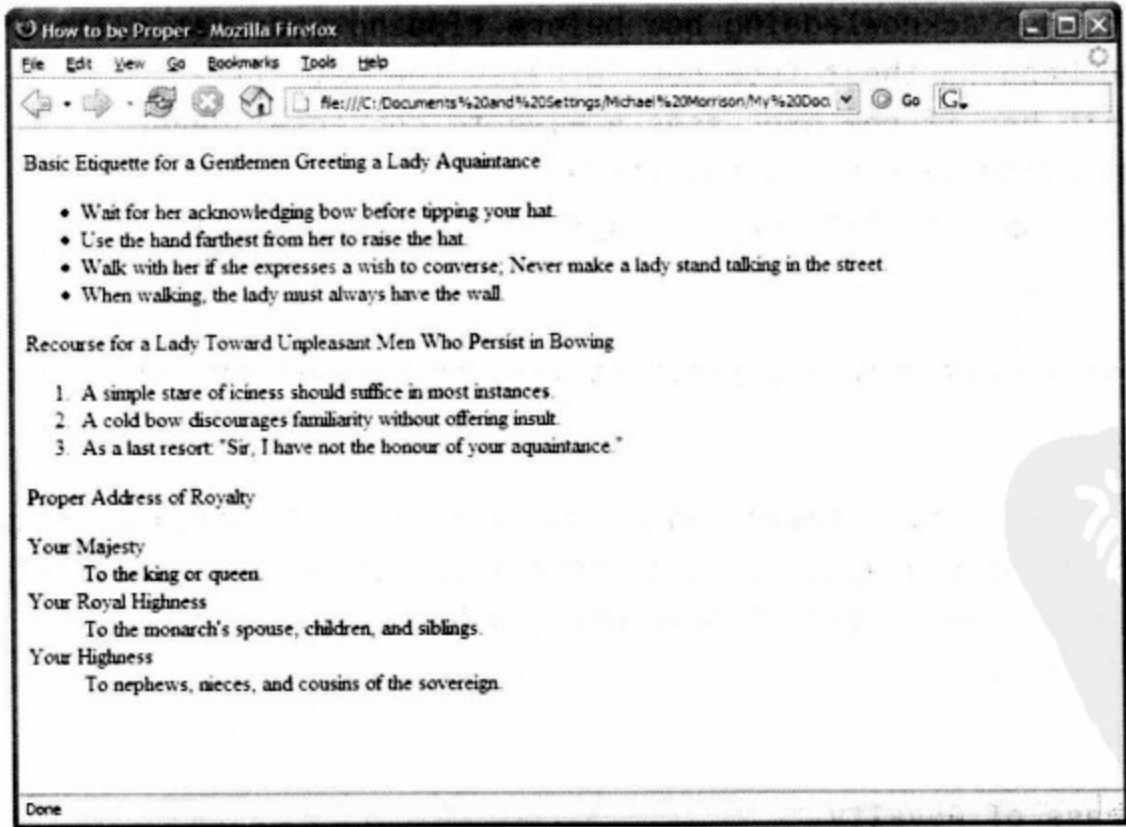


图 5.2  
三种 HTML 列表在 Mozilla Firefox 的最新版本中的显示

- 带编号的列表称为有序列表。有序列表以<ol>标签开始，以</ol>标签结束。在每个<li>标签处都自动添加编号和分行，且每行都缩进。
- 带项目符号的列表称为无序列表。无序列表以<ul>标签开始，以</ul>标签结束。无



序列表与有序列表类似，只是每个<li>标签处都自动添加项目符号而不是编号。<li>标签中的“li”表示列表元素（list item）。

- 术语及其定义组成的列表称为定义列表。定义列表以<dl>标签开始，以</dl>标签结束。在每个要定义的术语前都加上<dt>标签，每个定义前都加上<dd>标签。术语和定义自动缩进和分行。
- 有序列表是每个列表项前有编号或字母的缩进列表。
- 无序列表是每个列表项前有特殊项目符号的缩进列表。
- 定义列表是列表项前没有编号或符号的缩进列表。

#### 程序清单 5.2 无序列表、有序列表和定义列表

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>How to be Proper</title>
  </head>

  <body>
    <p>
      Basic Etiquette for a Gentlemen Greeting a Lady Acquaintance
    </p>
    <ul>
      <li>Wait for her acknowledging bow before tipping your hat.</li>
      <li>Use the hand farthest from her to raise the hat.</li>
      <li>Walk with her if she expresses a wish to converse; Never
        make a lady stand talking in the street.</li>
      <li>When walking, the lady must always have the wall.</li>
    </ul>
    <p>
      Recourse for a Lady Toward Unpleasant Men Who Persist in Bowing
    </p>
    <ol>
      <li>A simple stare of iciness should suffice in most instances.</li>
      <li>A cold bow discourages familiarity without offering insult.</li>
      <li>As a last resort: "Sir, I have not the honour of your
        acquaintance."</li>
    </ol>
    <p>
      Proper Address of Royalty
    </p>
    <dl>
      <dt>Your Majesty</dt>
      <dd>To the king or queen.</dd>
    </dl>
  </body>
</html>
```

```
<dt>Your Royal Highness</dt>
<dd>To the monarch' s spouse, children, and siblings.</dd>
<dt>Your Highness</dt>
<dd>To nephews, nieces, and cousins of the sovereign.</dd>
</dl>
</body>
</html>
```

注意：网页在不同网页浏览器中的显示可能很不一样。HTML 标准没有指定网页浏览器应如何格式化列表，因此使用旧网页浏览器的用户看到的缩进可能与你看到的不同。虽然现在所有网页浏览器对列表的显示几乎都是一样的，将来的软件对 HTML 列表的格式化可能也会不一样。

By the Way

## 5.4 将列表放在列表中

虽然设计定义列表的本意是要用于定义术语，但许多网页制作者在需要缩进的任何地方都使用定义列表。实际上，可以在需要缩进的文本前面加上<dl><dd>、在文本结束处加上</dl></dd>。然而，还有一种更好的方法，就是使用<blockquote>标签，该标签使内容缩进而不将其视为定义。

也可以通过在列表中嵌套另一个列表，从而进一步缩进列表项，如下所示：

```
<dl>
  <dd>this item will be indented</dd>
  <dl>
    <dd>this will be indented further</dd>
    <dl>
      <dl>
        <dd>and this will be indented very far indeed</dd>
      </dl>
    </dl>
  </dl>
</dl>
```

注意：嵌套是指一个标签完全出现在另一个标签内，如在上个示例中，<dl> 标签位于<dd>标签内。包含其他标签的标签称为父标签，被嵌套在内的标签称为子标签。通常（但不是必须）嵌套的标签要缩进，这样容易看出它们与父标签之间的关系。

By the Way

确保每个开始标签<dl>都有一个对应的结束标签</dl>。缩进代码使开始标签和结束标签对齐，有助于检查定义列表的标签是否配对。

有序列表和无序列表也可以互相嵌套，嵌套的级别可以很深。在程序清单 5.3 中，几个无序列表组成了一个复杂的缩进大纲。从图 5.3 可以看到，Firefox 自动为前三个级别的缩进

使用不同类型的项目符号，使列表更具可读性。

程序清单 5.3 使用列表来构造大纲

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Vertebrates</title>
  </head>

  <body>
    <h2>Vertebrates</h2>
    <ul>
      <li>
        Fish
        <ul>
          <li>Barramundi</li>
          <li>Kissing Gourami</li>
          <li>Mummichog</li>
        </ul>
      </li>
      <li>
        Amphibians
        <ul>
          <li>
            Anura
            <ul>
              <li>Goliath Frog</li>
              <li>Poison Dart Frog</li>
              <li>Purple Frog</li>
            </ul>
          </li>
          <li>
            Caudata
            <ul>
              <li>Hellbender</li>
              <li>Mudpuppy</li>
            </ul>
          </li>
        </ul>
      </li>
      <li>
        Reptiles
        <ul>
          <li>Nile Crocodile</li>
        </ul>
      </li>
    </ul>
  </body>
</html>
```

```
<li>King Cobra</li>
<li>Common Snapping Turtle</li>
</ul>
</li>
</ul>
</body>
</html>
```

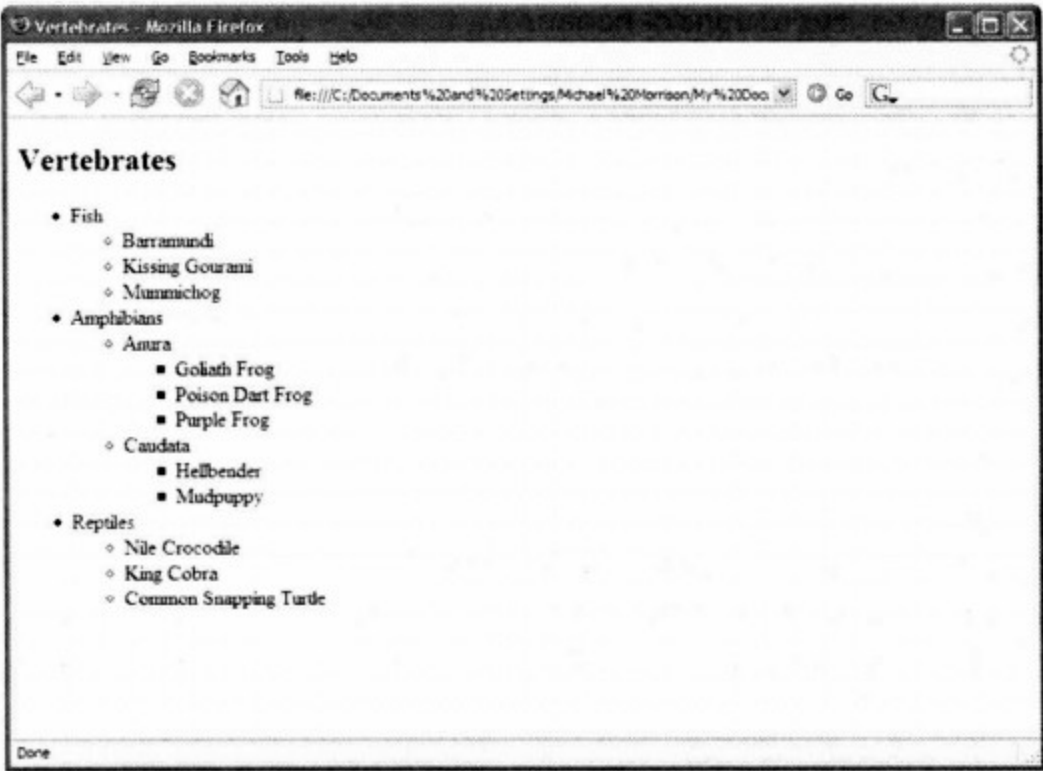


图 5.3  
多层列表使用整齐的缩进和不同的项目符号，更具可读性

对于图 5.3 所示的定义列表，网页浏览器通常都使用实心圆圈作为第一级项目符号，第二级项目符号使用空心圆圈，更深的级别将使用实心方框。然而，可以使用`<ul style="list-style-type:disc">`、`<ul style="liststyle-type:circle">`或`<ul style="list-style-type:square">`来代替`<ul>`，从而显式地指定每个级别要使用的项目符号类型。

甚至可以在无序列表的`<li>`标签内使用 `list-style-type` 样式规则，为单个列表项修改项目符号。例如，下列代码将在 `extra` 和 `super` 之前显示空心圆圈，而在 `special` 之前显示实心方框：

```
<ul style="list-style-type:circle">
  <li>extra</li>
  <li>super</li>
  <li style="list-style-type:square">special</li>
</ul>
```

`list-style-type` 样式规则也可用于有序列表，但在有序列表中，不是选择项目符号类型，而是选择每个列表项之前的编号或字母类型。程序清单 5.4 显示了如何在多级列表中使用罗马数字（`list-style-type:upper-roman`）、大写字母（`list-style-type:upper-alpha`）、小写字母（`list-style-type:lower-alpha`）和普通数字。图 5.4 显示了大纲的格式化结果。

程序清单 5.4 在多级列表内使用 style 属性的 list-style-type 样式规则

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
```

```

"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Advice from the Golf Guru</title>
  </head>

  <body>
    <h2>How to Win at Golf</h2>
    <ol style="list-style-type:upper-roman">
      <li>Training
        <ol>
          <li>Mental prep
            <ol style="list-style-type:upper-alpha">
              <li>Watch PGA on TV religiously</li>
              <li>Get that computer game with Jack whatsisname</li>
              <li>Rent "personal victory" subliminal tapes</li>
            </ol>
          </li>
          <li>Equipage
            <ol style="list-style-type:upper-alpha">
              <li>Make sure your putter has a pro autograph on it</li>
              <li>Pick up a bargain bag of tees-n-balls at Costco</li>
            </ol>
          </li>
          <li>Diet
            <ol style="list-style-type:upper-alpha">
              <li>Avoid baseball or football food
                <ol style="list-style-type:lower-alpha">
                  <li>No hotdogs</li>
                  <li>No pretzels</li>
                  <li>No peanuts and Crackerjacks</li>
                </ol>
              </li>
              <li>Drink cheap white wine only, no beer</li>
            </ol>
          </li>
        </ol>
      </li>
      <li>Pre-game
        <ol>
          <li>Dress
            <ol style="list-style-type:upper-alpha">
              <li>Put on shorts, even if it's freezing</li>
              <li>Buy a new hat if you lost last time</li>
            </ol>
          <li>definition lists
          </li>
          <li>Location and Scheduling

```



```

<ol style="list-style-type:upper-alpha">
  <li>Select a course where your spouse won' t find you</li>
  <li>To save on fees, play where your buddy works</li>
</ol>
</li>
<li>Opponent
  <ol style="list-style-type:upper-alpha">
    <li>Look for: overconfidence, inexperience</li>
    <li>Shun: suntan, stethoscope, strident walk, Florida accent</li>
    <li>Buy opponent as many pre-game drinks as possible</li>
  </ol>
</li>
</ol>
</li>
<li>On the Course
  <ol>
    <li>Tee first, then develop severe hayfever</li>
    <li>Drive cart over opponent' s ball to degrade aerodynamics</li>
    <li>Say "fore" just before ball makes contact with opponent</li>
    <li>Always replace divots when putting</li>
    <li>Water cooler holes are a good time to correct any errors in ball
placement</li>
    <li>Never record strokes taken when opponent is urinating</li>
  </ol>
</li>
</ol>
</body>
</html> definition lists

```

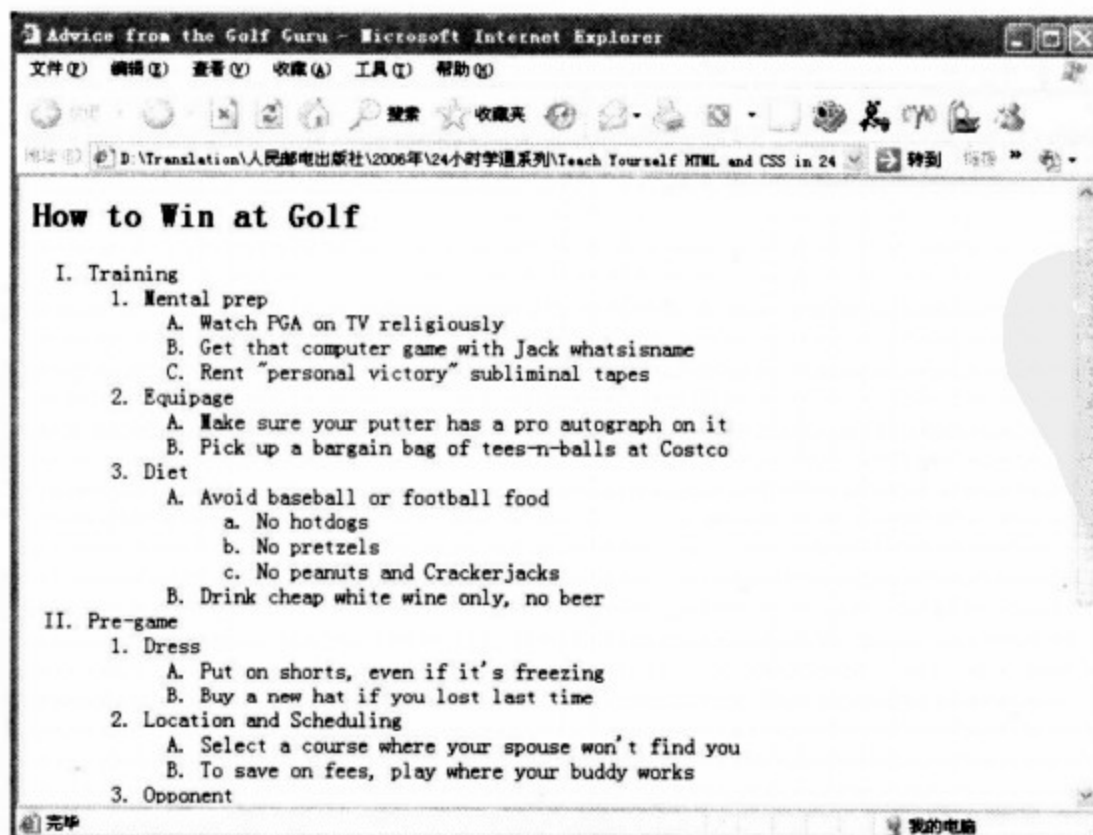


图 5.4  
格式良好的大纲  
使计划锦上添花

虽然程序清单 5.4 只在<ol>标签中使用 list-style-type 样式规则，你也可以在列表中特定的<li>标签中使用这个规则（但是很少有需要这样做的情况）。也可以用 list-style-type:decimal 显式地指定使用普通数字，list-style-type:lower-roman 指定使用小写罗马数字。

### By the Way

**注意：**通过有序列表、无序列表和定义列表的组合，可以将最复杂的信息以易读、有吸引力的方式组织起来。为触发你的灵感，我构造了一些列表，可以在组织自己的 HTML 列表前浏览一下。

详情请参阅 Sams Publishing 网站 <http://www.sampublishing.com/>。

## 5.5 粗体、斜体和特殊的文本格式化

在打字机时代，通常只使用纯文本，偶尔使用下划线突出内容。现在，粗体和斜体在书面通信中很普遍。通常也可以在网页上添加粗体和斜体文本。有几个标签和样式规则可用于文本格式化。

为文本添加粗体和斜体格式可使用标签<b>和<i>。对于粗体文本，将<b>标签放在文本前面、将</b>标签放在文本结尾。同样，将文本放在<i>和</i>标签之间，则文本被格式化为斜体。虽然这种方法仍可用于浏览器中，XHTML 也支持，但它不如使用 CSS 样式规则来格式化文本的新方法灵活和强大。

可以将一种格式嵌入在另一种格式中。例如，要将文本格式化为粗体和斜体，将<b><i>放在文本前面，将</b></i>放在文本结尾。为避免给有些网页浏览器造成困扰，要记得按照与开始标签相反的顺序加上结束标签。例如，不要使用如下的代码：

```
<b>Bold, <i>bold and elegant,</b> or just plain elegant.</i>
```

而应该用如下代码：

```
<b>Bold, <i>bold and elegant,</i></b><i>or just plain elegant.</i>
```

虽然你在第三部分将学到更多 CSS 样式规则，现在应提前学习一点，这样才能理解文本格式化的样式规则。font-weight 样式规则让你能够使用样式规则来设置字体的粗细。标准设置包括 normal、bold、bolder 和 lighter，其中 normal 是默认设置。斜体文本通过 font-style 规则来控制，该规则可设置为 normal、italic 或 oblique。如果要应用多种格式，可指定多个样式规则，如下面的示例所示：

```
<p style="font-weight:bold; font-style:italic">This paragraph is bold and italic!</p>
```

在这个示例中，在<p>标签的 style 属性中指定了两个样式规则。使用多个样式规则的关键在于它们必须使用分号分开。

文本样式不仅可用于段落。下列代码显示了如何将带项目符号的列表中的文本格式化为斜体：

```
<ul>
```

```
<li style="font-style:italic">Important Stuff</li>
<li style="font-style:italic">Critical Information</li>
<li style="font-style:italic">Highly Sensitive Material</li>
<li>Nothing All That Useful</li>
</ul>
```

不使用样式规则而使用传统的**<b>**和*<i>*标签，也可以构造出同样的列表。在<li>列表项或标题中，<b>、<i>或其他格式化标签应在</li>之前有配对的结束标签。在下面列表中，大多数<i>和</i>标签看起来可能是多余的，但遵守这种形式可确保网页完全符合新的 XHTML 标准，这样将来可能节省重写网页的时间：

```
<ul>
  <li><i>Important Stuff</i></li>
  <li><i>Critical Information</i></li>
  <li><i>Highly Sensitive Material</i></li>
  <li>Nothing All That Useful</li>
</ul>
```

也可以在标题中使用<b>标签和 font-weight 样式规则，但由于标题已经是粗体，因此更重的字体并没有效果。

**注意：**格式化粗体和斜体时，还有一种方法可以代替样式规则，那就是<strong>标签和<em>标签。在大多数浏览器中，<strong>标签的作用与<b>标签相同，而<em>标签的作用与<i>标签相同。

有人认为，标签<strong>和<em>是对标签<b>和<i>的改进，因为前两者只指出文本应突出，而没有指定突出效果如何实现。换句话说，浏览器不一定将<strong>解释为粗体、将<em>解释为斜体。这使得<strong>和<em>更适合于 XHTML，因为它们使文本增加了额外的意义，并影响文本的显示。即使如此，只要<b>和<i>这两个标签仍得到支持，使用它们也不会有问题。

By the Way

除<b>、<i>、<strong>和<em>，还有其他 HTML 标签可用于对文本进行格式化。表 5.1 对这些标签进行了总结（包括粗体和斜体标签），程序清单 5.5 和图 5.5 演示了每个标签的用法。

表 5.1 为文本增加特殊格式的 HTML 标签	
标签	功能
<small>	小字体。
<big>	大字体。
<sup>	上标。
<sub>	下标。
<em>或<i>	强调的（斜体）文本。
<strong>或<b>	粗体。
<tt>	等宽字体（打字机字体）。
<pre>	等宽字体，保留空格和分行。

Watch Out

**警告：**原来还有<u>标签，用于定义带下划线的文本，但现在由于两个原因不再使用。首先，人们会以为带下划线的文本是链接，如果遇到不是链接的下划线字体将对用户造成困惑。第二，<u>标签已被摒弃，也就是说HTML/XHTML语言中已不再使用它。网页浏览器现在还支持<u>标签以及用于定义带删除线文本的<strike>标签，在未来的一段时间内也还会支持，但是定义带下划线或删除线的文本最好使用样式表。这些问题将在第12章和第13章更详细地介绍。

程序清单 5.5 特殊的格式化标签

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Miracle Product</title>
  </head>

  <body>
    <p>
      New <sup>Super</sup><strong>Strength</strong> H<sub>2</sub>O
      <em>plus</em> will knock out any stain, <big>big</big> or
      <small>small</small>.<br /> Look for new
      <sup>Super</sup><b>Strength</b> H<sub>2</sub>O <i>plus</i>
      in a stream near you.
    </p>
    <p>
      <tt>NUTRITION INFORMATION</tt> (void where prohibited)
    </p>
    <pre>
              Calories   Grams   USRDA
              /Serving   of Fat   Moisture
Regular          3         4       100%
Unleaded         3         2       100%
Organic          2         3        99%
Sugar Free       0         1       110%
    </pre>
  </body>
</html>
```

<tt>标签通常将字体修改为 Courier New，这是一种等宽字体（等宽表示所有字母和空格的宽度都是一样的）。然而，网页浏览器允许用户将等宽的<tt>字体修改为他们选择的字体（在Microsoft Internet Explorer中，选择“工具/Internet选项/字体”，在Mozilla Firefox中，选择“工具/选项/常规/字体和颜色”）。等宽字体对于有些用户来说甚至可能不一定是等宽，虽然大

多数人将使用浏览器设置的标准字体。

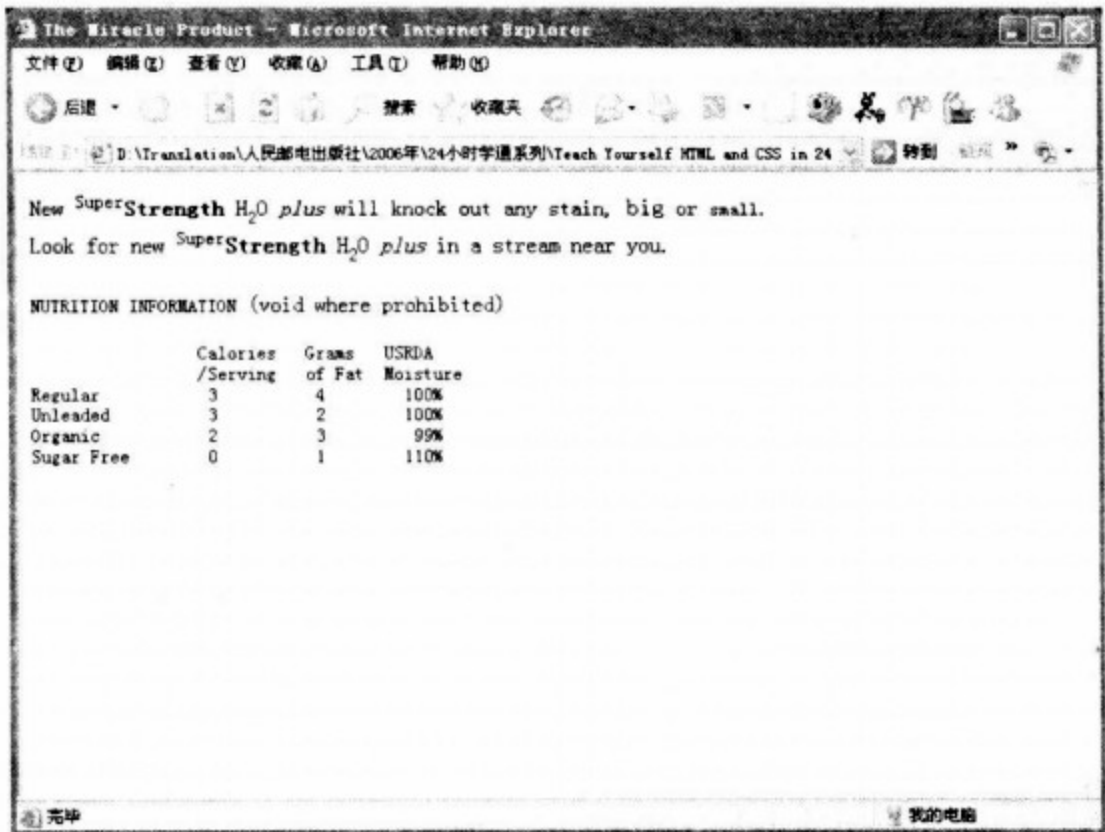


图 5.5  
程序清单 5.5 中的  
格式化字符的显  
示结果

<pre>将文本格式化为等宽字体，但它还有其他一些独特而有用的功能。第 2 章提到，HTML 文件中的多个空格符和分行通常被忽略，但<pre>使空格符和分行被保留。例如，如果没有<pre>，图 5.5 中最后的文本将如下所示：

```
calories grams usrda /serving of fat moisture regular
3 4 100% unleaded 3 2 100% organic 2 3 99% sugar free 0 1 110%
```

即使在每行最后加上<br />标签，各列也不会对齐。然而，如果在前面加上<pre>、在最后加上</pre>，则各列将整齐排列，因为原来的空格被保留——不需要<br />标签。

将文本分列还有更好的方法，你将在第三部分学到。<pre>标签是一种简便的方法，让你能够保留等宽文本文件中的对齐排列。

**注意：**可以使用<pre>标签在段落之间插入额外的竖直间隔。例如，要在“up”和“down”之间插入几个空行，可以输入：

```
up<pre>
</pre>down
```

By the Way

## 5.6 调整字体

<big>、<small>和<tt>标签让你能够对网页上文本的大小和外观进行一些基本控制。然而，有时候可能需要进行更多的控制。在介绍对 XHTML 代码中的文本进行格式化的方法前，先来看看在 CSS 之前是如何实现的。



在样式表出现之前，用<font>标签来控制网页中文本的字体。例如，下面的 HTML 代码将改变网页中一些文本的大小和颜色：

```
<font size="5" color="purple">this text will be big and purple.</font>
```

可以看到，<font>标签的属性 size 和 color 可以修改文本的字体。虽然这种方法可行，但它被一种更好的文本格式化方法取代，这就是 CSS 样式规则。下面是一些用于控制字体的主要样式规则：

- font-family: 设置字体。
- font-size: 设置字号。
- color: 设置字体颜色。

### By the Way

**注意：**你将在第 9 章学习更多关于文本颜色的控制。第 9 章还将介绍如何创建自定义的颜色以及控制文本链接的颜色。

font-family 样式规则让你能够设置用于显示文本的字体。通常，这个样式可以也应该指定多个值（用逗号分开），这样，如果用户系统中没有第一种字体，浏览器可以使用其他字体。这很重要，因为每个用户在常用的基本字体（Arial、Times 等）组成的核心字体集外，安装的字体可能不一样。提供多种字体供选择，当不能使用理想字体时，还可以退而求其次，使用指定的其他字体。下面是一个使用 font-family 设置文本段落字体的示例：

```
<p style="font-family:arial, sans-serif, 'times roman'"
```

这个示例中有几个值得注意的地方。首先，arial 是指定的首要字符。指定字体时，大小写并不重要，因此，arial 与 Arial 或 ARIAL 相同。第二个地方是字体名 times roman 使用了单引号，因为该名称中包含空格。

样式规则 font-size 和 color 也常用于控制字号和颜色。font-size 可设置为预定义的大小，如 small、medium 或 large，也可设置为具体的点数，如 12pt 或 14pt。color 样式可设置为预定义的颜色，如 white、black、blue、red 或 green，也可设置为具体的十六进制颜色，如 #FFB499。下面的示例使用了前一个段落示例，指定了 font-size 和 color：

```
<p style="font-family:arial, sans-serif, 'times roman'; font-size:14pt; color:green">
```

### By the Way

**注意：**第 9 章将介绍什么是十六进制颜色。现在，只需知道它可用于指定精确的颜色而不只是 green、blue 或 orange 等。

程序清单 5.6 和图 5.6 所示的网页使用了这些样式规则来改进前面的曲棍球队员示例，进行了一些文本格式化。队员的姓名和球衣号码以更精确的字体显示，实际上仍与原来的<h1>标签相似。更重要的是，队员的其他信息以海军蓝显示，每项信息的描述都以粗体显示。

## 程序清单 5.6 使用字体样式规则对曲棍球队员网页进行改进

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Music City Mafia - Terry Lancaster</title>
  </head>

  <body>
    <p style="font-family:verdana, arial; font-size:18pt; font-weight:bold">16
    - Terry Lancaster</p>
    <p style="font-family:verdana, arial; font-size:12pt; color:navy">
      <br />
      <span style="font-weight:bold">Nickname:</span> Big T<br />
      <span style="font-weight:bold">Position:</span> RW<br />
      <span style="font-weight:bold">Height:</span> 6' 3"<br />
      <span style="font-weight:bold">Weight:</span> 195<br />
      <span style="font-weight:bold">Shoots:</span> Left<br />
      <span style="font-weight:bold">Age:</span> 40<br />
      <span style="font-weight:bold">Birthplace:</span> Nashville, TN
    </p>
    <hr />
    <p style="font-family:verdana, arial; font-size:12pt; color:navy">
      <span style="font-weight:bold">Favorite NHL Player:</span> Brett
      Hull<br />
      <span style="font-weight:bold">Favorite NHL Team:</span> Nashville
      Predators<br />
      <span style="font-weight:bold">Favorite Southern Fixin:</span> Skillet
      Fried Potatoes<br />
      <span style="font-weight:bold">Favorite Meat and Three:</span>
      Swett's<br />
      <span style="font-weight:bold">Favorite Country Star:</span>
      Patsy Cline<br />
      <span style="font-weight:bold">Favorite Mafia Moment:</span> "Chet
      finishing a game with his eyelid completely
      slashed through."
    </p>
  </body>
</html>

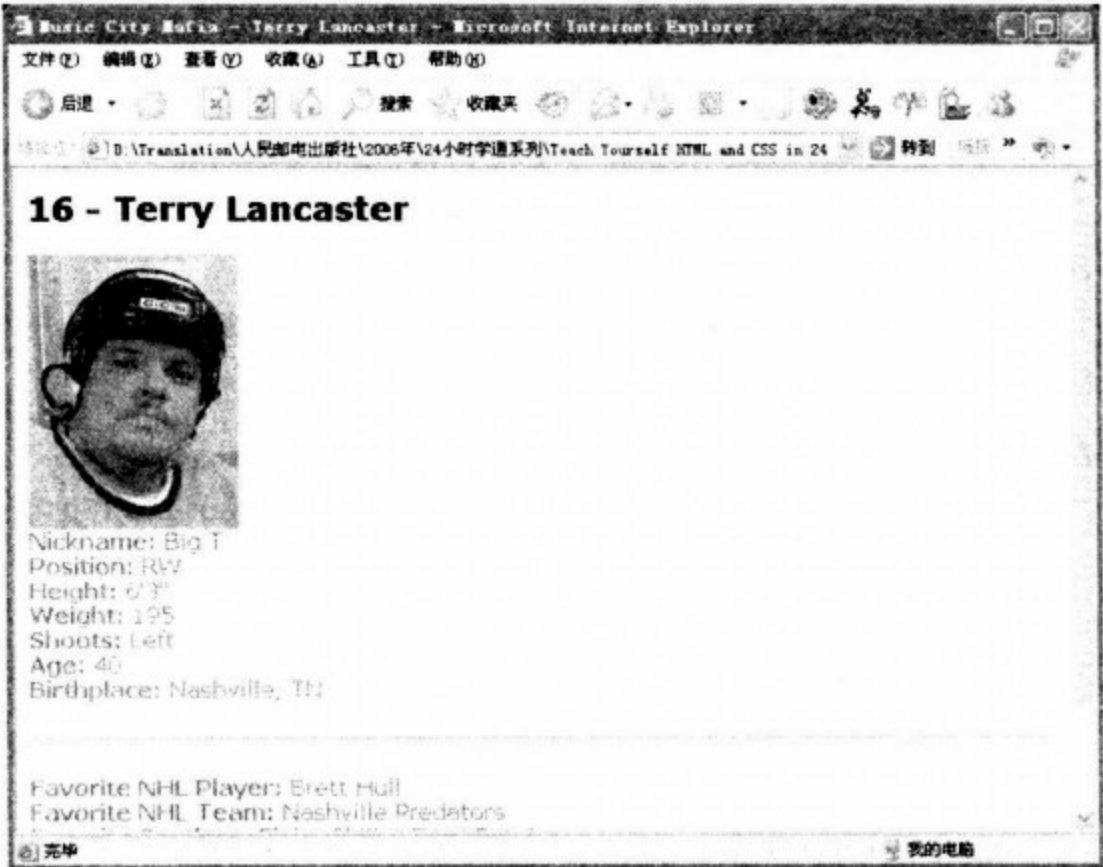
```

**注意：**你可能想知道这个示例代码中重复出现的<span>标签。这个标签让你能够将一组样式应用于一行中特定的内容，也就是说你可以在一个段落中隔离出单个的单词或句子，而不是像使用<div>标签或<p>标签那样将样式应用于整块文本。你将在第 12 章学到<span>标签的更多细节。现在，只需知道它让你能够分隔出一些文本以使用特定字体。

**By the  
Way**

图 5.6

如果你的计算机上安装了 Verdana，程序清单 5.6 所示的网页将以这种字体显示



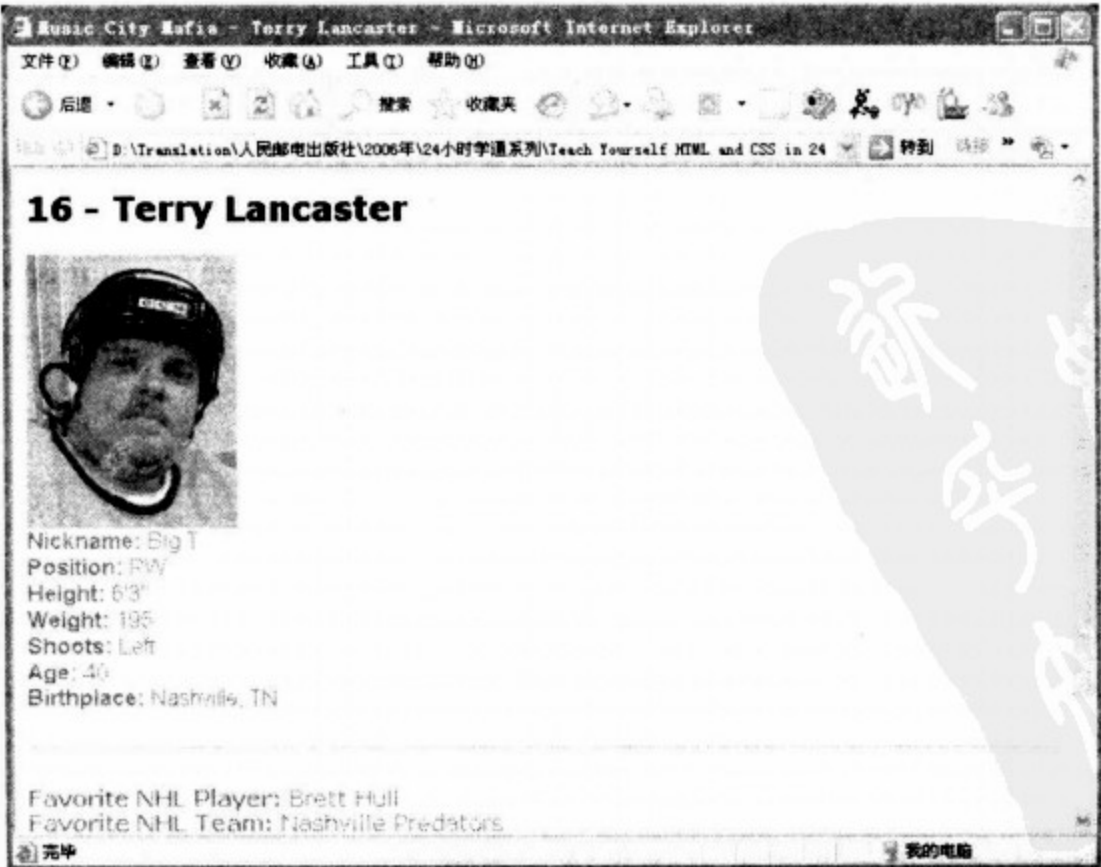
下面是程序清单 5.6 中为队员的姓名和球衣号码设置字体的代码：

```
<p style="font-family: verdana, arial; font-size:18pt; font-weight:bold">
```

如果用户系统中的网页浏览器可以找到 Verdana 字体，浏览器将使用这种字体；否则，浏览器将寻找是否有 Arial 字体。图 5.7 显示了网页在没有安装 Verdana 字体但有 Arial 字体的计算机上的显示结果。如果这两种字体都没有安装，浏览器将使用默认字体（通常是 Times New Roman）来显示。

图 5.7

如果没有安装 Verdana 字体，程序清单 5.6 中的文本将显示为 Arial 或 Times New Roman 字体



因为只有用户系统中安装的字体才能显示，因此你实际上并不能控制网页中文本以什么字体显示。还有一点，正确拼写字体名很重要，许多常用字体的名称很类似。这意味着，除 Times New Roman 外，惟一可靠的字体选择是 Arial（在 Windows 机器上）和 Helvetica（在 Macintosh 机器上）。不要不敢指定其他字体，但要保证你的网页用 Times New Roman 显示也可接受。

**注意：**要查看最常用的 Web 字体，并看看你的计算机上安装了其中的哪些字体，可访问 Sams Publishing 网站 <http://www.sampublishing.com/>。

**By the way**

### 5.7 使用特殊字符

现在大多数字体都包括欧洲语言的特殊字符，如 Café 中的重音符号 é。还有一些数学符号和特殊的标点符号，如项目符号 •。

可以在 HTML 文档中任意处插入这些特殊符号，具体用法请参见表 5.2。更完整的列表可参见 [http://www.Webstandards.org/learn/reference/named\\_entities.html](http://www.Webstandards.org/learn/reference/named_entities.html)。

例如，要插入 café，应使用如下代码：

```
caf&#233;
```

HTML/XHTML 使用特殊代码（称为字符实体）来表示特殊字符，如©和®。字符实体总是以“&”开始、以“;”结束。表 5.2 列出了一些常用的字符实体，HTML 支持的字符实体远不止这些。

**提示：**想知道版权符号©和注册商标符号®的代码？这两个符号的代码分别是&copy;和&reg;。要创建没有注册的商标符号™，使用<sup>tm</sup>，要用小一点的符号可使用<small><sup>tm</sup></small>。

**Did you know?**

表 5.2 英语中常用的特殊字符

字符	数字代码	代码名称	说明
“	&#34;	&quot;	引号
&	&#38;	&amp;	和号
<	&#60;	&lt;	小于号
>	&#62;	&gt;	大于号
¢	&#162;	&cent;	分号（货币）
£	&#163;	&pound;	英镑符号
	&#166;	&brvbar; 或 &brkbar;	断开的竖直线
§	&#167;	&sect;	节号
©	&#169;	&copy;	版权
®	&#174;	&reg;	注册商标
°	&#176;	&deg;	度数符号
±	&#177;	&plusmn;	加或减
²	&#178;	&sup2;	上标 2

续表

字符	数字代码	代码名称	说明
3	&#179;	&sup3;	上标 3
.	&#183;	&middot;	居中的点
1	&#185;	&sup1;	上标 1
1/4	&#188;	&frac14;	分数四分之一
1/2	&#189;	&frac12;	分数二分之一
3/4	&#190;	&frac34;	分数四分之三
Æ	&#198;	&AElig;	大写的 AE 组合
æ	&#230;	&aelig;	小写的 ae 组合
É	&#201;	&Eacute;	重音大写 E
é	&#233;	&eacute;	重音小写 e
×	&#215;	&times;	乘号
÷	&#247;	&divide;	除号

虽然可用数字来指定字符实体，但每个符号都有一个代记名称，通常更容易记。下面是使用单词 `café` 的一种方法：

```
caf&eacute;
```

注意，表 5.2 还有尖括号、引号及和号的代码。如果要在网页中使用这些符号，要使用它们的代码；否则网页浏览器将这些符号解释为 HTML 命令。

程序清单 5.7 和图 5.8 显示了表 5.2 中一些符号的用法。

程序清单 5.7 特殊符号的代码

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Punctuation Lines</title>
  </head>

  <body>
    <p>
      Q: What should you do when a British banker picks a fight with you?<br />
      A: &pound; some &cent;&cent; into him.
    <hr />
    Q: What do you call it when a judge takes part of a law off the
      books?<br />
      A: &sect; violence.
    <hr />
    Q: What did the football coach get from the locker room vending machine
      in the middle of the game?<br />
      A: A &frac14; back at &frac12; time.
```



```
<hr />
Q: How hot did it get when the police detective interrogated the
mathematician?<br />
A: x&sup3;&deg;
<hr />
Q: What does a punctilious plagiarist do?<br />
A: &copy;
<hr />
</p>
</body>
</html>
```

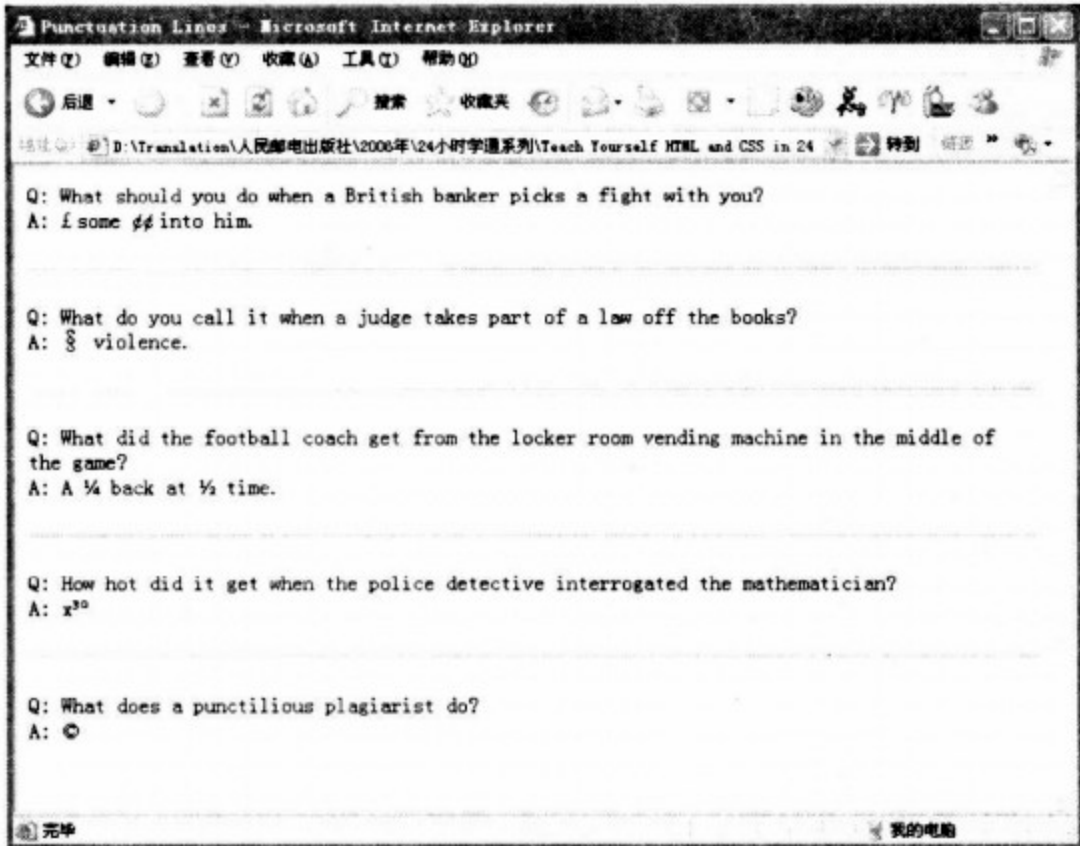


图 5.8  
程序清单 5.7 所示  
的HTML在大多数  
网页浏览器中的  
显示结果

## 5.8 总结

在这一章，读者学习用于指定许多 HTML 标签的选项和特殊行为的属性，还学习了如何使用 style 属性的 CSS 样式规则使文本居中对齐或右对齐。另外，还学习了如何创建和组合三种基本的 HTML 列表：有序列表、无序列表和定义列表。列表可以放在其他列表中，以创建大纲或其他复杂的文本排列。

本章还介绍了如何将文本显示为粗体、斜体或上标、下标、带下划线、特殊符号和重音字母。读者学习了如何适当地排列预格式化的等宽字体文本的段落、如何控制网页中文本的大小、颜色和字体。

表 5.3 对本章讨论的标签和属性进行了总结。顺便提一下，不要认为必须记住所有这些标签！这就是要有本书的原因：可以在需要时查阅这些标签。附录 B 列出了所有的 HTML 标签。

表 5.3

第 5 章介绍的标签和属性

标签/属性	功能
<em>...</em>	强调（通常是斜体）。
<strong>...</strong>	更重的强调（通常是粗体）。
<b>...</b>	粗体文本。
<i>...</i>	斜体文本。
<tt>...</tt>	打字机（等宽）字体。
<pre>...</pre>	预格式化的文本（原来的分行和空格都将被保留——这些分行和空格通常以等宽字体显示）。
<big>...</big>	比正常字体稍大的文本。
<small>...</small>	比正常字体稍小的文本。
<sub>...</sub>	下标。
<sup>...</sup>	上标。
<div>...</div>	要格式化的文本区域。
属性	
style="text-align:alignment"	将文本居中对齐、左对齐或右对齐（也可用于<p>、<h1>、<h2>和<h3>等）。
style="font-family:typface"	文本的字体名称，如 Arial（也可用于<p>、<h1>、<h2>和<h3>等）。
style="font-size:size"	字号，可设置为 small、medium、large、x-small 或 x-large 等。也可设置为具体的点数，如 12pt。
style="color:color"	修改文本的颜色。
<ol>...</ol>	有序（带编号的）列表。
属性	
style="list-style-type: numtype"	用于标记列表的编号类型。可能的值包括：decimal（十进制数字）、lower-roman（小写罗马数字）、upper-roman（大写罗马数字）、lower-alpha（小写 alpha 数字）、upper-alpha（大写 alpha 数字）以及 none（不使用数字）。
<ul>...</ul>	无序（带项目符号的）列表。
属性	
style="list-style-type: bullettype"	用于标记列表项的项目符号类型。可能的值包括：disc、circle、square 和 none。
<li>...</li>	用于<ol>或<ul>的列表项。
属性	
style="list-style-type:type"	标记列表项的编号或项目符号。可能的值包括：disc、circle、square、decimal、lower-roman、upper-roman、lower-alpha、upper-alpha 和 none。
<dl>...</dl>	定义列表。
<dt>...</dt>	定义项，作为定义列表的一部分。
<dd>...</dd>	与定义项对应的定义，也是定义列表的一部分。

5.9 问与答

问：我见过 Internet 上的网页使用三维小球或其他特殊图像作为项目符号。这是怎么实现的？

答：这种技巧有点超出了本章的范围，到第 8 章结束时你就知道怎么实现。

问：如何调整文本为每行的左右两边都是对齐的？

答：不能实现这一点，至少在 HTML/XHTML 中不能这么做。HTML 和 XHTML 本身不

支持对各行文本的对齐调整。但可以使用样式表来调整文本，这将在第三部分介绍。

问：如何找到安装在计算机上的字体的准确名称？

答：在 Windows 或 Macintosh 计算机上，打开“控制面板”，然后单击“字体”文件夹——这里列出了系统中安装的所有字体。在 font-family 样式规则中要使用准确的字体名，但大小写无所谓。

问：如何在网页中使用日语、阿拉伯语、汉语以及其他非欧洲语言字符？

答：首先，要浏览网页的用户计算机上必须安装相应语言的字体。用户必须在网页浏览器中选择语言字符集和字体。你可以使用 Windows 中的字符映射程序（或其他操作系统中类似的程序）获得任何语言中字符的数字编码；单击“开始/所有程序/附件”，然后选择“系统工具”，找到“字符映射”。如果要用的字符的编码是 214，则在网页中使用“&#214;”来表示这个字符。

要在网页中包含亚洲语言的简短信息（如“我们讲泰米尔语——请来电！”）的最好方法是，将它作为图像包含在网页中。这样所有人都能看到它，即使用户使用英语作为浏览网页的主要语言。

## 5.10 作业

作业包含一些测试题及练习，帮助巩固对这一章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 5.10.1 测验

1. 如何使网页的所有内容都居中对齐？
2. 如何缩进一个单词并在它的前面加上方框项目符号？
3. 使用定义列表显示：单词 glunch 的含义是“a look of disdain, anger, or displeasure”，单词 glumpy 的含义是“sullen, morose, or sulky”。
4. 如何在网页上显示“© 2004, Webwonks Inc.”？

### 5.10.2 答案

1. 如果你考虑将<div style="text-align:center">放在页面开头的<body>标签后，将</div>放在页面末尾的</body>之前，这是正确的。然而，<body>标签也支持 text-align 样式，也就是说可以不用<div>标签，直接将样式 style="text-align:center"放在<body>标签中，这样网页就都居中了。

2.

```
<ul style="list-style-type:square">
  <li>supercalifragilisticexpealidocious</li>
</ul>
```

将 `style="list-style-type:square"` 放在 `<li>` 标签中也能得到相同的结果，因为这个列表中只有一项。

3.

```
<dl>
<dt>glunch</dt><dd>a look of disdain, anger, or displeasure</dd>
<dt>glumpy</dt><dd>sullen, morose, or sulky</dd>
</dl>
```

4.

© 2004, Webwonks Inc.

下面的代码也能获得同样的效果：

© 2004, Webwonks Inc.

## 5.11 练习

试着为要放在网页中的信息创建一个有序列表。这将让你练习格式化 HTML 列表，还让你提前考虑第四部分将介绍的问题。



## 第 6 章

# 创建文本链接

在第 3 章，读者学习了使用[<a>](#)标签来创建 HTML 页面间的链接。这一章将进一步讨论链接，介绍如何使用这种标签让浏览者跳到同一网页的其他部分。这让你能够将文档分为不同的部分、创建包含链接的表或在页面末尾放置一个返回开头的链接。你还将学习如何链接到另一个网页的特定位置。

链接不仅可以用于连接网页；这一章的课程还将介绍如何在网页中嵌入一个指向电子邮件地址的链接，让读者能够立即编辑邮件并发送给你。还将介绍一个防止有人通过网页获得你的电子邮件地址，进而给你发送垃圾邮件的技巧。

### 6.1 使用锚在网页内部链接

在第 3 章提到过，[<a>](#)标签的名称来源自单词“anchor”，也就是说链接是网页中的一个目标点。到现在为止，你只知道如何使用[<a>](#)标签链接到其他网页，但这只是[<a>](#)标签的部分用途。

#### 6.1.1 使用锚标识网页中的位置

[<a>](#)标签将网页中的一个点标记为锚(anchor)，让你能够创建指向这个点的链接。本章稍后的程序清单 6.1 演示了一个链接，它指向页面内部的锚。先看看该程序清单中的[<a>](#)标签，以了解这种链接是如何创建的：

```
<a id="top"></a>
```

锚是网页中有名称的点。创建超文本链接和锚使用的标签相同（这就是为什么该标签名为[<a>](#)的原因）。

前几课提到过，[<a>](#)标签通常使用 href 属性来指定超文本链接的目标。在这个例子中，[<a>](#)标签仍指定一个目标，但并没有创建实际链接。相反，[<a>](#)标签为其在网页中的位置指定



了一个名称。必须包含</a>标签，且<a>标签的 id 属性必须有惟一的名称，但<a>和</a>之间不一定要有文本。<a>标签创建了锚点，这样可以从该页面或其他页面链接该锚点。

**By the way**

**注意:**老式 HTML 版本使用属性 name 而不是 id。新的 HTML 版本和 XHTML 不再使用 name 属性，而使用 id 属性。

## 6.1.2 链接到锚点位置

要链接到页面中的锚点，可使用<a>标签的 href 属性。看一下程序清单 6.1 中最后一个<a>标签就会明白：

```
<a href="#top">Return to Index.</a>
```

符号#指出，top 是当前文档中一个锚点的名称，而不是另一个页面。用户单击 Return to Index 时，网页浏览器将显示以标签<a id="top">打头的网页部分。

程序清单 6.1 使用<a>标签的 id 属性来设置锚点

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Alphabetical Shakespeare</title>
  </head>

  <body>
    <h2><a id="top"></a>First Lines of Every Shakespearean Sonnet</h2>
    <p>
      Don't ya just hate when you go a-courting, and there you are down on one
      knee about to rattle off a totally romantic Shakespearean sonnet, and
      zap! You space it. <i>"Um... It was, uh... I think it started with a
      B..."</i>
    </p>
    <p>
      Well, appearest thou no longer the dork. Simply pull this page up on your
      laptop computer, click on the first letter of the sonnet you want, and
      get an instant reminder of the first line to get you started. <i>"Beshrew
      that heart that makes my heart to groan..."</i> She's putty in your
      hands.
    </p>
    <h3 style="text-align:center">Alphabetical Index<br />
      (click on a letter)<br />
    <a href="#A">A</a> <a href="#B">B</a> <a href="#C">C</a>
    <a href="#D">D</a> <a href="#E">E</a> <a href="#F">F</a>
    <a href="#G">G</a> <a href="#H">H</a> <a href="#I">I</a>
```

```
<a href="#J">J</a> <a href="#K">K</a> <a href="#L">L</a>
<a href="#M">M</a> <a href="#N">N</a> <a href="#O">O</a>
<a href="#P">P</a> <a href="#Q">Q</a> <a href="#R">R</a>
<a href="#S">S</a> <a href="#T">T</a> <a href="#U">U</a>
<a href="#V">V</a> <a href="#W">W</a> <a href="#X">X</a>
<a href="#Y">Y</a> <a href="#Z">Z</a></h3>
<hr />
<h2><a id="A"></a>A</h2>
<p>
  A woman's face with nature's own hand painted,<br />
  Accuse me thus, that I have scanted all,<br />
  Against my love shall be as I am now<br />
  Against that time (if ever that time come)<br />
  Ah wherefore with infection should he live,<br />
  Alack what poverty my muse brings forth,<br />
  Alas 'tis true, I have gone here and there,<br />
  As a decrepit father takes delight,<br />
  As an unperfect actor on the stage,<br />
  As fast as thou shalt wane so fast thou grow' st,<br />
</p>
<p>
  <a href="#top"><i>Return to Index.</i></a>
</p>
<hr />
...
<h2><a id="X"></a>X</h2>
<p>
  (No sonnets start with X.)
</p>
<p>
  <a href="#top"><i>Return to Index.</i></a>
</p>
<hr />
<h2><a id="Y"></a>Y</h2>
<p>
  Your love and pity doth th' impression fill,<br />
</p>
<p>
  <a href="#top"><i>Return to Index.</i></a>
</p>
<hr />
<h2><a id="Z"></a>Z</h2>
<p>
  (No sonnets start with Z.)<br />
</p>
<p>
  <a href="#top">Return to Index.</a>
</p>
```

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

要记住这两种[<a>](#)标签之间的差别，有一个简单的方法：[<a href>](#)是将被单击的文本，而[<a id>](#)是单击后跳到的位置。同样，程序清单 6.1 中的每个[<a href>](#)链接都创建一个带下划线的链接，它们指向对应的[<a id>](#)锚点。例如，单击图 6.1 中 Alphabetical Index 下面的字母 B，将跳到图 6.2 所示的网页部分。

图 6.1  
程序清单 6.1 中的 [<a id>](#) 标签并不显示在网页中，而 [<a href>](#) 显示为带下划线的链接

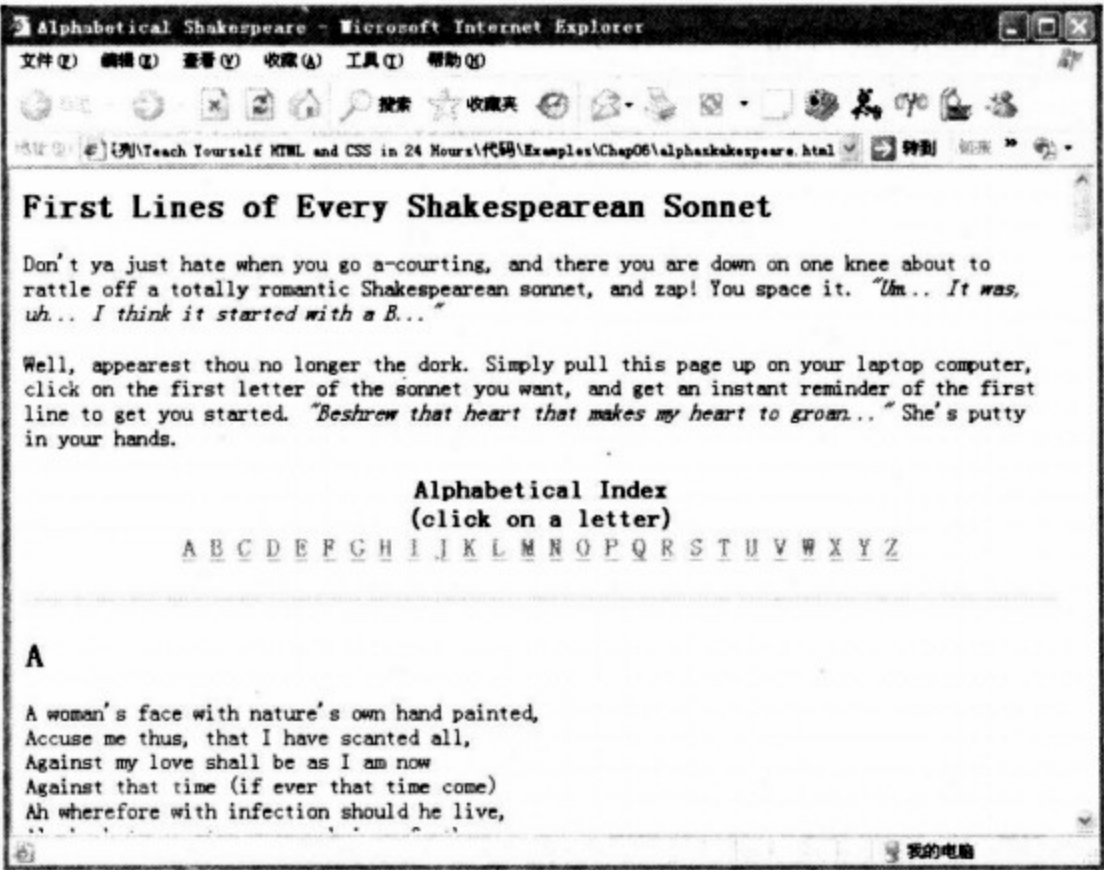


图 6.2  
单击图 6.1 所示页面中的字母 B，将跳到同一页面的对应部分



**注意：**通过<a>标签的 id 属性指定的锚点名称必须以字母字符开头。如果要为锚点 ID 编号，要以字母开始，如 photo1、photo2 等，而不能只使用 1、2 等。纯数字锚点 ID 在浏览器中是可行的，但这不是有效的 XHTML 代码。

By the Way

现在有一些互相链接的网页，可以在网页开头添加一个索引，让人们了解网页内容的大概：

- 在主页或较长的网页中每个主要的话题前面加一个<a id>标签。
- 将每个主要标题复制到网页开头的列表中，并将标题放在一个指向对应的<a id>标签的<a href>链接内。

**注意：**<a id>标签最常见的用途是创建字母索引。对于创建字母列表的人来说，有一个不好的地方是要输入 26 个指向 26 个锚点的链接，这相当繁琐。好消息是我已经创建了这样的链接，并将索引网页放在 Sams Publishing 网站 <http://www.sampublishing.com/> 上。

By the Way

## 6.2 链接到另一个网页的特定部分

被链接的锚点并不限于同一个页面，可以链接到另一个网页中的锚点，为此只需指定网页的地址或文件名，再加上#和锚点名。程序清单 6.2 显示了几个例子，下面是其中的一个：

```
<a href="sonnets.html#sonnet131">
You're bossy, ugly and smelly, but I still love you.</a>
```

程序清单 6.2 在<a href>标签中使用页面地址和锚点名

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Topical Shakespeare</title>
  </head>

  <body>
    <h2>Shakespearean Sonnets for Every Occasion</h2>
    <p>
      Choose your message for a genuine Shakespearean sonnet which expresses
      your feelings with tact and grace.
    </p>
  </body>
</html>
```

```

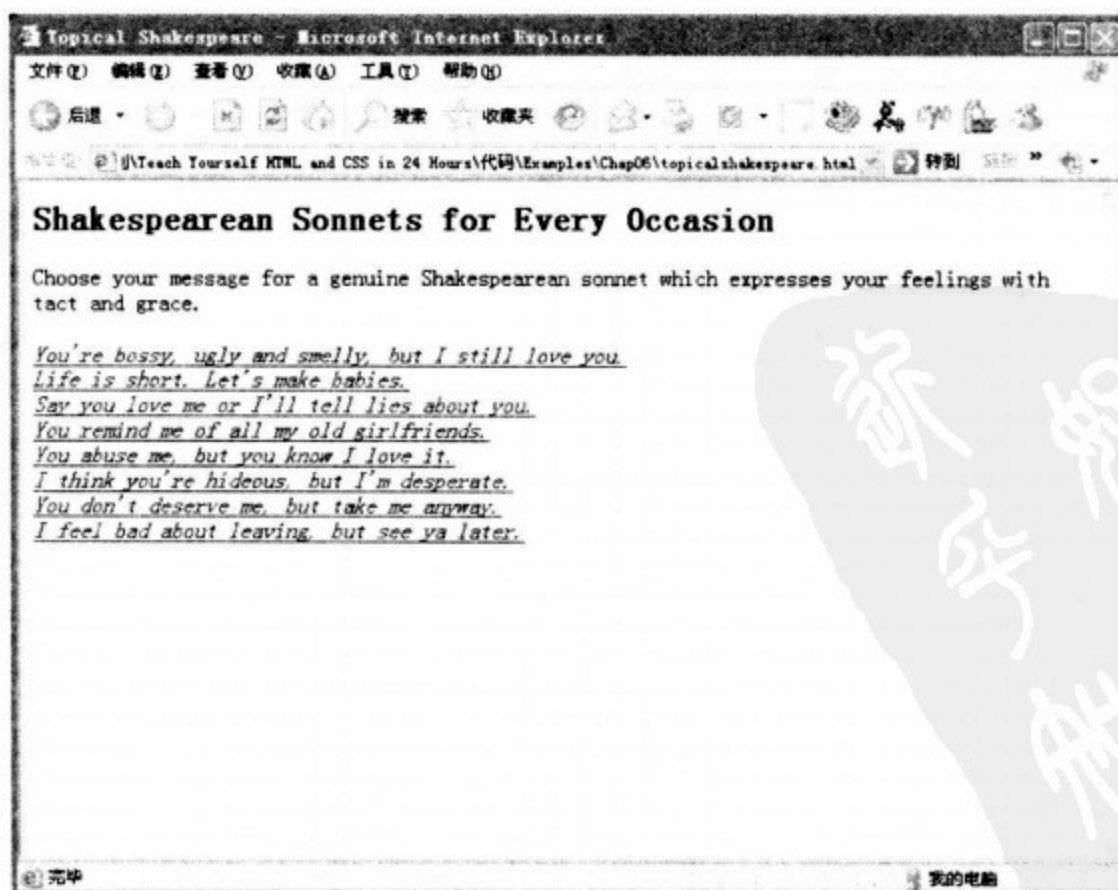
</p>
<p>
  <i><a href="sonnets.html#sonnet131">You' re bossy, ugly and smelly, but I
  still love you.</a><br />
  <a href="sonnets.html#sonnet2">Life is short. Let' s make babies.</a><br />
  <a href="sonnets.html#sonnet140">Say you love me or I' ll tell lies about
  you.</a><br />
  <a href="sonnets.html#sonnet31">You remind me of all my old
  girlfriends.</a><br />
  <a href="sonnets.html#sonnet57">You abuse me, but you know I love
  it.</a><br />
  <a href="sonnets.html#sonnet141">I think you' re hideous, but I' m
  desperate.</a><br />
  <a href="sonnets.html#sonnet150">You don't deserve me, but take me
  anyway.</a><br />
  <a href="sonnets.html#sonnet36">I feel bad about leaving, but see ya
  later.</a></i>
</p>
</body>
</html>

```

单击图 6.3 所示网页中的 “You’re bossy, ugly and smelly, but I still love you.”，将进入 sonnets.html 页面，并直接跳到页面中 `<a id="sonnet131"></a>` 所在的位置（见图 6.4）。sonnets.html 的 HTML 代码没有在这里列出，因为它太长了。这个页面只是一些十四行诗，每首诗前面有一个 `<a id>` 标签。注意，锚点 ID 可以是字母和数字的组合，只要 ID 以字母开头。在这个例子，我使用 sonnet 再加上诗的编号作为 ID。

图 6.3

程序清单 6.2 对应的网页，其中的链接指向 sonnets.html 页面的不同部分





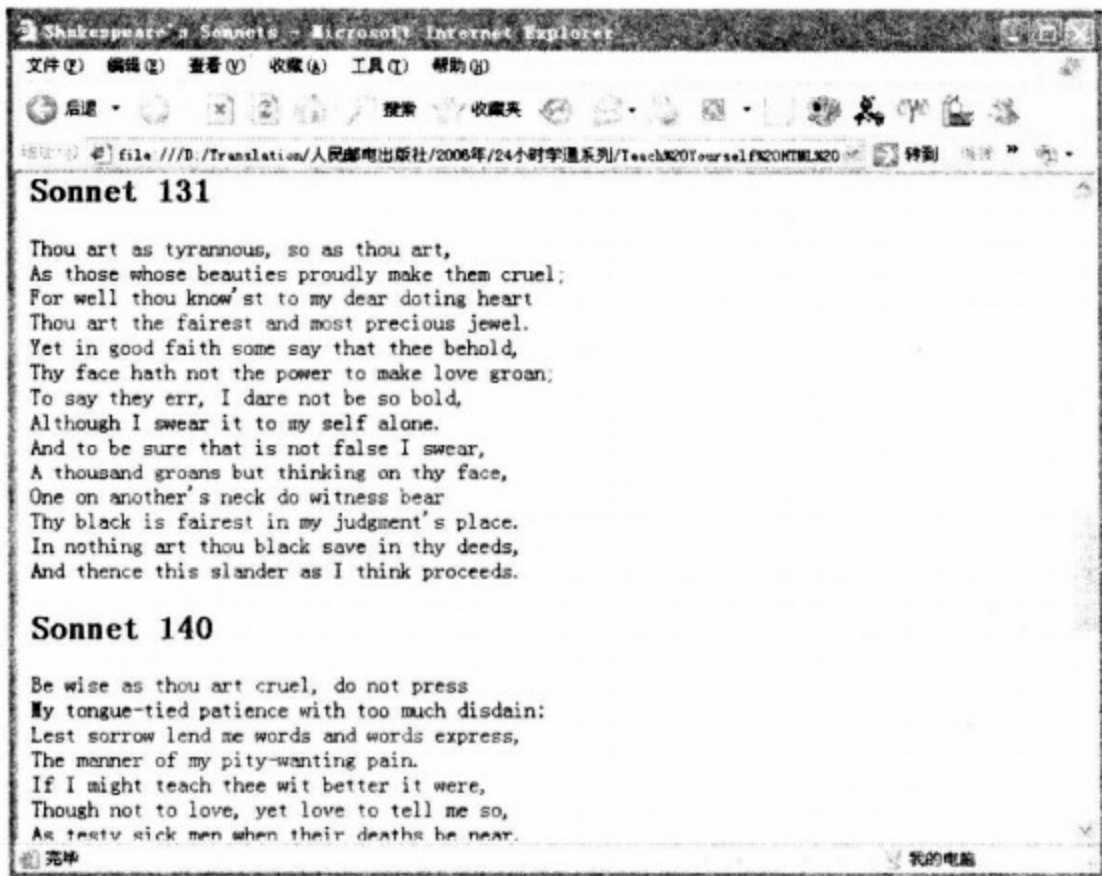


图 6.4

单击图 6.3 中的链接 bossy，将直接跳到 sonnets.html 页面的这个部分。该页面的 HTML 代码没有列出，但可从本书的网站获得

**休息一下：**提高网站知名度而不花太多营销费用的一种流行方法是，加入链接交换服务，如 Linkalizer (<http://www.linkalizer.com/>) 或 GoTop (<http://www.gotop.com/>)。这种服务让你能够在在一组主动推销自己的站点中互相包含站点链接，组成一个互相链接的网络。对新网站来说，链接交换是一个较廉价的提高知名度的方法。

Coffee Break

**警告：**只有链接标签 `<a href>` 才能包含符号 #。不要将符号 # 放在 `<a id>` 标签中；在这种情况下，指向该名称的链接将无法使用。

Watch Out!

### 6.3 在网页中链接 Email 地址

除网页之间及网页内部各部分之间的链接外，`<a>` 标签还可链接电子邮件地址。这是网页用户与你联系的最简单方法。当然，也可以只显示电子邮件地址，并相信用户如果要联系你，会在电子邮件程序中输入该地址，但这种方法更容易出错。可以提供一个指向电子邮件地址的链接，使用户无须花费什么功夫就能给你发邮件：

指向电子邮件地址的 HTML 链接如下所示：

```
<a href="mailto:dicko@netletter.com">Send me an email message.</a>
```

与其他 `<a>` 链接一样，Send me an email messag 将显示在网页中。在大多数网页浏览器中，当用户单击该链接时，将打开一个窗口，在这个窗口中可以输入消息并立刻发送给你——用户用于发送和接受电子邮件的程序将自动被调用。

如果希望人们看到真正的电子邮件地址（这样他们就能记下这个地址或使用其他电子邮件程序给你发送消息），可将该地址放在 href 属性中以及 `<a>` 与 `</a>` 标签之间，如下所示：

```
<a href="mailto:dicko@netletter.com">dicko@netletter.com</a>
```

前面提到，用户单击电子邮件链接时，将启动电子邮件程序，且在该程序中创建一个填好了收信人的新邮件。可在链接中提供更多的信息，使邮件的主题和正文也有默认值。这可以通过在 mailto 链接中添加 subject 和 body 属性来实现。这两个属性与常规 HTML 属性略有不同，因此用法也有些不同。属性与电子邮件地址之间用问号 (?) 分开，然后每个属性之间用和号 (&) 分开。下面的示例在上一个例子中指定主题和正文：

```
<a href="mailto:dicko@netletter.com?subject=Book Question&body=
When is the next edition coming out?">dicko@netletter.com</a>
```

用户单击该链接时，将创建一封电子邮件，收件人为 dicko@netletter.com，主题为 Book Question，正文为 “When is the next edition coming out?”。

### Did you know?

**提示：**如果只想指定电子邮件的主题而不指定正文，只需链接中的和号与 body 属性即可。

在将电子邮件地址插入到网页前，我要提出警告及对应技巧。你肯定很熟悉垃圾邮件制造者，他们创建电子邮件地址数据库，然后发送大量垃圾邮件广告给这些电子邮件地址。垃圾邮件制造者收集电子邮件的一个方法是，使用程序自动搜索网页中的 mailto 链接。我自己就有过将电子邮件地址发布到网页上，然后收到大量垃圾邮件的经历。因此现在我对公布电子邮件地址很谨慎。

幸运的是，有个技巧可以阻止大多数垃圾邮件制造者，这就是使用字符实体对电子邮件地址进行编码，这样可以骗过从网页上收集电子邮件地址的程序。以上面的电子邮件地址 dicko@netletter.com 为例，如果在这个地址中用字符实体来代替一些字符，大多数电子邮件收集程序将抛弃这样的地址。小写的 ASCII 字符实体以 a 的代码 &#97; 开始，然后按字母顺序依次递增；因此，字母 c 的代码为 &#99;。用字符实体来代替这个例子中的两个 c，得到如下代码：

```
<a href="mailto:di&#99;ko@netletter.&#99;om">Send me an email message.</a>
```

有很重要的一点需要注意，网页浏览器将字符实体 &#99; 解释为 c，也就是说，从浏览器的角度来看，最终结果是一样的。然而，自动电子邮件收集程序搜索网页的 HTML 代码，在这种情况下就是一个混乱的电子邮件地址。

将学到的关于电子邮件链接的知识应用于一个示例，如程序清单 6.3 和图 6.5 所示。这是曲棍球队员网页的另一个版本，其中包含一个链接，指向我朋友 Terry 的电子邮件。

### By the way

**注意：**如果将 Terry 的电子邮件地址公布在成千上万本书中，将是很危险的，因此示例代码中的电子邮件地址并不是真的，请读者谅解。

#### 程序清单 6.3 使用 <a> 标签链接到电子邮件地址

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
```

```
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
```

```
<head>
```

```
<title>Music City Mafia - Terry Lancaster</title>
```

```
</head>
```

```
<body>
```

```
<p style="font-family:verdana, arial; font-size:18pt; font-weight:bold">16
```

```
Terry Lancaster</p>
```

```
<p style="font-family:verdana, arial; font-size:12pt; color:navy">
```

```
<br />
```

```
<span style="font-weight:bold">Nickname:</span> Big T<br />
```

```
<span style="font-weight:bold">Position:</span> RW<br />
```

```
<span style="font-weight:bold">Height:</span> 6' 3"<br />
```

```
<span style="font-weight:bold">Weight:</span> 195<br />
```

```
<span style="font-weight:bold">Shoots:</span> Left<br />
```

```
<span style="font-weight:bold">Age:</span> 40<br />
```

```
<span style="font-weight:bold">Birthplace:</span>
```

```
<a href="http://en.wikipedia.org/wiki/Nashville%2C_Tennessee">Nashville,  
TN</a>
```

```
</p>
```

```
<hr />
```

```
<p style="font-family:verdana, arial; font-size:12pt; color:navy">
```

```
<span style="font-weight:bold">Favorite NHL Player:</span>
```

```
<a href="http://www.nhl.com/players/8448091.html">Brett Hull</a><br />
```

```
<span style="font-weight:bold">Favorite NHL Team:</span>
```

```
<a href="http://www.nashvillepredators.com/">Nashville Predators</a><br
```

```
/>
```

```
<span style="font-weight:bold">Favorite Southern Fixin:</span>
```

```
<a href="http://southernfood.about.com/od/potatorecipes/r/blbb442.htm">
```

```
Skillet Fried Potatoes</a><br />
```

```
<span style="font-weight:bold">Favorite Meat and Three:</span>
```

```
<a href="http://www.hollyeats.com/Swetts.htm">Swett' s</a>
```

```
(<a href="http://maps.google.com/maps?q=2725+clifton+ave,+nashville,+tn"  
rel="external">map</a>)
```

```
<br />
```

```
<span style="font-weight:bold">Favorite Country Star:</span>
```

```
<a href="http://www.patsycline.com/">Patsy Cline</a><br />
```

```
<span style="font-weight:bold">Favorite Mafia Moment:</span>
```

```
"<a href="mcmplayer_chale.html">Chet</a> finishing the game with his  
eyelid completely slashed through."
```

```
</p>
```

```
<hr />
```

```
<p style="font-family:verdana, arial; font-size:12pt">
```

```
<a href="mailto:l&#97;ncastert@musiccitym&#97;fi&#97;.com?subject=  
Fan Question&amp;body=What' s your secret?">Contact Terry.</a>
```





**提示：**将网页制作者的电子邮件链接放在每个网页的底部已成为一种惯例。这样不仅使别人更容易联系你，也使他们在你可能没有测试的网页上遇到问题时可以告诉你。不要忘记使用电子邮件地址字符实体的技巧，以避免电子邮件地址被垃圾邮件制造者收集。

**Did you  
Know?**

如果仔细查看图 6.5，将注意到除电子邮件链接外还有另一个新链接。我指的是出生地信息字段，它现在显示的是链接“Nashavill, TN”。在这个最新的例子中，我使用一个链接指向我最喜欢的网站之一，即信息丰富的 Wikipedia——免费的在线百科全书。有关该网站的更详细信息，请往下阅读。

## 6.4 链接到 Wikipedia

在第 3 章，我极力说明了 Google Maps 是如何有用以及在通过网页与人们共享地理位置信息时有多方便。我要推荐的另一个非常优秀的网站是 Wikipedia，其网址为 <http://www.wikipedia.org/>。Wikipedia（维基百科全书）是免费的在线百科全书，这听起来可能并不是那么特别。Wikipedia 的特别之处在于，每个人都可以编辑它。是的，你现在就可以访问 Wikipedia，并在上面编辑任何一条百科全书条款。你可能会认为 Wikipedia 的开放性可能会使它变得很混乱，但它实际上运行得非常好。关心特定主题的人们将监视这些主题，保证其内容是客观而准确的。我并不是说 Wikipedia 是完美的，但它确实是比较成功的在线社会“实验”之一，已成为非常有用的信息工具。

那么，Wikipedia 与介绍 HTML 和 CSS 的书有什么关系呢？我认为有时在网页中包含指向 Wikipedia 的链接很有用，这样，访问网站的人可以了解到更多信息。在刚看到的曲棍球队网站中，我包含了一个指向 Wikipedia 上关于该队员出生地条款的链接，这样访问者可以了解到关于那个城市的更多信息。下面是该页中的链接代码：

```
<a href="http://en.wikipedia.org/wiki/Nashville%2C_Tennessee">Nashville, TN</a>
```

这行代码揭示了 Wikipedia 另一个有趣的地方：它的链接目标地址直接来自链接的主题。具体地说，到 Wikipedia 的链接是在 Wikipedia 网页地址后加上主题而形成的。在这个例子中，链接的主题是 Nashville, Tennessee。这里有一个技巧，就是将逗号改为%2C——这是在网页中表示逗号的特殊代码，并将空格用下划线（\_）代替。然后，将该主题加到 Wikipedia 的 URL <http://www.wikipedia.org/wiki> 之后。

**注意：**如果你忘记了，这里再提醒一下：URL 表示统一资源定位符（Uniform Resource Locator），它是 Web 资源（如网页）的地址。

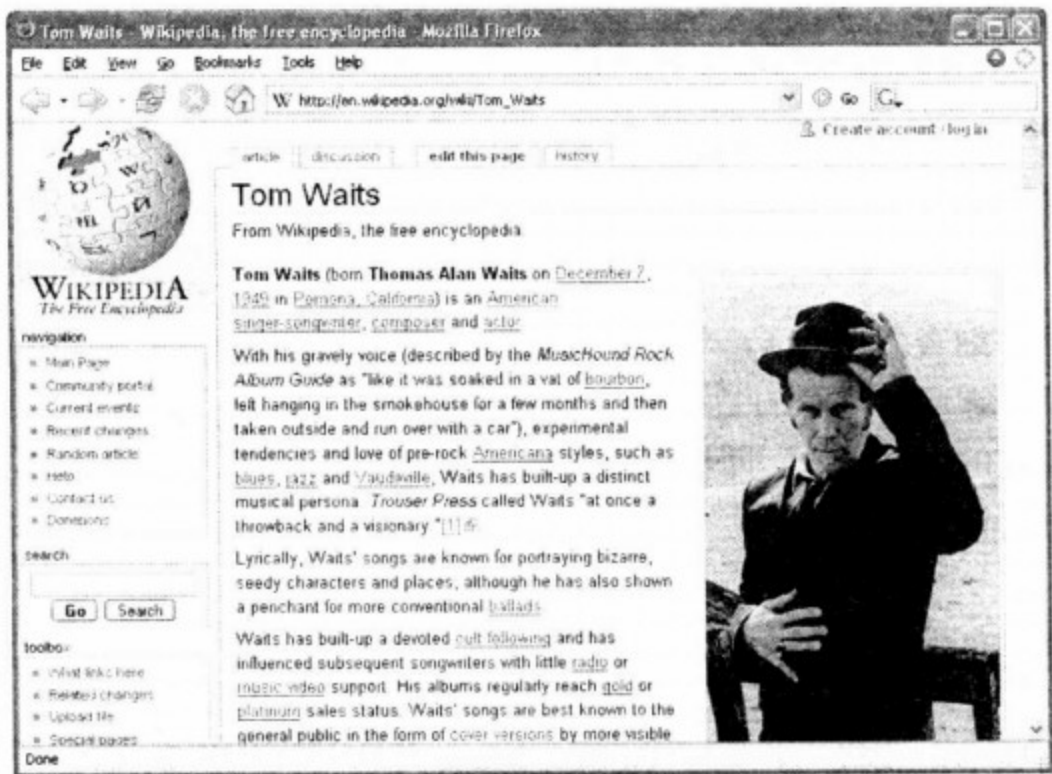
**By the  
Way**

关于 Wikipedia 地址，好消息是可以直接从网页浏览器中将其复制并粘贴到网页中。举个例子，假设要链接 Wikipedia 上关于音乐家 Tom Waits 的条款。根据上面对 Wikipedia 地址如何组成的解释，可以推断出，该页面的地址应是 [http://www.wikipedia.org/wiki/Tom\\_Waits](http://www.wikipedia.org/wiki/Tom_Waits)。但不需要进行猜测，可以在 Wikipedia 上输入“Tom Waits”进行检索，结果将得到如图 6.7 所示的页面。



图 6.7

Wikipedia 是优秀的资源，可从网页链接到 Wikipedia 上的有趣话题



上图表明，关于 Tom Waits 的 Wikipedia 地址的猜测是正确的。然而，不需要猜测，可以从网页浏览器的地址栏复制该地址。

Wikipedia 之所以与本课有直接关系，是因为可以使用 Wikipedia 页面的目录来创建指向该页面特定部分的链接。例如，如果访问 Tom Waits 的页面 ([http://en.wikipedia.org/wiki/Tom\\_Waits](http://en.wikipedia.org/wiki/Tom_Waits))，并稍微向下滚动，将找到一个包含一系列标题，如 Early Career、Discography 和 Tours 等的目录。这些标题对应于该页中的一些锚点，这意味着可以将目录中标题表加在页面地址后，创建指向该页中这些部分的链接。别忘记在标题前要使用字符#，如下所示：

[http://en.wikipedia.org/wiki/Tom\\_Waits#Discography](http://en.wikipedia.org/wiki/Tom_Waits#Discography)

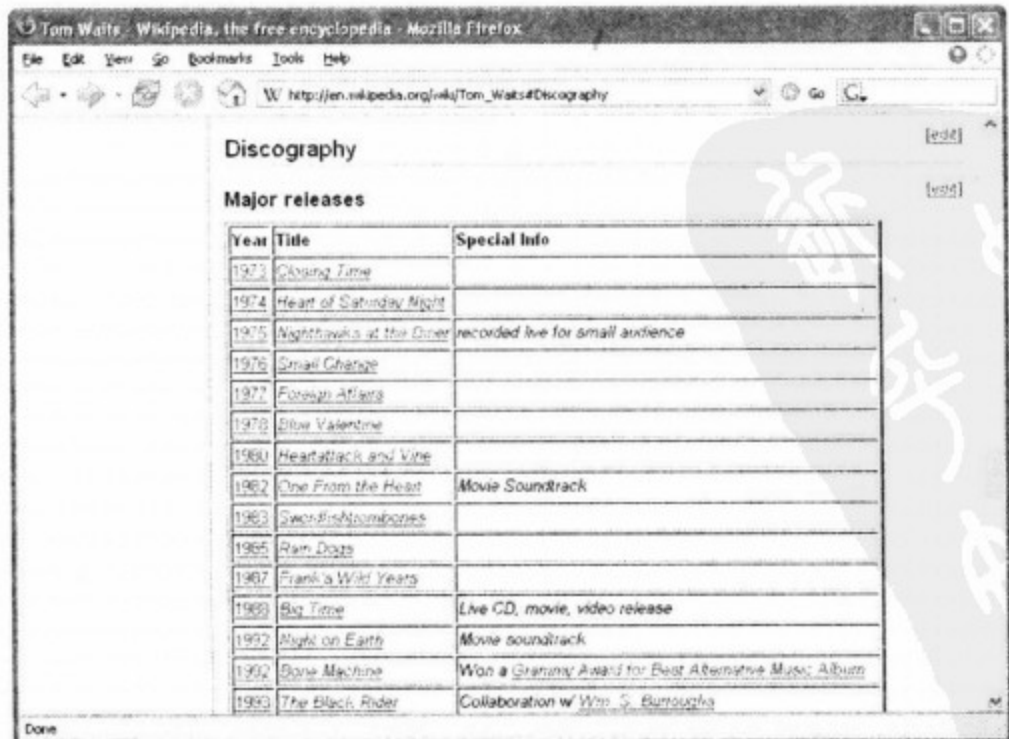
**Did you know?**

**提示：**如果标题包含空格，创建指向锚点的链接时要用下划线 ( \_ ) 代替空格。

这个链接将跳到 Tom Waits 页面中的 Discography 部分，如图 6.8 所示。

图 6.8

大多数 Wikipedia 页面都包含根据页面目录创建的锚点



## 6.5 总结

这一章介绍了第 3 章没有讲到的<a>标签的两种用法。读者学习了如何在网页中创建锚点以及如何创建指向特定锚点的链接；还学习了如何链接到电子邮件地址，让用户能够轻松地给你发邮件。读者还学习了如何保护电子邮件地址免受垃圾邮件的骚扰。最后，还介绍了如何在网页中链接到 Wikipedia 百科全书的主题。表 6.1 总结了这一章讨论的<a>标签的两个属性。

表 6.1 第 6 章介绍的 HTML 标签和属性

标签/属性	功能
<a>...</a>	通过 href 属性，创建指向另一个文档或锚点的链接；通过 name 属性，创建可被链接的锚点
属性	
href="address"	要链接的文档或锚点的地址
id="name"	锚点在这个文档中的名称

## 6.6 问与答

问：可以将 href 和 id 放在同一个<a>标签中吗？在什么情况下需要这样做？

答：可以，将命名的锚点和链接在一起时，可减少输入。然而，通常最好分别使用<a href>和<a id>，以免造成混乱，因为它们在 HTML 文档中的作用截然不同。

问：如果无意中拼错了锚点名或忘记在名称前加上#，将发生什么问题？

答：如果链接到不存在的锚点或拼错了锚点名，将链接到相应网页的开头。

问：如果提供邮件服务和网页服务的公司不同，邮件链接仍可以使用吗？

答：可以。可以在链接中放置 Internet 上的任何电子邮件地址，且可以正常使用。电子邮件链接不能使用的惟一情况是，单击链接的人没有在其计算机中安装电子邮件程序，这样就不能发送电子邮件。

## 6.7 作业

作业包含一些测试问及练习，帮助巩固对这一章介绍的内容的理解。试着回答所有问题，然后再看后面的“答案”小节。

### 6.7.1 测验

1. 编写 HTML，当用户单击网页顶部的 About the authors 时，跳转到网页底部的制作者及有贡献者的姓名表。

2. 假如你的公司有三个雇员，创建一个公司目录网页，列出每个雇员的一些信息。编写这个网页的 HTML 以及从另一个网页链接到一个雇员的 HTML。

3. 如果你的电子邮件地址是 bon@soir.com, 如何将文本 goodnight greeting 放入链接, 让用户单击该链接就可以撰写并发送邮件到你的电子邮箱中? 如何对电子邮件地址中的 o 进行编码, 对垃圾邮件制造者隐藏电子邮件地址?

## 6.7.2 答案

1. 在网页开头输入:

```
<a href="#credits">About the Authors</a>
```

在姓名列列表的前面输入:

```
<a id="credits"></a>
```

2. 公司目录网页如下所示:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html>
  <head>
    <title>company directory</title>
  </head>

  <body>
    <h1>company directory</h1>
    <h2><a id="jones"></a>jane jones</h2>
    <p>
      ms. jones is our accountant... etc.
    </p>
    <h2><a id="smith"></a>sam smith</h2>
    <p>
      mr. smith is our salesman... etc.
    </p>
    <h2><a id="bharwaniji"></a>r.k. satjiv bharwahniji</h2>
    <p>
      mr. bharwahniji is our president... etc.
    </p>
  </body>
</html>
```

如果这个文件名为 directory.html, 从另一个网页到一个雇员信息的链接如下所示:

```
<a href="directory.html#bharwaniji">about our president</a>
```

3. 在网页中输入:

```
send me a <a href="mailto:bon@soir.com">goodnight greeting</a>!
```

为对垃圾邮件发送者隐藏电子邮件地址，将地址中的“o”替换成“&#111;”，如下所示：

```
send me a <a href="mailto:b&#111;n@s&#111;ir.c&#111;m">goodnight  
greeting</a>!
```

## 6.8 练习

1. 从其他网页链接回主页时，你可能想跳过主页开头的一些介绍性信息。使用一个链接，它正好指向介绍性信息下面的锚点，这样可以避免将信息呈现给已经读过它的人，从而使网页显得不那么重复。另外，如果网页信息在网页浏览器上超过两屏，可考虑在网页底部放置一个返回开头的链接。

2. 浏览一下你的网页，看看是否有什么地方需要让人更容易对你所说的提出意见，在那些地方加入指向你的电子邮件地址的链接。应尽可能多地提供机会，让人与你联系，告知他们的要求或对产品的想法，尤其是在商业经营中。



## 第 7 章

# 创建网页图像

网页中放置图像和创造性文本，并不要求你是个艺术家；也不必花费成百上千美元在软件上。本章将介绍如何创建图像使网页的视觉效果更好。虽然本章使用流行的 Windows 图像软件（Corel 的 Paint Shop Pro），但读者也可使用 Windows 或 Macintosh 中各种主要的图像应用程序。

本章只关心为 Web 图像创建图像文件，因此不讨论 HTML 标签。读者将在第 8 章学习如何将图像集成到 HTML 页面中。

### Did you know?

**提示：**为节省用于创建网页图像和媒体文件的时间，最好方法之一当然是完全避免创建它们。要从网页上抓取图像，只要右击图像（在 Macintosh 上是按住 Ctrl 键并单击鼠标），然后，如果是在 Mozilla Firefox 中，选择“将图像另存为（Save As）”，如果是在 Internet Explorer 中，选择“将图片另存为（Save As）”。抓取背景图片也很简单，只要右击图片然后选择“将背景另存为（Save As）”。

你可能考虑使用无版权的剪贴画，这不需要获得版权许可。一个不错的在线剪贴画资源是微软的 Office 在线剪贴画和媒体网站，其网址为 <http://office.microsoft.com/clipart/>。Barry 的剪贴画服务器也是一个受欢迎的剪贴画网站，其网址为 <http://www.barrysclipart.com>。

### Watch Out!

**警告：**网站上的图像或媒体剪贴画只要版权所有者的授权就可以使用。在网页上使用有版权的媒体前一定要征得授权。关于 Web 上图像的版权，请参见 <http://www.2learn.ca/copyright/images.html>。不征得许可，除可能带来许多法律麻烦外，也是一种很不专业的做法，更别说不道德了。

## 7.1 选择图像软件

可以使用任何计算机图像软件来创建网页图像，从最简单的随操作系统一起发布的免费



画图程序，到昂贵的专业软件，如 Adobe Photoshop。如果计算机连接了数码相机或扫描仪，这些仪器可能自带了图像软件，可用于创建网页图像。

**注意：**Adobe Photoshop 无疑是图像编辑软件的翘楚。然而，它很昂贵，且如果没有计算机图形的经验，它对你来说也非常复杂。Adobe 现在提供一个更便宜、更容易使用的 Photoshop 版本，称为 Photoshop Elements，你可以看一看。关于 Adobe 的产品，可参见 Adobe 的网站 <http://www.adobe.com/>。

**By the Way**

如果觉得某个软件用于创建 Web 图像不错，可使用它来完成本章的课程。如果它无法执行这里介绍的某些功能，它可能就不是好的 Web 图像工具。在这种情况下，如果使用的是 Windows 计算机，可考虑下载 Paint Shop Pro 或 Adobe Photoshop Elements 的试用版；如果是 Macintosh 用户，可使用 Adobe Photoshop Elements。对于 Windows 用户，还有一个非常有趣的新图像软件，那就是 Google Picasa，可在 <http://picasa.google.com/> 免费下载。Picasa 是 Google 进入图像编辑市场的敲门砖，它提供了一些其他图像编辑软件没有的特性。然而，Picasa 更适合于组织和编辑数码照片，而不是从头创建 Web 图像。即使如此，它仍是一个强大（且免费）的工具，值得一试。

#### 练习

有一个优秀的软件，它价格也不贵，能够提供创建 Web 图像所需的一切功能，那就是 Corel 的 Paint Shop Pro。如果你使用的是 Windows 计算机，可以在阅读本课前下载具有完整功能的 Paint Shop Pro 试用版。（Macintosh 用户可以从 <http://www.adobe.com/> 下载 Adobe Photoshop Elements，因为现在 Paint Shop Pro 只能用于 Windows。如果你更喜欢 Adobe 产品，Photoshop Elements 也有 Windows 版。）下面是如何获得 Paint Shop Pro 的步骤。

1. 启动网页浏览器，访问 <http://www.corel.com/>。网站将提示你选择所在地区。选择后将看到 Corel 网站的主页。
2. 单击“试用下载”链接，然后单击 Paint Shop Pro 试用版旁边的“下载”按钮。
3. 单击“下载”按钮后，文件将传输到硬盘上。浏览器将询问运行还是保存该文件。选择“保存”，并指定在硬盘上保存该文件的位置——记住这个文件夹！
4. 下载完成后，使用 Windows 资源管理器找到下载的文件，双击它开始安装 Paint Shop Pro。

**注意：**从网上下载的 Paint Shop Pro 软件是功能完整的共享试用版。如果觉得它在创建网页上很有用，可支付一定的费用向 Corel 注册该软件。Paint Shop Pro 试用版只能使用 30 天，因此到那时你必须考虑注册该软件还是寻找其他办法。

**By the Way**

你在本书看到的几乎所有图像都是使用 Paint Shop Pro 创建的，本章也将使用 Paint Shop Pro 来演示一些关键的 Web 图像技巧。当然，使用 Paint Shop Pro 创建图像有许多方法，这里不可能全部介绍。幸运的是，Paint Shop Pro 的最新版本包含学习中心，指导一些常见的图像编辑操作——调整大小、自动调整亮度和颜色等。

## 7.2 必须知道的图像基础知识

将图像和多媒体放到 Internet 上时，总有两股力量会互相冲突。你的眼睛和耳朵希望一切都尽可能地详细和精确，而你的时钟和钱包却希望文件尽可能小。精致的、颜色丰富的图像意味着大文件，这即使在快速网络上也将增加传输时间。如何使多媒体的质量最大化同时又使文件最小？为进行选择，需要了解颜色和分辨率是如何营造出质量的主观感觉的。

图像的分辨率是组成图像的点或像素（组成数字图像的点）的数量。高分辨率的大型图像通常比低分辨率的小型图像需要更长的传输和显示时间。分辨率通常以图像的宽乘以高表示，单位是像素；例如，300×200 的图像宽 300 像素，高 200 像素。

分辨率不是图像文件的大小（和传输时间）最主要的决定因素。这是因为网页上使用的图像总是以压缩格式存储和传输。图像压缩是一种数学运算，将图像中的重复模式都删除。图像压缩的数学运算很复杂，基本原理是重复模式或相同颜色的大块区域在图像存储到磁盘上时可以删除。这使图像文件变得小得多，在 Internet 上传输也更快。然后网页浏览器在显示图像时再恢复图像原来的样子。

在本章余下的篇幅中，你将学习如何创建美观而文件较小的图像。将用到的技巧取决于图像的内容和用途。网页图像的用途与网页的用途一样多，但现在有四种图像用法最常见：

- 人、产品或地方的照片；
- 图形横幅或徽标；
- 用于网页间链接的花哨按钮或图标；
- 网页的背景纹理或墙纸。

最后一种将在第 9 章介绍，现在可以学习创建其中三种图像。

## 7.3 准备照片图像

要将照片放在网页上，需要扫描仪或数码相机，即便是手机内置的数码相机也可以。通常需要使用扫描仪或数码相机自带的软件将图片保存到硬盘上。然而，可以在 Paint Shop Pro 和大多数其他图像软件中直接控制扫描仪——具体细节请参见相关的软件文档。

**Did you  
know?**

**提示：**如果没有扫描仪或数码相机，大多数冲洗胶卷的商店只要很少的费用就可以将 35mm 的胶卷传输到 CD-ROM 上。然后你可以用 Paint Shop Pro 直接从 CD-ROM 上打开和修改图像文件；实际上，要将 CD-ROM 上的文件复制到硬盘上，然后再修改，因为 CD-ROM 可能是只读的。

将图片复制到计算机中后，可用 Paint Shop Pro（或其他类似的软件）对图片进行修改，然后放到 Web 上。

### 7.3.1 裁剪图像

网页图像应尽可能简洁，因此通常需要裁剪或缩小数码照片。在 Paint Shop Pro 中，屏幕左边的学习中心窗口提供了一些常用图像编辑技巧，包括图像裁剪。执行下列步骤，在 Paint Shop Pro 中裁剪图片：

1. 单击 Learning Center 窗口中的 Adjust，然后单击 Crop。Learning Center 窗口如图 7.1 左边所示。

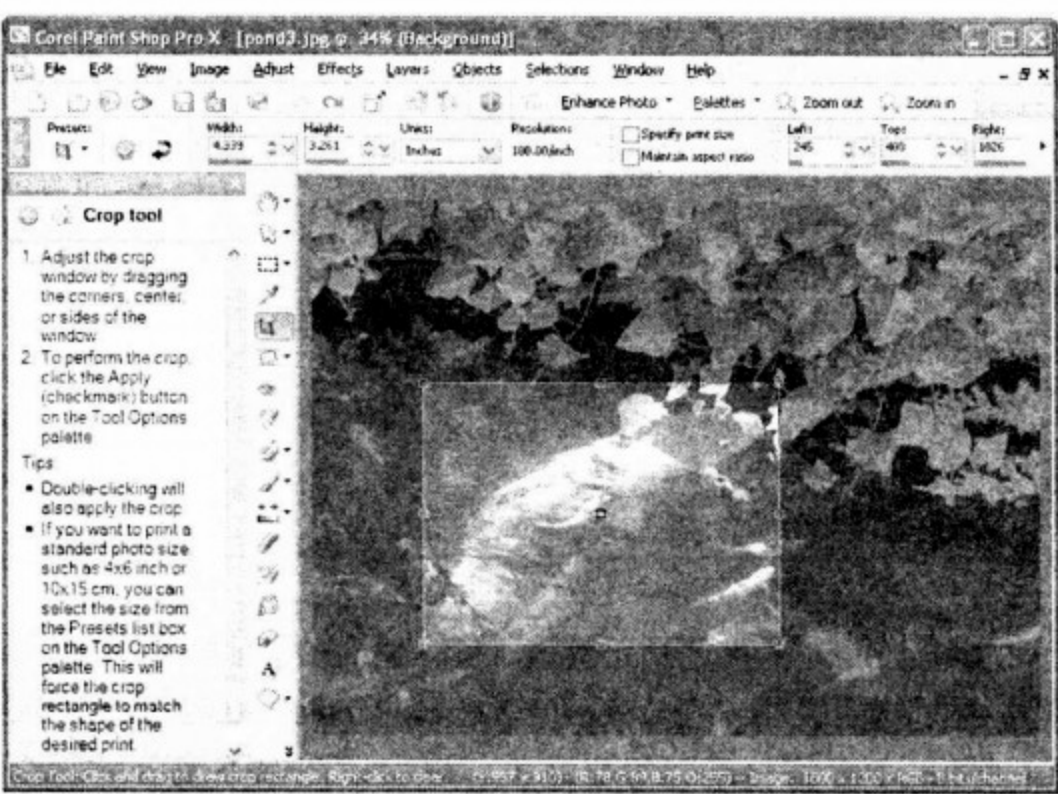


图 7.1  
使用矩形选择工具尽可能紧凑地裁剪图像

2. 单击要保留的图像部分的左上角，并按住鼠标左键向右下角拖曳。

**提示：**如果裁剪图像时要保留原来的宽高比，按住 Shift 键再单击并拖曳鼠标。

**Did you Know?**

3. 单击靠近应用程序窗口顶部的工具调板中的 Apply 按钮（绿色勾号）。

即使图像经过裁剪后，它可能还是比用于网页所需的尺寸大。根据网页设计，可能需要将大图像限制在 800×600、640×480 像素以内或更小。在有些情况下，可能还要为图像创建一个缩略图链接，它指向大图像，缩略图可能要在 100 像素以内。

**提示：**在 Paint Shop Pro 中，当前图像的分辨率显示在窗口的右下角。图像可能看起来比实际大或小，因为 Paint Shop Pro 自动调整图像以适应窗口的大小。当前缩放比例显示在每个图像窗口的标题栏中。要查看图像在网页上的大小，可以在主菜单中选择 View/Zoom to 100%。

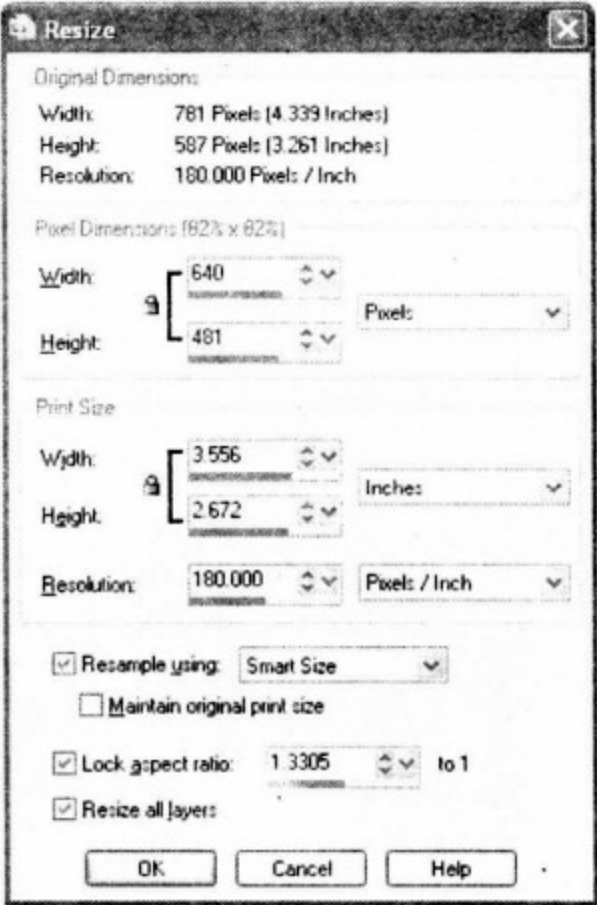
**Did you Know?**

### 7.3.2 调整图像的大小

要通过改变图像分辨率来调整图像的大小，可使用菜单 Image/Resize。也可以选择

Learning Center 窗口中的 Adjust，然后选择 Resize。这时将弹出如图 7.2 所示的 Resize 对话框。

图 7.2  
要修改图像的大小，选择菜单 Image/Resize 打开该对话框



最常使用的重新采样（Resample）设置包括：智能大小（Smart Size）、锁定宽高比（Lock Aspect Ratio）和调整所有选定图层的大小（Resize All Layers selected）。如果选择锁定宽高比，则输入图像的宽度时，Paint Shop Pro 将自动计算出高度，以保证图像不被压扁变形。如果指定高度，Paint Shop Pro 也将自动改变宽度。另外，我在调整图像大小时，选择根据像素尺寸而不是比例来调整，因为我希望图像最后接近 640×480。

### 7.3.3 改变图像颜色

许多照片需要进行颜色校正，以便在计算机屏幕上显示的效果最好。与大多数照片编辑软件一样，Paint Shop Pro 提供了许多选项用于调整图像的亮度、对比度和色彩平衡。

大多数选项都很直观，其中最重要、最强大的也是最易于使用的。从主菜单中选择 Adjust/One Step Photo Fix，或单击 Learning Center 的 Adjust 区域中的 One Step Photo Fix。图像的亮度、对比度和颜色将自动调整，很多时候自动调整的结果很不错。

另一个选项是对图像校正过程进行控制：选择 Adjust/Smart Photo Fix（也可以在 Learning Center 的 Adjust 区域中选择）。如图 7.3 所示，可以移动 Smart Photo Fix 对话框中的多个滑块来调整亮度、饱和度和焦距（清晰度），直到对图像满意为止；通常默认值就够了，单击 Suggest Setting 按钮可以恢复默认设置。

在 Paint Shop Pro 中，大多数其他图像编辑工具都提供了如图 7.3 所示的预览窗口，因此，尝试是了解每个工具功能的最好方法。



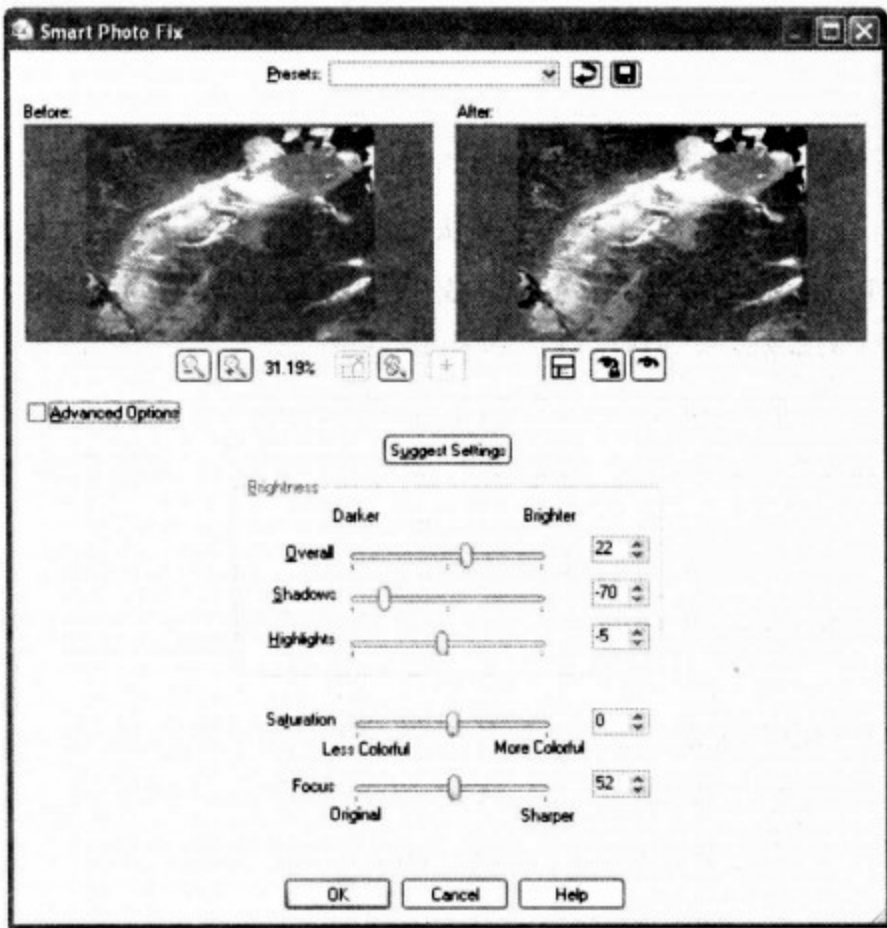


图 7.3  
Smart Photo Fix 让你能够在建议修复的基础上进一步调整照片校正过程

### 7.3.4 控制 JPEG 压缩

照片图像保存为 JPEG 格式的效果最好。调整好照片的大小和外观后，选择 **File/Save As**，并将编码类型设置为 **Standard Encoding**。

单击 **Options** 按钮，将打开如图 7.4 所示的 **Save Options** 对话框。可以在 1%（高质量、大文件）和 99%（低质量、小文件）之间调整 **Compression Factor** 设置，控制保存 JPEG 文件的压缩比例。

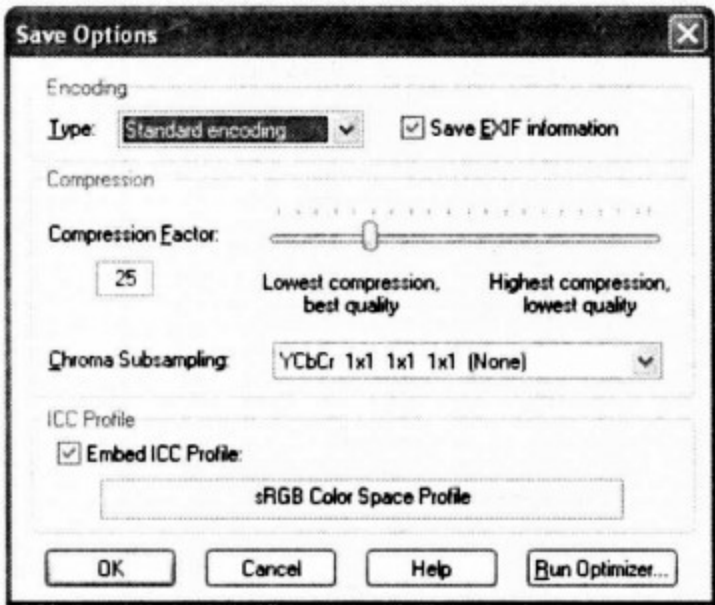


图 7.4  
Paint Shop Pro 允许保存 JPEG 图像时降低图像质量以减小文件大小

你可能希望试验不同 JPEG 压缩比例如何影响图像的质量。通常 25%对于大多数照片图像是不错的速度和质量平衡。



## 7.4 创建横幅和按钮

从头开始创建的图像，如横幅和按钮时，要考虑的因素与照片很不一样。

创建横幅或按钮时要做的第一个决定是图像应多大。现在访问 Web 的大多数用户的屏幕分辨率至少是 800×600，有的可能要大得多；例如，我的屏幕现在设置为 1280×1024 像素。应规划好图像大小以便在较小的屏幕（800×600）中也能正常显示，还能留下空间给滚动条和边距。重要的大小约束是页面的水平宽度，因为水平滚动页面争议很大，其常常使 Web 用户感到不便。竖直滚动页面就比较容易接受，因此页面的高度可以比屏幕的最小高度还大。

**Did you know?**

**提示：**屏幕分辨率随着时间的推移而逐渐提高。现在可以假设大多数 Web 用户使用的是 800×600 的分辨率，不久后分辨率最低可达 1024×768。

假设最小目标分辨率是 800×600 像素，这意味着横幅和标题图像的宽度不应超过 770 像素，高度不应超过 430 像素，这是除去滚动条、工具栏和浏览器窗口的其他部分剩下的最大可见区域。在页面中，普通的照片和艺术画的每边应在 100~300 像素之间，按钮和图标应更小，在 20~100 像素之间。

选择 File/New 创建新图像时，将进入如图 7.5 所示的对话框。如果不确定图像应为多大，不必担心——只要接受默认大小即可。如果能够确定，就输入新图像的宽和高。如果输入的宽和高是以像素为单位，则可以不管 Resolution。另外，创建大多数图像时都可以使用设置 Raster Background。开始时应将颜色深度设置为 RGB - 8bits/channel，因为这样可以创建高质量的图像。

图 7.5  
在开始创建图像之前就要决定图像大致的大小



至于图像的背景颜色，通常应选择白色，以便和大多数网页浏览器用于显示网页时的背景颜色相匹配(你将在第 9 章学习如何修改页面的背景颜色)。如果页面的背景颜色不是白色，

可以选择其他背景颜色。实际上，你可能希望创建没有背景的图像，在这种情况下可以选中复选框 **Transparent**。如果图像的背景是透明的，图像下面的网页将能够透过透明区域显示出来。

输入图像的宽度和高度并单击 **OK** 后，将显示一块空白画布——这有点可怕，如果你和大多数人一样有艺术恐惧症的话。幸运的是，使用诸如 **Paint Shop Pro** 等计算机图像处理软件可以很容易地为大多数 Web 应用创建看起来很专业的图像。

**注意：**如果新建的图像以灰白棋盘图案显示，就表明选择了透明背景。**Paint Shop Pro** 中的透明区域总是以这种棋盘图案显示。

**By the Way**

通常，希望在网页图像中包含一些好看的文本。例如，你可能希望在网页顶部放置一个标题，它使用花体字并有阴影或其他特殊效果。按照下列步骤进行操作，可在 **Paint Shop Pro** 中实现这一点：

1. 单击 **Learning Center** 窗口中的 **Text and Graphics**，然后单击 **Add Text**。
2. 在图像上要显示文本的地方单击。这时将出现如图 7.6 所示的 **Text Entry** 对话框。

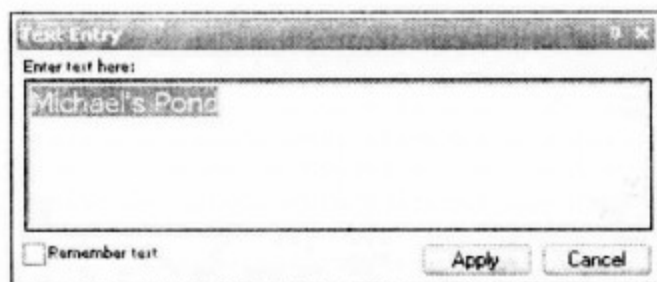


图 7.6

使用 **Paint Shop Pro** 的文本工具在图像中创建美观的文本

3. 输入要在图像中添加的文本，然后单击并拖曳鼠标以选中文本。
4. 从 **Paint Shop Pro** 窗口右边的调色板中选择文本的颜色。可能要为文本的轮廓和填充区域选择同一颜色，这在调色板中显示为两个重叠的方框。

**注意：**在 **Paint Shop Pro** 中，通过主工具栏调整 **Text Entry** 对话框中的文本时，该对话框将继续显示。直到完成文本编辑后该对话框窗口才关闭。

**By the Way**

5. 在工具调板中选择文本的字体、字号和其他文本样式等（见图 7.7）。务必将 **Anti-Alias**（消除锯齿）设置为 **Smooth**，这可以使文本的边缘是光滑的。单击 **Text Entry** 对话框中的 **Apply** 按钮，接受新文本。

6. 单击 **Pick** 工具（箭头），然后拖曳文本调整它在图像中的位置（通常位于图像的正中央）。

**提示：**使用 **Pick** 工具选中文本后，可用键盘上的箭头按键准确地调整文本在图像中的位置。

**Did you Know?**

刚将文本添加到图像中时，文本四周将显示一个矩形。这意味着文本处于选中状态，菜单操作将作用于文本。例如，你可能在主菜单中选择了 **Effect/3D Effect/Drop Effect**，给文本增加阴影。这时将显示如图 7.8 所示的对话框，提示是否要在增加阴影效果前将文本转换为光栅对象——只要单击 **OK**，对文本进行转换即可。

图 7.7

在 Paint Shop Pro 程序主窗口的工具调板设置字体、字号和其他字体设置

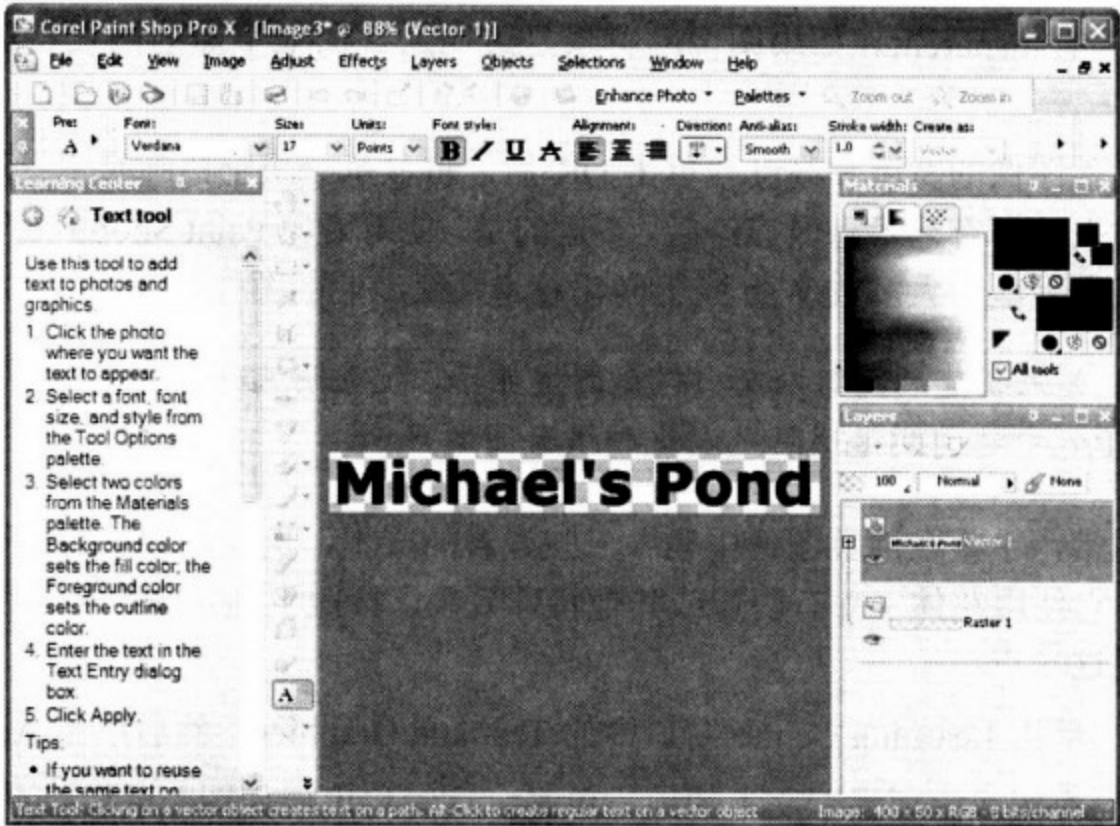
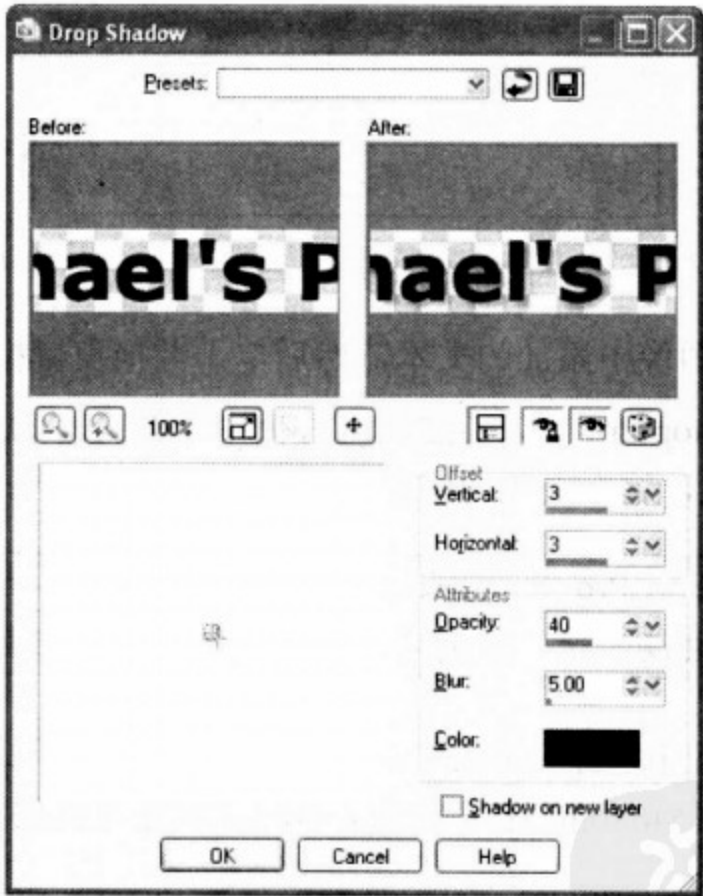


图 7.8

与 Paint Shop Pro 中大多数菜单选择一样，Effect/3D Effect/Drop Effect 也提供易于使用的预览



注意，在对图像真正实施修改前，可以调整阴影效果并在小型预览窗口中查看效果。这样用户很容易试验各种不同的效果，从而知道每种效果的作用。使用文本工具，并选择菜单 Effect/3D Effect 中选择 Buttonize、Chisel、Cutout、Drop Shadow 或 Inner Bevel，就能创建各种美观的 Web 图像。

在对文本应用效果后，可能还想对文本进行变形、模糊和锐化等。为此，只需选择 Effect/Effect Brower，然后选择要使用的效果。这将打开如图 7.9 所示的对话框，可在其中的效果列表中进行选择，并预览每种效果。



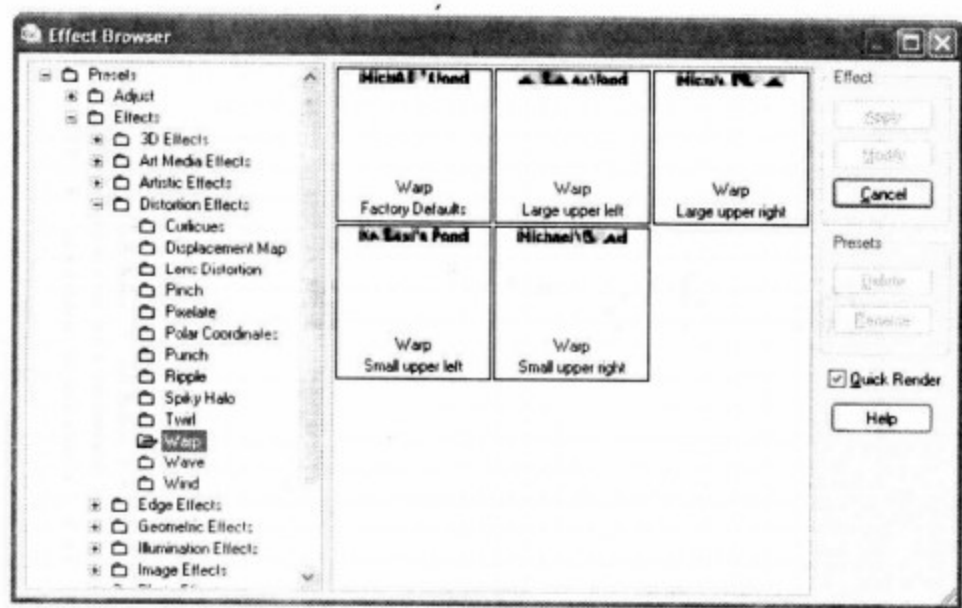


图 7.9  
选择 Effect/Effect Brower，试验所有改变图像的特效，然后选择所需的一种

在该图中，使用的是 Warp（变形）效果，将文本以某种奇怪的方式拉伸和扭曲。尝试效果浏览器中的各种选项，是一种非常有趣的体验！

**休息一下：**如果对创建有趣的文本效果感兴趣，又不想浪费时间使用 Paint Shop Pro 等图像编辑工具，可以考虑使用其他在线工具。Cool Text 是一个在线图像生成器，让你能够输入文本和自定义的效果，Cool Text 将为你生成图像。花些时间尝试一下位于 <http://cooltext.com/> 的 Cool Text。

Coffee Break

## 7.5 减少图像的颜色数

要减少图像的大小以缩短下载时间，最有效的方法之一是减少图像中使用的颜色数。这可能极大地降低有些照片图像的质量，但非常适合于大多数横幅、按钮和其他图标。

在 Paint Shop Pro 中，可以通过选择 Image/Decrease Color Depth，然后选择一个具体的减少颜色选项（大多数图像编辑软件都有类似的选项）。如果图像的颜色很少，就选择 16 Color Palette；如果图像的颜色很多（或图像使用 16 色的效果不好），则选择 256 Color Paletter。程序将根据图像的颜色范围自动找到最适合的 16 色或 256 色调色板。

**注意：**如果图像只使用了两三种颜色，将其保存前仍应该选择 Image/Decrease Color Depth /16 Color Palette。否则，图像文件将保留许多颜色——即使实际使用的颜色很少，从而浪费空间。

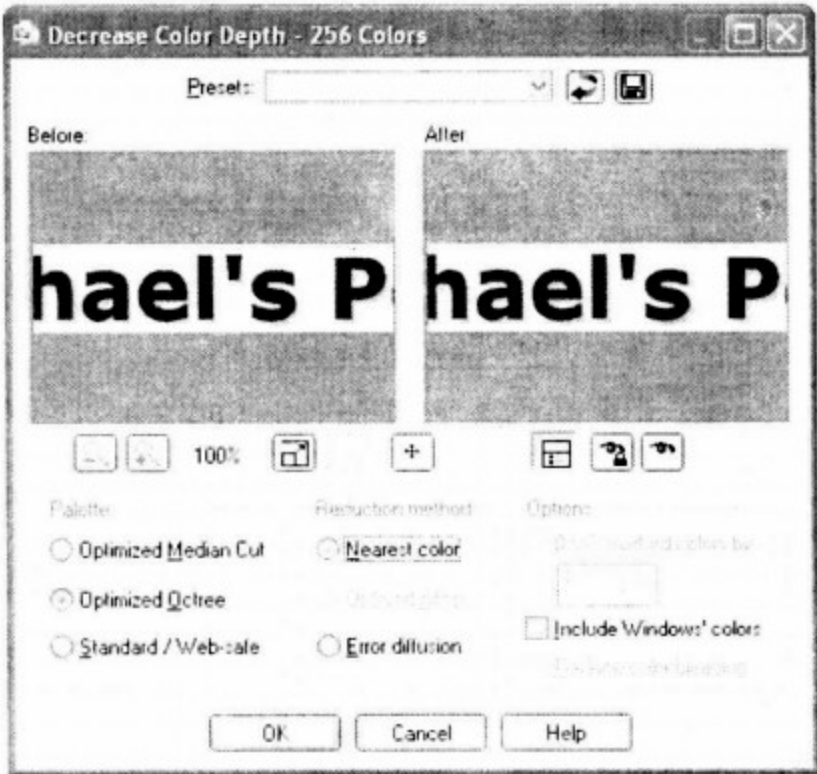
By the Way

减少图像的颜色数时，将看到一个包含很多选项的对话框（见图 7.10）。对于网页图像，通常选择单选按钮 Optimized Octree 和 Nearest color。对话框右边的所有选项都不要选择，这些选项对图像的质量几乎没有什么改进。

**提示：**仿色（在 Paint Shop Pro 也称为误差扩散）是图像编辑软件使用的一种技术，通过交替显示两种颜色接近的像素来模拟调色板中没有的颜色。例如，交替显示红色和白色像素来模拟粉红色。在有些情况下，仿色可使图像的质量看起来更好，但通常网页图像应避免仿色。为什么？因为仿色极大地增加了图像的信息复杂度，通常使图像文件更大、下载起来更慢。

Did you Know?

图 7.10  
减少图像的颜色数  
可以显著地降低文  
件大小，且对图像  
外观不会有太大的  
改变



有一种特殊的图像文件格式，其颜色只有有限的几种，这就是可交换的图像文格式 (Graphics Interchange Format, GIF)。要在 Paint Shop Pro 中保存 GIF 图像，选择 File/Save As，然后选择 GIF CompuServe Graphics Interchange 作为图像类型。GIF 图像格式是为包含纯色区域的图像设计的，如网页标题和其他插图图像，并不适合用于照片。

比 GIF 图像格式更好的一种格式是 PNG，现在所有主要的网页浏览器都支持这种格式。GIF 图像格式允许指定一种透明颜色，即网页的背景将透过图像的这些区域显示出来。然而，PNG 允许指定不同的透明度，而不只是一种透明颜色。在 Paint Shop Pro 中选择“PNG Portable Network Graphics”图像类型将图像保存为 PNG 图像。建议对这两种格式都试一试。在第 9 章，将演示为什么将透明图像放在网页中时 PNG 格式的这种高级透明特性胜过 GIF 的透明度。

## 7.6 交错 GIF 和渐进 JPEG

JPEG、GIF 和 JPEG 这三种图像格式都提供了一种功能，让图像能够更快地显示。图像可以以一种特殊方式存储，显示时先大概显示图像的草图，当文件全部下载后再填充细节。这起到一种很有意义的心理效果，因为这样使人们有东西可看，而不必坐着干等大型图像慢慢显示在屏幕上。

以这种方式保存的文件称为交错 GIF、交错 PNG 或渐进 JPEG。尽管叫法不同，但这三种格式的视觉效果类似。交错 GIF 或交错 PNG 文件首先是一块一块地显示，下载完后再填补更多的细节。同样，渐进 JPEG 文件首先是模糊的，然后渐渐清晰起来。

大多数可以处理 GIF 文件的图像软件都将图像保存为交错或非交错的。例如，在 Paint Shop Pro 中，可在保存 GIF 文件前在 Save As 对话框中单击 Option 按钮，然后选择单选按钮 Version 89a 和 Interfaces (89a 是最新的 GIF 版本) (见图 7.11)。PNG 也提供了类似的选项，允许在 Save As 对话框中指定存为交错的还是非交错的。



要保存为渐进 JPEG 文件，选择 **Save As** (或 **Save Copy As**)，然后选择图像类型 **JPG JPEG**，单击 **Options** 按钮，并选择 **Progressive Encoding** 而不是 **Standard Encoding**。现在网页上使用的许多较大的 GIF 和 JPEG 图像都是交替和渐进，给 Web 用户以更好的图像浏览体验。

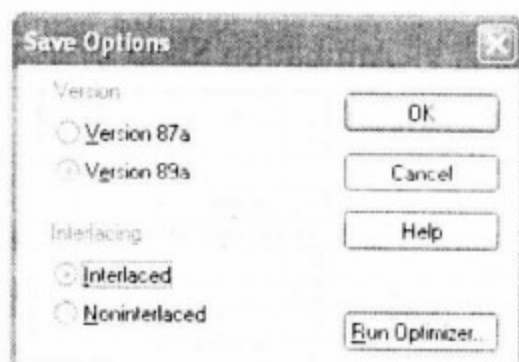


图 7.11

Paint Shop Pro 可以将图像保存为交错 GIF，这样在加载的时候可以更快地显示

**注意：**3KB 以下的图像文件通常加载都很快，因此看不出交错或渐进显示的过程。实际上，很小的图像使用交错显示时反而更慢。较大的图像才使用交错或渐进。

**By the Way**

## 7.7 创建 Web 动画

GIF 图像格式可以用于创建动画，为网页增加动感和趣味。GIF 动画的传输也比大多数视频或其他多媒体文件快。图像编辑器 Paint Shop Pro 原来包含一个称为 Animation Shop 的模块，专用于创建网页 GIF 动画。现在这个模块已经成为一个独立的产品，仍称为 Animation Shop，开发商是 Corel，它从开发商 Jasc 那里购买了 Paint Shop Pro。可从 Corel 网站 <http://www.corel.com/> 下载 Animation Shop 的免费试用版。还有一些其他的 GIF 动画软件，包括免费软件和高级商业软件包。Animation Shop 提供了许多优秀的功能，且易于使用、价格低廉。

创建 GIF 动画的第一步是创建一系列要连续显示的图像。每个图像称为一帧。可以使用喜欢的任何图像软件来创建这些图像，但如果要使用 Animation Shop 将这些图像组成动画，Paint Shop Pro 无疑是最好的选择。

如果知道如何使用 Paint Shop Pro (或其他高级图像软件) 的图层特性，将发现创建动画帧更容易，因为可以很容易显示或隐藏图片的一部分来制造图层变化或移动的效果，以模拟动画。但不用担心，这里不讨论图层操作。可以通过复制普通的单层图像，然后绘制一些变化或移动图像的某部分，以创建下一帧，这样实现动画效果。

当动画的各帧都准备好后，使用菜单 **Image/Decrease Color Depth** 将帧限制为 256 色或 16 色，然后在图像编辑器中将各帧保存为独立的 GIF 文件。

## 7.8 将图像组装成动画

保存各个 GIF 文件后，启动 Animation Shop，将这些文件组装成单个动画文件。

**Did you Know?**

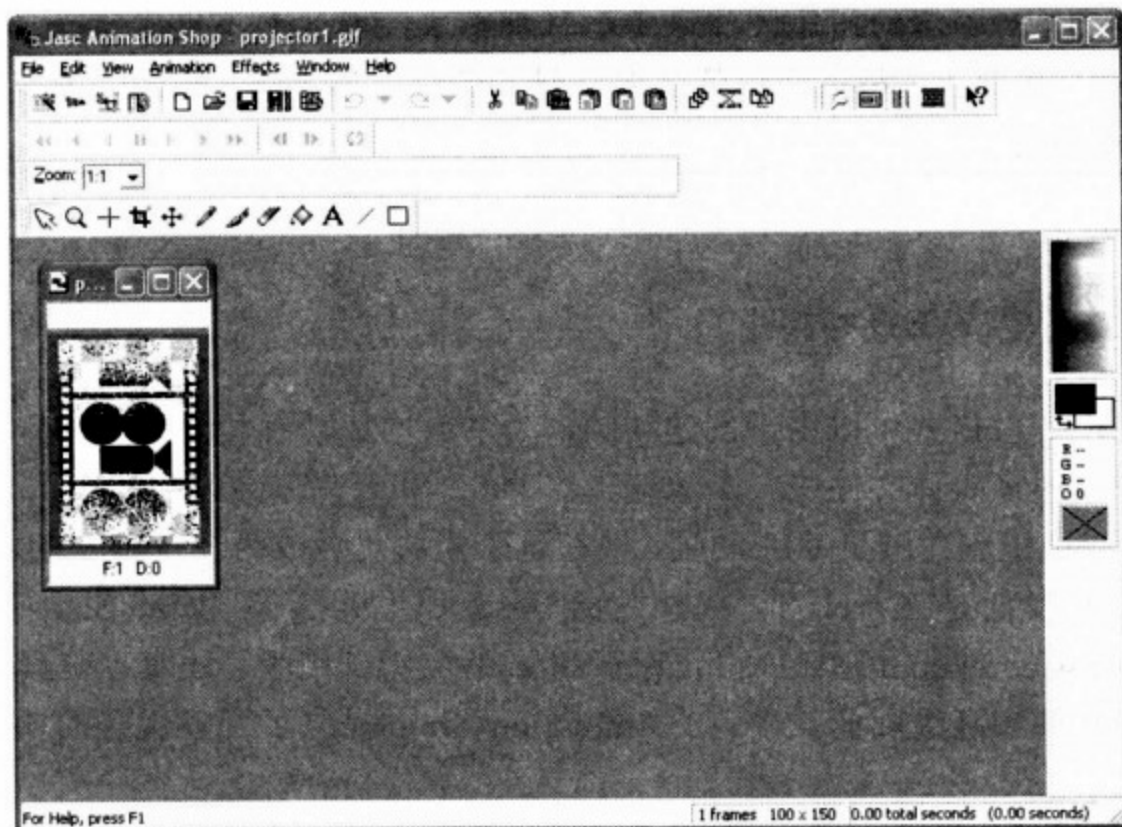
**提示：**要使用 Animation Shop 创建简单的 GIF 动画，最快的方法是选择 File/Animation Wizard。这将通过提问，引导你完成下面将讨论的所有步骤。然而，本章将介绍如何手工地创建动画，而不使用动画向导。这将成为你使用本章最后要讨论的高级动画技巧做好准备。

这里的原理简单得不能再简单：只要告诉 Animation Shop 要显示哪些图片以及以什么样的顺序显示。此外，还要指定两个细节：每幅图片显示多少时间、整个序列重复显示多少次。下面是组装动画的步骤。

1. 在 Animation Shop 中选择 File/Open，然后选择要用作动画第一帧的图像文件。如图 7.12 所示。注意，图像的透明度将被保留，因此透出了灰色棋盘图案。在这里，创建的是一个电影放映机动画。

图 7.12

这是单帧 GIF 图像在 Animation Shop 中打开时的样子



2. 选择 Animation/Insert Frames/From File，打开如图 7.13 所示的对话框。单击 Add File 按钮，然后选择要用作动画第二帧的图像。再次单击 Add File 添加第三帧，依次类推，直到列表中包含所有要用于该动画的图像。将 Insert Before 设置为 2，这样新帧将插入到原始帧的后面。单击 OK。

3. 现在应该看到所有帧依次排列，就像电影胶片（见图 7.14）。如果不能同时看到所有帧，可以调整动画窗口的大小，使用滚动条在胶片中来回移动。选择 View/Animation 可预览动画。如果帧的顺序有错，只要使用鼠标将顺序有错的帧拖曳到正确的位置。当然，动画可能移动得很快，以至于无法判断顺序。这是因为 Animation Shop 默认将动画设置为每帧显示 1/100 秒。通常需要增加每帧显示的时间。

4. 要设置帧的显示时间，首先单击帧，该帧的边框将变为蓝色和红色，然后选择 Animation/Frame Properties。也可以右击帧，然后在弹出菜单中选择 Frame Properties。这将打开一个对话框，从中指定帧显示的时间，单位为 1/100 秒。在这个例子中，设置为 25 即可，

即每帧的显示时间为 1/4 秒；设置为 100 就相当于 1 秒。

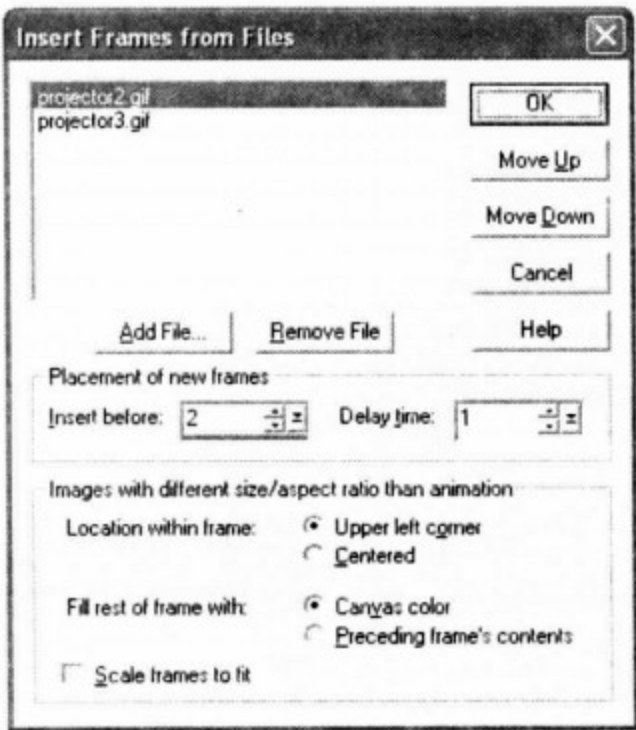


图 7.13  
选择 Animation/Insert Frames/From File 将打开该对话框。使用 Add File 按钮选择要添加到动画中的图像

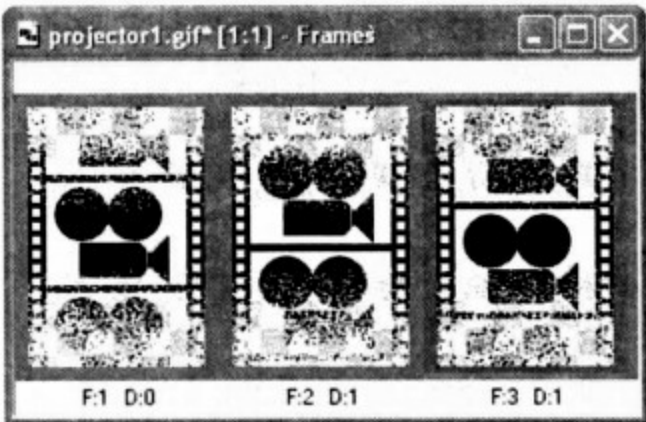


图 7.14  
Animation Shop 并列显示动画的所有帧，就像电影胶片一样

**提示：** Animation Shop 将自动记住你最后一次设置的帧显示时间，并自动将新添加的帧的显示时间设置为该值。因此，如果在第一次添加新帧前设置了显示时间，新帧将自动利用最开始的帧设置。

**Did you know?**

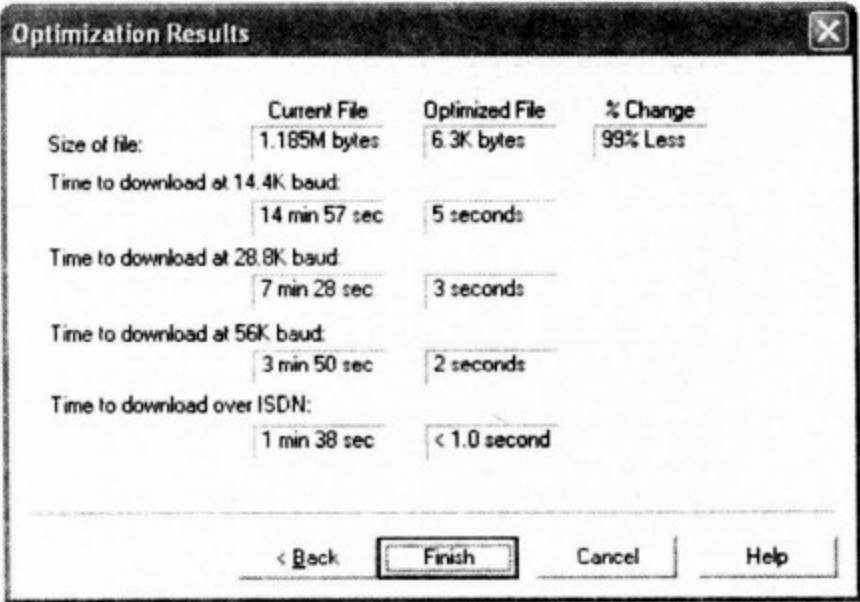
5. 只剩最后一个细节动画就要创建好了！ Animation/Frame Properties（或右击动画中的帧，在弹出菜单中选择 Frame Properties），打开 Animation Properties 对话框。要将电影放映动画设置成只要用户在浏览就一直运行，因此，单击 Loop 标签，然后选择 Repeat the Animation Indefinitely。但有时可能需要将动画设置成只播放一次（或几次），然后将最后一帧显示为静态图像。在这种情况下，应选择第二个选项，并输入要重复的次数。
6. 现在动画创建好了。选择 File/Save As 保存动画。指定文件名后单击 Save 按钮，将有一个滑块控件让你在良好的图像质量和较小的文件大小选择进行折衷。通常 Animation Shop 根据滑块设置能够选择最合适的优化。将滑块向上滑动将提高图像质量，向下滑动将减小文件。设置好后单击 Next。
- 进行一些配置后，可以看到动画预览，然后提示你单击 Next。单击后将看到如图 7.15 所示的报告。这样很容易确定图像是否太大，然后单击 Back，返回并调整滑块。当文件大小



可接受时，单击 Finish。

图 7.15

这个报告有助于判断在图像质量和文件大小是否取得了恰当的平衡



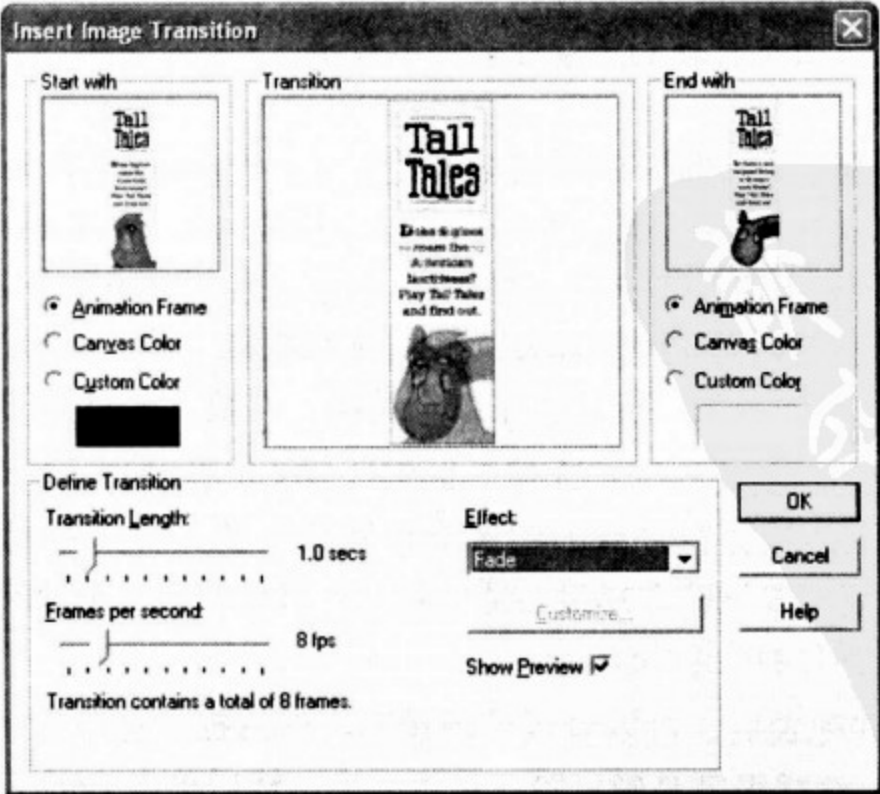
### 7.9 生成过渡和文本效果

Animation Shop（与其他 GIF 动画软件一样）不仅能够将多个 GIF 图像组装成动画文件，还能够生成令人印象深刻的特效，甚至创建滚动的文本横幅。不幸的是，由于篇幅有限，本书无法详细介绍如何使用这些特性，但这些特性都很简单，借助于 Animation Shop 在线帮助系统就能学会。

图 7.16 和图 7.17 演示了创建的过渡和文本效果。使用图像过渡功能（Effect/Insert Image Transition）在横幅广告之间平滑地切换（你将在第 17 章看到用于 JavaScript 动画中的横幅广告）。另一个例子使用了文本过渡（Effect/Insert Text Effect）让文本 Limited Online Offer!在白色背景上平滑地滚动。这并不需要任何图像，因为文本过渡效果生成自己的文本图片。

图 7.16

使用 Effect/Insert Image Transition 在图像之间生成淡入淡出、溶解、渐隐和其他自动过渡



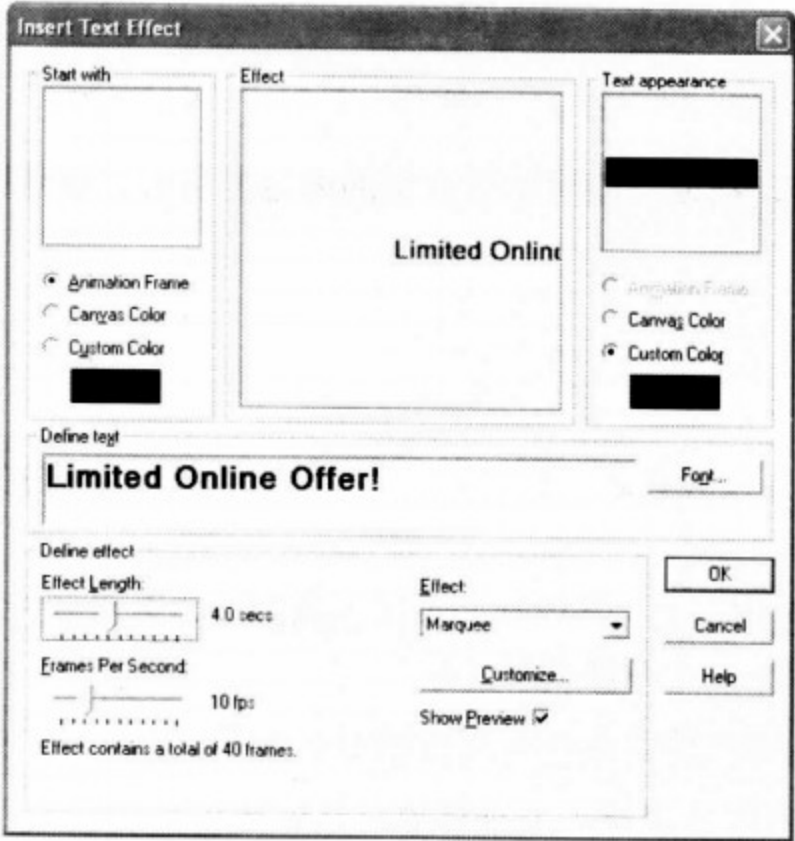


图 7.17  
使用 Effect/Insert Text Effect 在一帧和一系列帧中生成移动文本或特效文本

## 7.10 总结

本章读者学习了制作网页图像所需的基础知识；学习了如何下载流行的图像程序 Paint Shop Pro，并使用它来处理照片、横幅、按钮和其他网页图像（这些技巧也适用于众多其他的图像软件）；还学习了如何在各种用于网页图像的图像文件格式中进行选择，以及如何在图像质量和速度之间折衷。

## 7.11 问与答

- 问：是否应雇用美工来设计网页，而不是自己学习这些内容？

答：如果有足够多的钱且要求网站给人以深刻的视觉效果，或者你认为人们常常抱怨的带有黄绿色装饰的丑陋建筑实际上相当好看，那么雇用专业人员可能是个不错的主意。但要记住，你可能比任何人都更了解自己想要的，而这在创建优秀的网页方面常常比艺术技巧更重要。
- 问：我创作过用于打印的图像，制作网页图像是否与此截然不同？

答：是的。事实上，打印图像与网页图像的很多规则是相反的。网页图像要尽量使用低分辨率，而打印图像应尽量使用高分辨率。计算机屏幕是黑底白字，打印图像是白底黑字。另外，人们可能在图像只显示到一半时就决定不再查看网页了。应尽量避免设计打印图像的旧习惯。
- 问：我有一个 Windows AVI 视频剪辑，可以将它转换成 GIF 动画吗？

答：可以。只要用 Animation Shop 打开这个 AVI 文件，就可以将它转换成 GIF 动画（你可以选择是否减少帧数；建议每隔三帧采样一次，或使文件缩小合理的比例）。也可以直接将 AVI 文件嵌入到网页中，这将在第 19 章讨论。



## 7.12 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 7.12.1 测验

1. 假如扫描了一张马的照片，并要将它放到网页中，要将其文件设置为多大？使用什么文件格式来保存它？
2. 贵公司的徽标是一个黑色字母 Z，背景是一个红色圆圈。如果要将它放到网页中，应绘制或扫描为多大的文件？使用什么文件格式来保存？
3. 是否应将 100×50 像素的按钮图像保存为交错的 GIF 文件？
4. 如何将 GIF 动画从不断重复播放修改为只播放三次就停止？

### 7.12.2 答案

1. 可以小到 100×40 像素或大到 300×120 像素，这取决于图像对网页的重要性。最好使用 JPEG 格式，并将压缩比例设置为 50%。当然，也可以提供一个缩略图链接，它指向一个大型图像。你将在下一章学习如何将图像用作链接。
2. 对于徽标，通常大约 100×100 像素就可以了。但像这么简单的图像的压缩比例将很高；因此，如果愿意，可以使其大到 300×300 像素，保存为 16 色 GIF 文件。
3. 不。对于这么小文件，不使用交错时的加载速度与使用交错时一样快，甚至更快。
4. 使用 Animation Shop，选择 Animation/Animation Properties。单击 Loop 标签，选择 Play It，然后输入 3。

## 7.13 练习

1. 如果你有一些公司（或个人）的照片，从中找出一些可改善网站的照片。将这些照片扫描，这样就有了一个图像库，将来创建网页可以随时取用。如果有数码相机拍摄的照片，显然可以跳过扫描步骤，直接制作网页所需的图像。
2. 为重要的商业网站设计图像前，首先应试着使自己的个人主页丰富多彩。这让你有机会学习 Paint Shop Pro（或其他图像软件），在完成工作时让你显得了解要完成的任务。

## 第 8 章

# 将图像放到网页中

在第 7 章中，读者学习了如何为网页创建数字图像，包括静态图像和动画。本章将表明，使用 HTML 将这些图像放在网页中有多简单。将动画放在网页中与将静态图像放在网页中一样，并没什么特殊的地方。

### 练习

现在你已经准备好两三幅图像，可以在本章的学习中将它们放在网页中。如果你有 GIF、PNG 或 JPEG 格式的图像文件（文件扩展名分别为.gif、.png 和.jpg），可以使用这些文件。也可以使用在前一章创建的其他图像。

**提示：**在 Sams Publishing 网站，可以找到许多图像和多媒体热点列表和网站的链接，在这些网站上可以找到可用的图像。

**Did you  
Know?**

在有些 Web 搜索引擎和目录上，如 google.com、yahoo.com 和 lycos.com，也可以找到与所需主题相关的图像网站。它们还能帮助你找到许多专门提供免费或价格低廉的可复用媒体的网站。另外，不要忘记 Microsoft Office 在线剪贴画和媒体网站有大量的剪贴画库，其网址为 <http://office.microsoft.com/clipart/>。

## 8.1 将图像放到网页中

要将图像放在网页上，首先将图像文件放在 HTML 文本文件所在的文件夹下。将下列 HTML 标签插入到要显示图像的地方。用实际图像文件替换 myimage.gif:

```

```

<img /> 标签的 src 和 alt 属性是 XHTML 网页要求的。src 属性标识图像文件，alt 属性指定关于图像的描述性文本。当用户不能看到图像时，将看到 alt 属性指定的文本。在稍后的 8.2 节将学到关于 alt 属性的更多知识。

**By the Way**

**注意：**<img />标签也支持 title 属性，该属性用于描述图像。与 alt 属性不同的是，title 属性是在图像可见时提供图像描述。alt 属性的作用更为重要，主要是在图像不能显示时——如盲人用户“浏览”网页时——发挥作用。alt 属性是必需的，但同时提供 alt 和 title 属性更好，如果要保证图像在两种情况都有描述的话。

程序清单 8.1 是一个如何使用<img />标签的例子，它在页面的顶部和底部插入几幅图像。网页浏览器显示程序清单 8.1 所示的 HTML 页面时，将自动获取并显示图像文件，如图 8.1 所示。注意，这些图像是在上一章使用 Paint Shop Pro 创建和编辑的图像。

**程序清单 8.1 <img />标签用于将图像放在网页中**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael' s Pond</title>
  </head>

  <body>
    <p>
      
    </p>
    <p>
      My backyard pond is not only a fun hobby but also an ongoing home
      improvement project that is both creative and relaxing. I have numerous
      fish in the pond, all Koi from various places as far as Japan, Israel,
      and Australia. Although they don' t bark, purr, or fetch anything other
      than food, these fish are my pets, and good ones at that. The pond was
      built in a matter of weeks but has evolved over several years through
      a few different improvements and redesigns. I still consider it a work in
      progress, as there are always things to improve upon as I continue to
      learn more about how the pond ecosystem works, how to minimize
      maintenance, and how to get a more aesthetically pleasing look.
    </p>
    <p>
       <br />
       <br />
       <br />
      
    </p>
```

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

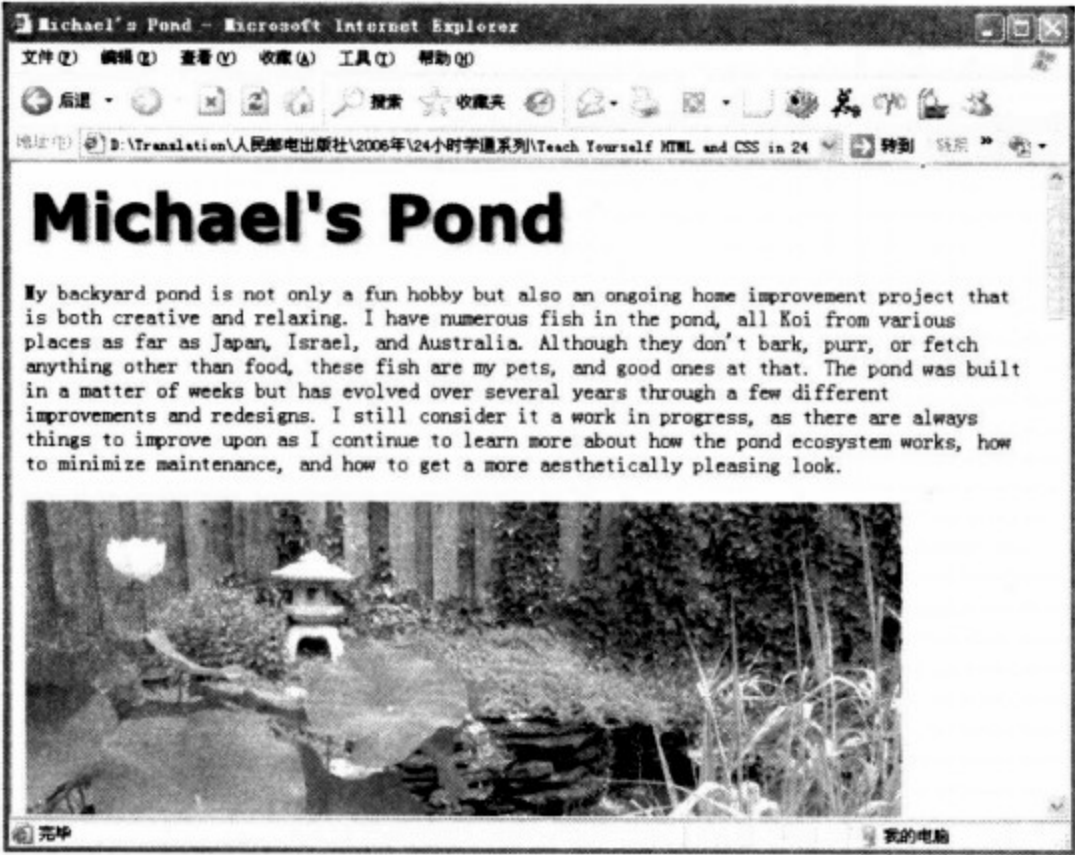


图 8.1  
网页浏览器显示  
程序清单 8.1 中的  
HTML 页面时，将  
添加 图 像  
pondtitle.gif 、  
pond1.jpg 、  
pond2.jpg 、  
pond3.jpg 和  
pond4.jpg

img 表示 image（图像），而 src 表示 source（源），也就是图像文件所在的位置。在第 1 章讨论过，网页图像总是存储在与网页文本独立的文件中，虽然它在浏览器中是作为网页的一部分显示的。

与<a href>标签（第 3 章介绍过）一样，可以在<img />标签的 src 属性中指定完整的 Internet 地址。如果图像与 HTML 文件在同一文件夹下，也可以只用图像的文件名。还可以使用相对地址，如 images/birdy.jpg 或 ../smiley.gif。

**注意：**从理论上说，可以在网页中包含来自任何 Internet 网页的图像。例如，可以在网页中加入下列代码，将我的一幅图像放在你的网页中：

```

```

浏览器要显示你的网页时，将从我的 Web 服务器上获取这幅图像。可以这样做，但不应该这样做！这不仅是一种不好的方法（通常，从他人服务器获取信息会花费他人的钱），而且这也会使你的网页显示得更慢。你也无法控制图像的修改或删除。

如果有人允许你发布他网页中的图像，将这些图像复制到你的计算机，并使用本地的文件拷贝，如。换句话说，你的网页中的所有图像应放在你的 Web 服务器中。

**By the  
Way**

## 8.2 使用文本描述图像

程序清单 8.1 中的每个<img />标签都包含一段简短的文本信息，如 alt="One of the Larger



Fish Cruises for Dinner”。alt 表示 “alternate text”（替换文本），因为它指定的消息将在用户关闭浏览器的自动图像下载功能时显示。其他人也能看到 alt 属性中的消息——因为有时图像文件在 Internet 上传输需要一段时间，大多数网页浏览器先在页面中图像的位置显示 alt 消息（如图 8.2 所示）。

图 8.2  
等待图像显示时用户将看到 alt 消息



即使图像取代 alt 消息后，将鼠标指向图像时，alt 消息通常显示在一个小框（工具提示）中。alt 消息也能帮助有视力障碍（或使用基于语音界面阅读网页）的人。

考虑到可能看到消息的用户情况是不同的，因此网页上的每个<img />标签必须包含一个合适的 alt 属性。通常最好使用对图像的简短描述，但网页制作者有时也在 alt 消息中放一些短小的广告消息或幽默笑话。对于小的或不重要的图像，完全可以省略 alt 属性，但这个属性是<img />标签要求的。这并不是说如果没有这个属性网页就不能正常显示，而是意味着违反了最新的 XHTML 标准。如果真的不需要这个属性，如对于一些小的或装饰性的图像，可以赋予它一条空文本消息（alt=""）。

还有一个 title 属性，它对<img />标签不是必需的。title 属性类似于 alt 属性，除了一点：它的显示与图像的可见性无关。然而，在两个属性都指定的情况下，在工具提示中 title 属性将覆盖 alt 属性。因此，通过文本来描述图像最好的解决方法是：使用 alt 属性存储对图像的简短文本描述，然后使用 title 属性来提供关于图像的相关注释或有用的提示作为浏览时的工具提示。

**Coffee Break**

**休息一下：**说到文本和图像，你可能没有意识到可以使用 Google 搜索 Web 上的图像。Google 的图像搜索服务位于 <http://www.google.com/imghp?hl=en&tab=wi&q=>。除 URL 比较奇怪外，Google 图像搜索是一个功能非常强大的搜索工具，可以搜索到一些有趣的结果。如果没什么要搜索的，试着搜索一下你的名字——搜索结果可能让你感到意外！



## 8.3 将图像变为链接

你可能注意到了，图 8.1 中页面上的图像很大，如果要显示多幅图像，这样的图像就不是很理想。这种情况下，最好创建较小的缩略图，链接到每幅图像的较大版本。这样就可以将缩略图安排在一个网页上，用户可以比较方便地看到所有内容，即使他们看到的只是要显示的一些样本。缩略图是使用图像链接改善网页的一种方式。

可以像创建文本链接一样，使用<a href>标签将任何图像变为指向另一个网页或图像的可单击链接。程序清单 8.2 包含了修改后的池塘网页的代码，它利用图像链接和缩略图来改进网页布局。网页如图 8.3 所示。

程序清单 8.2 缩略图是有效使用图像链接的成功例子

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

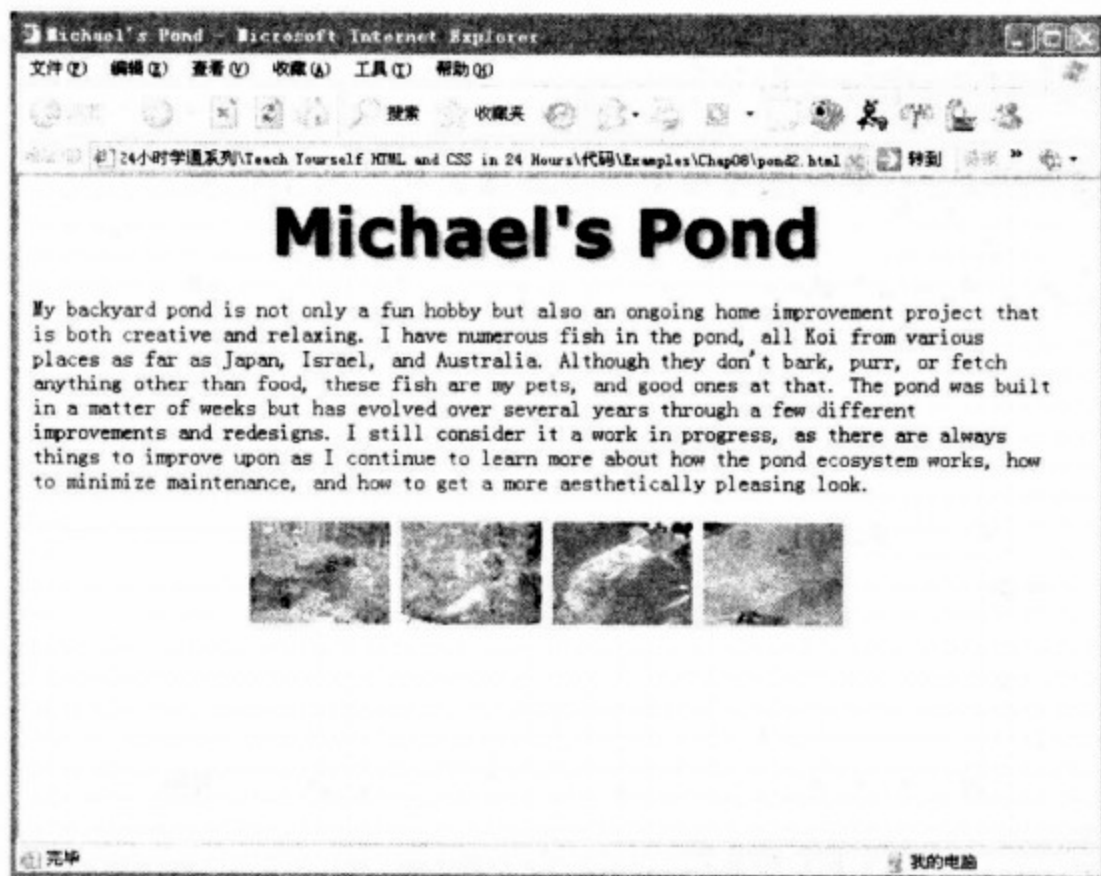
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael' s Pond</title>
  </head>

  <body>
    <p style="text-align:center">
      
    </p>
    <p>
      My backyard pond is not only a fun hobby but also an ongoing home
      improvement project that is both creative and relaxing. I have numerous
      fish in the pond, all Koi from various places as far as Japan, Israel,
      and Australia. Although they don't bark, purr, or fetch anything other
      than food, these fish are my pets, and good ones at that. The pond was
      built in a matter of weeks but has evolved over several years through
      a few different improvements and redesigns. I still consider it a work in
      progress, as there are always things to improve upon as I continue to
      learn more about how the pond ecosystem works, how to minimize
      maintenance, and how to get a more aesthetically pleasing look.
    </p>
    <p style="text-align:center">
      <a href="pond1.jpg"></a>
      <a href="pond2.jpg"></a>
      <a href="pond3.jpg"></a>
<a href="pond4.jpg"></a>
</p>
</body>
</html>

```

图 8.3

使用缩略图链接可以显著地改善包含大型图像的网页的布局



正如代码显示的，这个网页版本并没有太大的改变，只是现在每个池塘图像都使用一个链接标签。另外，现在这个网页上的图像都是缩略图，这通过它们的文件名很容易看出来（pond1\_sm.jpg、pond2\_sm.jpg 等）。文件名中的“\_sm”表示“small”，而<a>标签引用的是大型图像。

通常，网页浏览器在每幅图像链接的四周绘制一个彩色矩形。与文本链接一样，通常矩形在还没有访问链接前是蓝色的，而访问后变为紫色。因为通常都不希望显示这个难看的边框，因此应在<img />标签中包含样式 style="border-style:none"。你将在第 13 章学习到 border 属性。

### Did you Know?

**提示：**在网页中使用图像链接的一个实际的例子是 PayPal 的 Buy Now 按钮。这种按钮让用户能够通过 PayPal 向你购买东西，PayPal 负责收钱并将购买信息通知你。PayPal 的 Buy Now 按钮涉及到表单的使用，这将在第 18 章介绍。你还将在第 18 章学习如何创建和使用 PayPal Buy Now 按钮。

第 3 章和第 6 章介绍的文本链接的链接规则和用法，同样也适用于图像链接。例如，也可以使用<a href="name">链接到同一页面的另一个部分。

单击池塘网页中的缩略图时，浏览器将打开完整的图像，如图 8.4 所示。

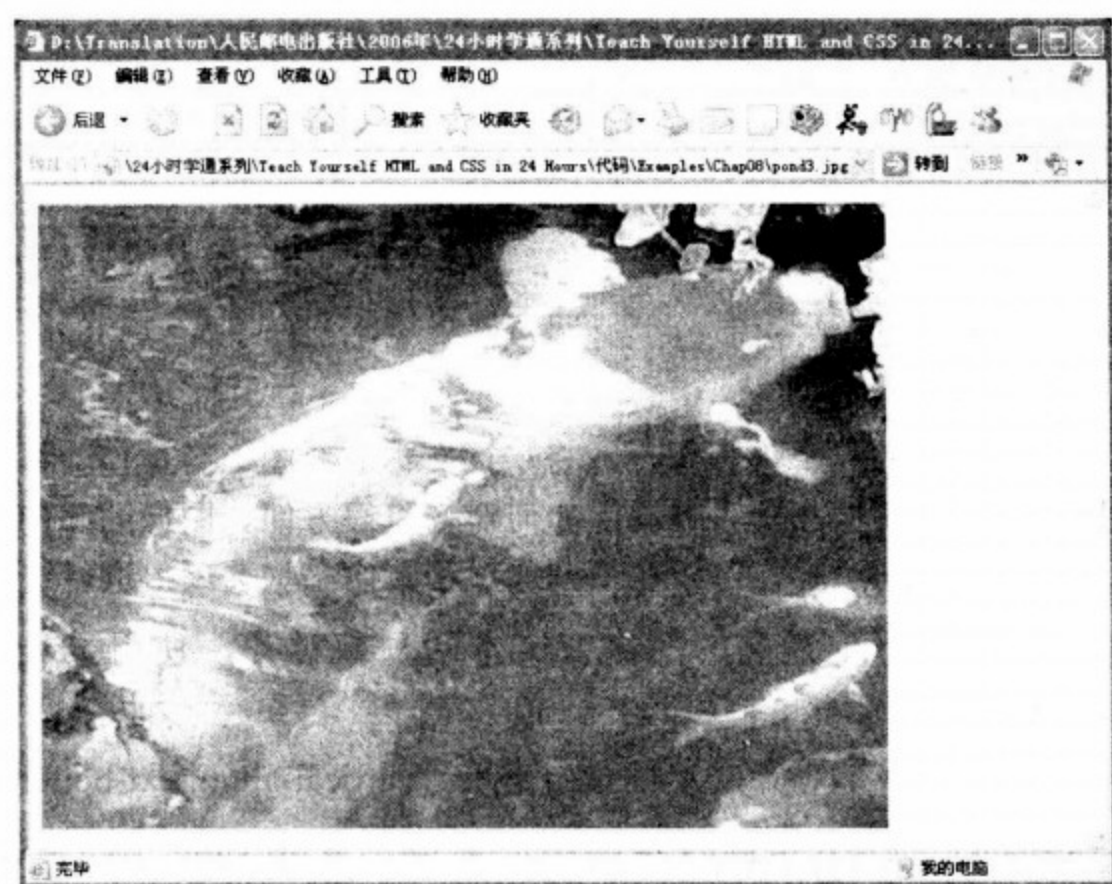


图 8.4  
单击缩略图链接，  
将在浏览器中打  
开大型图像

## 8.4 对齐图像

与文本类似，也可以使用特殊的属性来对齐网页中的图像。不仅可以水平对齐图像，还可以垂直地将图像与包围图像的文本或其他图像对齐。

### 8.4.1 水平对齐图像

第 5 章讨论过，可用<div style="text-align:center">、<div style="text-align:right">和<div style="text-align:left">将元素居中对齐、右对齐或左对齐。这三个样式设置可用于文本和图像，也可用于<p>标签。

例如，程序清单 8.2 中的第一个<img /> 标签位于<p style="text-align:center">标签和结束标签</p>之间，这使得标题图像位于页面中央。与文本一样，图像通常是左对齐的，除非设置了 style="text-align:center"或 style="text-align:right"，指定居中对齐或右对齐。换句话说就是，left 是 text-align 的默认值。

程序清单 8.2 和图 8.3 显示了最后四幅图像，可以看出，可用<div style="text-align:center">一次将多幅图像居中对齐。因为图像之间没有<br />标签或<p>标签，因此四幅图像都显示在一行上，这一行在浏览器窗口中水平居中。

也可以在<img />标签内直接使用 float 样式，使文本围绕图像。下面便是一个例子：

```



<img style="float:left" />使图像左对齐，文本围绕在图像的右边。类似的，
```

`<imgstyle="float:right" />`使图像右对齐，文本围绕在图像的左边。没有将图像浮动在中央的概念，因为这样就无法确定文本应显示在图像的哪一边。

### By the Way

**注意：**float 样式属性的功能实际上比这里介绍的还要强大，它不仅可用于图像。也可使用 float 样式属性创建一些有趣的页面布局，这将在第 14 章介绍。

## 8.4.2 垂直对齐图像

有时可能希望在一行文本中插入一个小图像，或将一行文本放在图像旁边作为标题。在这些情况下，如果可以控制文本和图像在垂直方向上如何对齐将很方便。图像的底部与文本的底部对齐，还是文本和图像的中间对齐？可以在下列选项和其他一些选项中进行选择。

- 要使图像的顶部与同一行中最高的图像或文本的顶部对齐，使用`<img style="vertical-align:text-top" />`。
- 要使图像的底部与文本的底部对齐，使用`<img style="vertical-align:text-bottom" />`。
- 要使图像的中心与同一行中其他内容的中心对齐，使用`<img style="vertical-align:middle" />`。
- 要使图像的底部与文本的基线对齐，使用`<img style="vertical-align:baseline" />`。

### By the Way

**注意：**vertical-align 样式属性也支持 top 和 bottom 这两个值，这两个值分别用于将一行中图像的顶部和底部都对齐，而不管这一行中的文本。

程序清单 8.3 和图 8.5 演示了这四个选项的用法。现在四个池塘缩略图排列在页面下半部分，每幅图像旁边都有一些描述性文本。每幅图像都使用不同的 vertical-align 样式属性来对齐图像与文本。

程序清单 8.3 这个例子演示了水平对齐、垂直对齐和文本围绕

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael's Pond</title>
  </head>

  <body>
    <p style="text-align:center">
      
    </p>
    <p>
      My backyard pond is not only a fun hobby but also an ongoing home
      improvement project that is both creative and relaxing. I have numerous
```



```
fish in the pond, all Koi from various places as far as Japan, Israel,
and Australia. Although they don't bark, purr, or fetch anything other
than food, these fish are my pets, and good ones at that. The pond was
built in a matter of weeks but has evolved over several years through
a few different improvements and redesigns. I still consider it a work in
progress, as there are always things to improve upon as I continue to
learn more about how the pond ecosystem works, how to minimize
maintenance, and how to get a more aesthetically pleasing look.
</p>
<p>
  <a href="pond1.jpg"></a>
  The Lotus, Floating Hyacinth, Japanese Lantern, and Waterfall All Add to
  the Drama of the Pond<br/>
  <a href="pond2.jpg"></a>
  Feeding Time is Always Full of Excitement<br/>
  <a href="pond3.jpg"></a>
  One of the Larger Fish Cruises for Dinner<br/>
  <a href="pond4.jpg"></a>
  A Dragonfly Hovers Over the Lotus for a Quick Drink<br/>
</p>
</body>
</html>
```

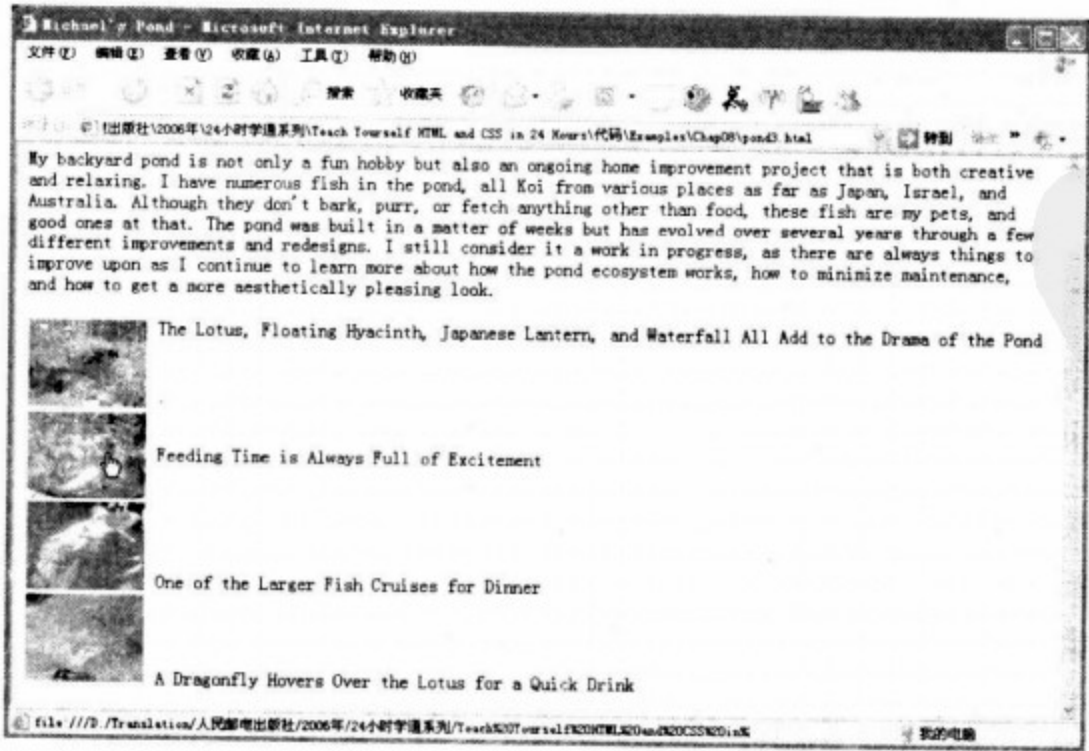


图 8.5  
程序清单 8.3 所列  
的新修改的池塘  
网页显示在网页  
浏览器中的样子



**Did you know?**

**提示：**如果在标签中不包含 align 属性，图像的底部与文本的基线对齐。也就是说，实际上不必输入 style="vertical-align:baseline"，因为这是默认设置。然而，如果想为图像指定一个边距并更精确地对齐文本，可以显式地将 vertical-align 设置为 text-bottom。看图 8.5 中的最后一幅图像，由于图像设置了边距，且将 vertical-align 设置为 baseline，因此文本的位置低于图像的底部。

注意，图 8.5 中有一个细微的变化，就是新的 title 属性，现在用于为所有图像显示消息“Click to view larger picture”。因为现在网页上文本描述与图像出现在一起，因此这样的改变是合理的。现在每幅图像必需的 alt 属性包含一个简单描述。

title 属性的这种用法，解决了一些网页开发人员使用 alt 属性只为了提供工具提示的问题。问题在于有些浏览器显示一个包含 alt 文本的弹出窗口（工具提示）。然而，只要使用 title 就可以实现这个目的，就像这个例子一样，然后使用 alt 属性实现其原来的目的，即提供用于替代图像的文本。可能的话，应尽量同时使用 alt 和 title 属性。



现在在网页中添加一些图像，并尝试使用 text-align、vertical-align 和 float。开始之前，先看看如何在网页中添加一张鱼的图像（fish.jpg）。

- 将图像文件 fish.jpg 复制到 HTML 文件所在的目录下。
- 使用文本编辑器，在 HTML 文件中要显示图像的位置添加![Fishy](fish.jpg)。
- 如果要使图像居中，在前添加

>，然后在它后面加上</div>。要使文本围绕图像，在标签中添加 style="float:right" 或 style="float:left"。最后，直接在标签中使用 vertical-align 来控制图像与相邻图像或文本之间的垂直对齐。
- 如果还有时间的话，可以试着将多个大小不同的图像以不同的垂直对齐设置进行组合。



## 8.5 指定图像的宽度和高度

因为文本在 Internet 上的传输比图像快，因此大多数网页浏览器将在图像之前显示网页的文本。这让用户在看到图像前有一些文本可看，使整个页面的显示显得更快些。

可以显式地指定每幅图像的宽度和高度，确保网页可以尽可能快地显示。这样的话，网页浏览器可以在加载网页，等待图像传输完毕的同时，为每幅图像留出准确的空间。

**Did you know?**

**提示：**虽然指定图像的宽度和高度有助于网页浏览器更快地显示图像，然而，如果在制作网页的过程中图像的大小不能确定，这就比较痛苦了。修改网页中的图像时，你肯定不希望不断修改图像的 width 和 height 属性。因此，在标签指定 width 和 height 属性，可作为网页已经准备好要发布前的一种优化方法。

使用 Paint Shop Pro 或其他图像软件确定网页中每幅图像的宽和高——以像素为单位。在 Paint Shop Pro 中，将鼠标指向图像时，主程序窗口的右下角将显示这种信息。然后在<img />标签中包含这些尺寸，如下所示：

```

```

**提示：**为图像指定的 width 和 height 不一定要与图像实际的宽和高相等。不相等时，网页浏览器将自动尝试将图像缩小或拉伸到指定的大小。然而，通常这样做并不好，因为浏览器在调整图像大小方面做得并不好。如果要使图像显示得小一点，最好在图像编辑器中调整好它的大小。

如果允许网页浏览器调整图像的大小，通常图像会由于压缩或拉伸而变得很难看，但在一种情况下该功能很有用：可以保存一个完全透明的小型图像，然后将它的 width 和 height 指定为要创建的白色区域的宽和高，将其用作隔板。这是网页开发人员在设计复杂页面布局时使用的一种技巧。

**Did you know?**

## 8.6 总结

本章介绍了如何使用<img />标签将图像放到网页上。读者学习了如何包含一个简短的文本信息，用于在图像载入时以及鼠标指向图像时显示。还学习了对每幅图像的横向与纵向排列进行对齐，以及如何使文本在图像左边或右边环绕。

最后，学习了如何利用缩略图链接到更大的图像，作为一种改进网页布局的方法。这需要使用第 3 章介绍的<a>标签。读者还没有学到如何将图像用作自定义的网页背景，这将在第 9 章介绍。

表 8.1 总结了这一章介绍的<img />属性及相关的样式属性。

表 8.1	第 8 章介绍的 HTML 标签与属性
标签/属性	功能
<img />	将图像放在网页中
属性	
src="地址"	图像的地址或文件名
alt="取代图像的描述"	显示在图像位置上的对图像的描述，主要提供给不能看到图像的用户看的
title="标题"	作为图像标题显示的文本信息，通常显示在图像上的一个弹出窗口（工具提示）中
width="宽度"	图像的宽度，单位是像素
height="高度"	图像的高度，单位是像素
style="border-style:none"	如果图像是链接，就去掉图像周围的边框
style="vertical-align:alignment"	图像在垂直方向上按 text-top、top、text-bottom、bottom、middle 或 baseline 对齐
style="float:float"	使图像浮动在一边，文本环绕在另一边。可选的值有 left、right 和 none（默认值）

## 8.7 问与答

问：我在 Internet 的网页上发现一个很好的图像。我可以直接使用“图片另存为”将图

像保存下来，然后将它放到我的网页中吗？

答：这样做是很容易的，但不幸的是，如果没有使用图像的许可权，在许多国家这样做是非法的，除非这些图像来自无版权的剪贴画网站。首先应该得到图像原创者的书面许可。大部分网页都包含作者的 Email 地址，很容易向作者请求许可——比上法庭简单多了。

问：放在<img />标签的 alt=后面的信息应多长？

答：从理论上说，想多长都可以。实际上，应使信息足够短，使其可以在图像本身的空间内显示。对于大图像，10 个词比较好，但我也见过有人放置一小段文本。对于小图像，一个词就好了。

问：我按照你所说的那样使用了<img />标签，但浏览网页时只看到一个包含“¥”或类似的形状的小框。这是为什么呢？

答：你看到的是图像受损标记，可能是因为以下两种情况之一：网页浏览器找不到图像文件；浏览器不能理解图像文件的格式。要解决这个问题，先使用 Paint Shop Pro（或者你喜欢的图像软件）打开图像文件，选择“另存为”，将文件以 GIF 或 JPEG 格式保存。同时，将文件保存在包含引用它的<img />标签的网页所在的文件夹下，还要确保磁盘中的文件名与<img />中的文件名完全一致（包括大小写）。将这幅图像文件保存到网页所在的文件夹里；如果将图像放在不同的文件夹，要确保使用的相对文件夹是正确的。

问：在本书的例子中，为什么每个<img />标签的结尾都包含一个斜杠？我在 Internet 上看到的网页都没有这样做的。

答：正如在第 2 章讨论的（是的，很久以前的事了），XTMHL 标准要求任何空标签（没有结束标签的标签）都要在结尾包含一个斜杠。虽然将来不太可能有人编写不接受传统的无斜杠<img>标签的程序，但应该使用<img />，使网页能够适应不断发展的 Web。人们曾经以为，在编写程序时，不需要使用四位数的年份编码。

## 8.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 8.8.1 测验

1. 如何将图像文件 elephant.jpg 插入到网页的顶部？
2. 如何在网页浏览器不能显示图像 elephant.jpg 时显示单词 Elephant？
3. 编写 HTML 代码，使图像 elephant.jpg 显示在网页的右边、大标题“Elephants of the World Unite!”显示在图像的左边。
4. 编写 HTML 代码，使一幅小的老鼠图像（名称为 mouse.gif）显示在“Wee sleekit, cow'rin”和“tim'rous beastie”之间。

## 8.8.2 答案

1. 将这幅图像文件复制到 HTML 文本文件所在的文件夹，然后在 HTML 文本文件的 `<body>` 标签后输入 `<p></p>`。

2. 使用以下 HTML:

```

```

3.

```

<h1>Elephants of the World Unite!</h1>
```

4.

```
Wee sleekit, cow'rin,<img'rin,tim'rous
beastie
```

## 8.9 练习

使用任何小型图像作为项目符号，使列表更好看。如果还想使列表缩进，使用 `<dl>` 定义列表，对每一项使用 `<dd>`（而不使用 `<ul>` 和 `<li>`，那样将生成单调的标准项目符号）。下面是一个例子，它使用图像文件 `star.gif` 作为项目符号：

```
<dl><dd>A murder of crows</dd>
<dd>A rafter of turkeys</dd>
<dd>A muster of peacocks</dd></dl>
```

# 第 9 章

## 自定义背景和颜色

从第 1 章到第 8 章，几乎每个网页都使用白色背景和黑色文本。本章将学习如何指定网页的背景颜色和文本颜色；还将学习如何创建自定义的背景图像，如何使背景透过网页中的图像显示出来。

本书印刷出来的黑白照片显然不能准确地说明颜色，因此读者需要在线查看示例网页。也可以在学习过程中在自己的网页中尝试各种颜色。

**Did you know?** 提示：在 <http://www.sampublishing.com/> 可以找到本章的所有示例。

### 9.1 背景颜色和文本颜色

将网页的背景颜色指定为白色以外的颜色，比你想的还要简单。例如，要将网页的背景颜色指定为蓝色，只要在<body>标签中加上 `style="background-color:blue"`。当然，可以使用各种颜色。可以从标准的颜色名称列表中选择颜色，包括极具描述性的 `BurlyWood`、`LavenderBlush` 和 `MistyRose`。这听起来更像是设计师的颜料盒。当然，也可以使用 16 种标准的 Windows 颜色：`Black`、`White`、`Red`、`Green`、`Blue`、`Yellow`、`Magenta`、`Cyan`、`Purple`、`Gray`、`Lime`、`Maroon`、`Navy`、`Olive`、`Silver` 和 `Teal`。如果想显得更有艺术气息，可用 `Fuchsia` 来代替 `Magenta`，用 `Aqua` 来代替 `Cyan`。

**Did you know?** 提示：指定颜色名称时，大小写无关紧要。因此，使用 `Black`、`black` 和 `BLACK` 都可以，大多数网页设计人员都坚持使用全部小写或首字符大写。我个人更偏向于使用小写，因为这样代码显得更清晰。

也可以在<body>标签中指定文本颜色。例如，程序清单 9.1 使用了如下所示的<body>标签：



```
<body style="background-color:teal; color:fuchsia">
```

读者可能猜到了, color:fuchsia 将网页的文本颜色指定为 fuchsia (紫红色) 与 magenta (洋红色) 相同。当 color 属性放在<body>标签内时, 它影响该网页的所有文本。可以在网页的特定元素中设置 color 属性, 覆盖<body>标签中的设置。

**提示:** 有关可用于在 HTML 代码中指定颜色的所有颜色名称, 可参阅 [http://www.w3schools.com/css/css\\_colornames.asp](http://www.w3schools.com/css/css_colornames.asp)。该网页列出了每种颜色的名称及其对应的十六进制表示(稍后将进一步介绍)和在屏幕上的显示情况。

**Did you  
know?**

也可以修改文本链接的颜色, 包括链接的各种属性, 如用户将鼠标指向链接时链接的颜色, 以及用户单击链接时链接的颜色。设置文本链接的颜色需要使用一个内部的样式表, 这将在第 12 章介绍。本章将学习如何通过 CSS 将颜色应用于文本链接的所有细节。

程序清单 9.1 和图 9.1 演示了如何设置网页的背景颜色和文本颜色。因为图像以纯青鳉 (teal) 色为背景色, 因此它们完全融入到网页背景中 (不需要使用透明图像, 本章稍后将介绍透明图像)。

#### 程序清单 9.1 在<body>标签中指定背景颜色和文本颜色

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

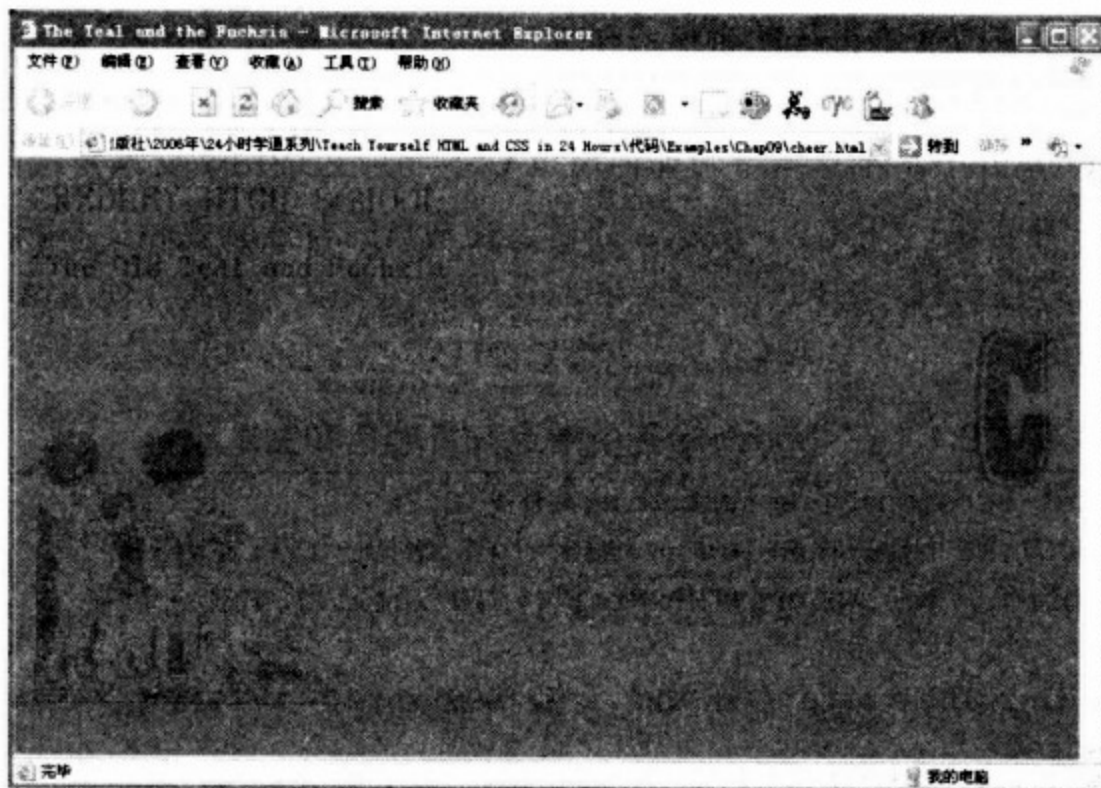
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Teal and the Fuchsia</title>
  </head>

  <body style="background-color:teal; color:fuchsia">
    <h1>CREDLEY HIGH SCHOOL</h1>
    <h2>"The Old Teal and Fuchsia"</h2>
    <div style="text-align:center">
      <p style="font-style:italic">
        
        Oh, hail! Hail! Sing Credley!<br />
        Our colors jump and shout!<br />
        Deep teal like ocean's highest waves,<br />
        Fuchsia like blossoms bursting out!
      </p>
      <p style="font-style:italic">
        
        As Credley conquers every team<br />
        So do our brilliant colors peal<br />
        From mountain tops & florist shops<br />
        Sweet sacred fuchsia, holy teal!
      </p>
      <p style="font-style:italic">
```

```
Our men are tough as vinyl siding<br />
Our women, strong as plastic socks<br />
Our colors tell our story truly<br />
We may be ugly, but we rock!
</p>
</div>
</body>
</html>
```

图 9.1

在彩色屏幕上，这个原本很美的网页背景为青灰色，正文为紫红色



这个网页是一个反面的示例，有意将网页颜色弄得很糟。幸运的是，修改<body>标签中的两个颜色，可以改变整个网页的外观。也就是说，一行代码就足显著改变网页的外观。

## 9.2 指定自定义颜色

如果没有任何颜色名称能够提供想要的颜色，可以使用自定义颜色，这时需要指定颜色由多少红、绿、蓝混合而成。这有点像调音，将适量的红、蓝和绿混合在一起，就能得到想要的颜色。

自定义颜色的格式为#rrggbb，其中rr、gg和bb分别是表示颜色中的红、绿、蓝三个分量的两位十六进制数。如果不熟悉十进制数，不必担心。只要记住FF是最大值，00是最小值，在指定颜色分量时遵循下列几点：

- FF 表示完全的亮度；
- CC 表示 80%的亮度；
- 99 表示 60%的亮度；
- 66 表示 40%的亮度；
- 33 表示 20%的亮度；

➤ 00 表示没有这种颜色成分。

例如，亮红色为#FF0000，深绿色是#003300，蓝紫色是#660099，而中灰色是#999999。要将网页的背景色为红色，文本色为深绿色，使用如下所示的 HTML 代码：

```
<body style="background-color:#FF0000; color:#003300">
```

虽然在<body>标签中指定的颜色将应用于网页上所有的文本，但可以在<span>标签内使用颜色名称或十六进制颜色编码来修改 color 样式属性，从而改变特定单词或段落的颜色。这将在第 12 章介绍。

**提示：** <http://www.sampublishing.com/>列出了 216 种最常用的十六进制颜色编码及对应的颜色。

**Did you know?**

应该知道，不同的计算机显示器可能以不同的色调显示颜色。我最近为客户设计了一个使用漂亮蓝色背景的网页，结果发现该公司的总裁在其计算机上看到的是很可爱的紫色背景。接近泥土的中性色，如中灰色、褐色和象牙色，在许多计算机显示器上的显示结果更不可预测，且显示器上的颜色可能根据房间的采光条件和时间而改变。

这里要说明的一点是：尽量使用颜色名称，除非能够精确地控制目标用户的计算机显示器或愿意在各种不同的显示器上进行测试。当然，在不同的计算机上测试网页很重要，也应该这么做。

## 9.3 使用背景图像拼贴

背景拼贴让你能够指定一个图像，用作文档中所有文本和图像背后的墙纸。可以在页面开头的<body>标签中使用样式属性 background-image 指定背景图像文件名：

```
<body style="background-image:url(backimage.jpg)">
```

与其他网页图像一样，背景拼贴必须是 GIF、PNG 或 JPEG 格式，可以使用图像软件来创建背景图像。例如，程序清单 9.2 中<body>标签引用的 water.jpg 就是一个带水纹的小型背景图像。如图 9.2 所示，大多数网页浏览器将在网页中的文本和图像背后重复显示这个图像，就像地板上的瓷砖一样。

程序清单 9.2 在<body>标签中使用样式属性 background-image 平铺背景图像

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael's Pond</title>
  </head>

  <body style="background-image:url(water.jpg)">
```

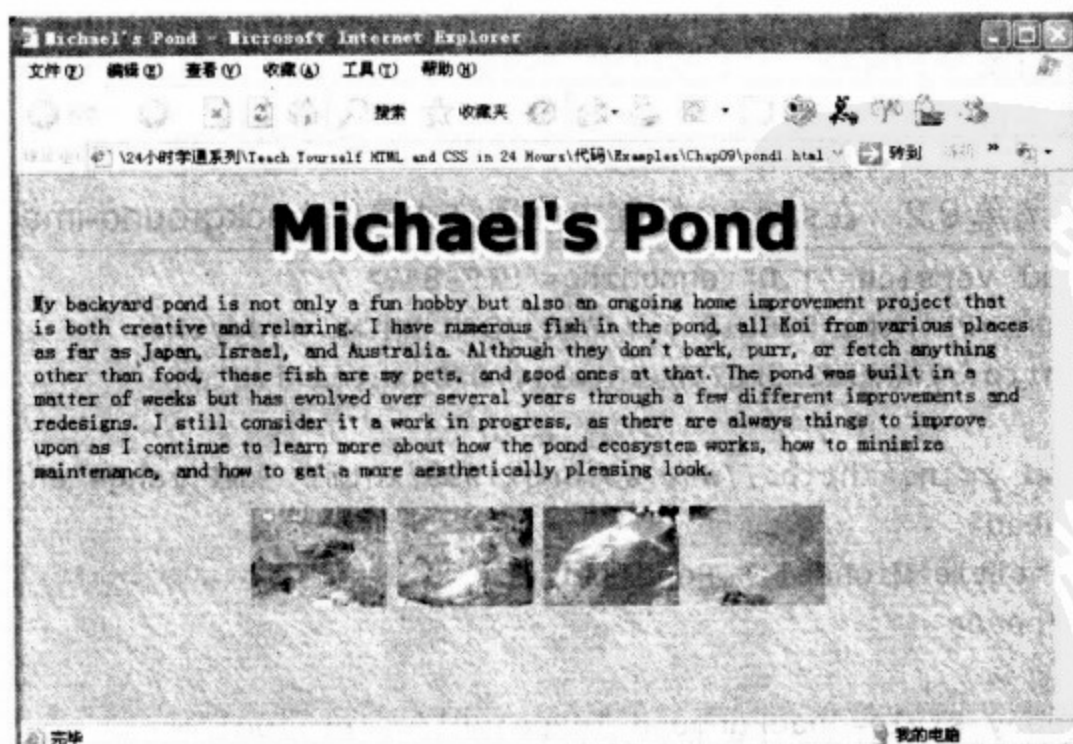
```

<p style="text-align:center">
  
</p>
<p>
  My backyard pond is not only a fun hobby but also an ongoing home
  improvement project that is both creative and relaxing. I have numerous
  fish in the pond, all Koi from various places as far as Japan, Israel,
  and Australia. Although they don't bark, purr, or fetch anything other
  than food, these fish are my pets, and good ones at that. The pond was
  built in a matter of weeks but has evolved over several years through
  a few different improvements and redesigns. I still consider it a work in
  progress, as there are always things to improve upon as I continue to
  learn more about how the pond ecosystem works, how to minimize
  maintenance, and how to get a more aesthetically pleasing look.
</p>
<p style="text-align:center">
  <a href="pond1.jpg"></a>
  <a href="pond2.jpg"></a>
  <a href="pond3.jpg"></a>
  <a href="pond4.jpg"></a>
</p>
</body>
</html>

```

图 9.2

water.jpg 文件(该文件在程序清单 9.2 中指定, 图 9.3 单独显示了这个图像)自动重复显示, 覆盖整个网页背景





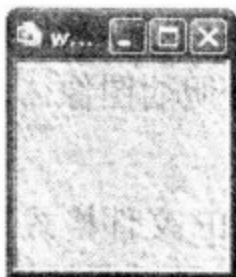


图 9.3  
背景图像 water.jpg  
很小，但当它重复  
排列时（见图 9.2）  
几乎是无缝的

**注意：**你可能在想，如果同时设置了网页的背景颜色和背景图像会怎样。背景图像将覆盖背景颜色，然而，如果背景图像中有透明区域，背景颜色可以透过这些区域显示出来。因此，如果想利用背景图像中的透明区域，可以同时使用这两个样式。

By the  
Way

**警告：**设计背景图像拼贴时应注意不要对网页本身的主要内容造成干扰。Web 上的许多网页由于夸张的背景几乎无法阅读。  
如果你计划将公司徽标或婴儿照片作为网页的墙纸，要停下来想一想。如果你用纸发送重要的消息给别人，会把它写在带徽标背景的纸上还是空白纸上？背景应与好的纸一样：美但不造成干扰。

Watch  
Out!

如图 9.2 所示，池塘网页因水的背景图像而显得更有趣。然而，现在标题图像的四周出现了有趣的白色边缘，这显得很不合适。出现这个问题是因为 GIF 图像只支持一种透明的颜色，这不能适应不同色调的文本投影。这需要使用第 7 章介绍的 PGN 图像，下一小节将这样做。

还有一个样式属性也会影响平铺的背景图像，这就是 background-repeat 属性。它决定背景图片在网页中如何平铺。这个样式可能的取值包括 repeat、repeat-x、repeat-y 和 no-repeat。默认为 repeat，它使图像在 X 方向和 Y 方向都重复；这也说明了我在这个示例中为什么不必使用这个样式属性。repeat-x 和 repeat-y 设置分别允许图像沿着网页的水平方向和垂直方向重复。最后，no-repeat 设置使背景图像只在页面的左上角显示一次。

下面是在页面的水平方向平铺背景图像的一个示例：

```
<body style="background-image:url(backimage.jpg);  
background-repeat:repeat-x">
```

**提示：**你可能认为将 background-repeat 设置为 no-repeat 没有意义。然而，有时候可能遇到一些不寻常的网页设计，要求使用很大的背景图像，这样就不需要平铺图像。这时就可以使用 no-repeat 设置。

Did you  
Know?

## 9.4 使用透明的图像

本章稍后你将学习如何创建自己的背景拼贴，但现在先说一下如何让背景透过前景的一些部分显示出来。



网页图像总是矩形的。然而，聪明的读者将注意到，在图 9.2 中，背景拼贴通过标题图像的一部分显示出来，因此背景图像看起来并不是规则的。这是因为标题图像中部分是透明的，从而使背景显露出来。通常可以使用部分透明的图像，使图像在任何背景颜色或背景图片下看起来更好。

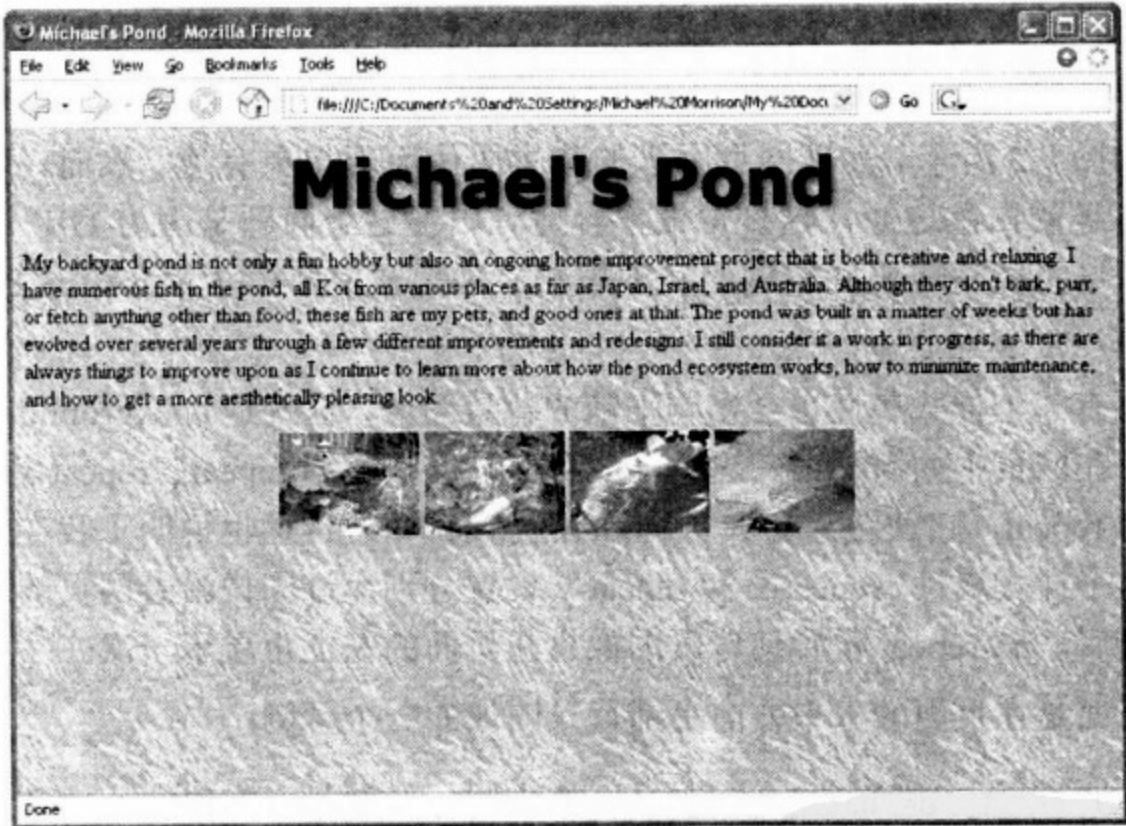
要使图像部分透明，图像必须是 PNG 或 GIF 文件格式（JPEG 图像无法变成透明的）。大多数支持 GIF 格式的图像软件让你能够指定一种透明颜色，而 PNG 图像则可以指定透明范围。PNG 优于 GIF，很大程度上就是因为这一点。最新的网页浏览器都支持 PNG 图像。关于 PNG 图像格式的详情，请参阅 <http://www.libpng.org/pub/png/pngintro.html>。

在上一节的池塘网页中，标题图像为透明的 GIF。然而，由于 GIF 图像只支持一种透明颜色，因此阴影不能达到希望的效果。透明的 GIF 图像通常在只有一种背景图像或颜色下效果才比较好。在这个示例中，池塘标题在白色背景下的效果很好，而在平铺图像背景下的效果不那么好。

解决办法是平铺背景使用 PNG 图像，它支持不同级别的透明度，而不只是一种透明颜色。图 9.4 显示了使用 PNG 版本的背景图像的池塘网页。

图 9.4

标题图像使用透明度得到改善的 PNG 格式，阴影效果更好，不受背景的影响



该图表明，现在阴影允许背景图像显露出来，这才是我们希望的效果。透明 GIF 和透明 PNG 之间的区别看似很小，但对网页的视觉效果可能有很大的影响。

**Coffee Break**

**休息一下：**在在线地图工具 Google Maps (<http://maps.google.com/.Zoom>) 上可以看到 PNG 透明图像示例。找到你家乡的地图并放大，然后输入要搜索的东西，如 pizza。屏幕上将出现多个标记，指出符合搜索请求的位置。单击其中的一个标记，将打开一个弹出窗口显示更多的信息。看到弹出窗口的漂亮阴影了吗？这就是一个透明 PNG 图像！这个示例中的效果非常好，因为所有底层的地图细节都可以通过阴影显示出来。

按下列步骤，在 Paint Shop Pro 中保存透明 GIF。

1. 选择吸管工具，右击要使其变为透明的颜色。
2. 选择 Image/Palette/Set Palette Transparency。如果图像的颜色超过 256 种，程序提示将把颜色数减少到 256，单击 OK 即可。
3. 将打开如图 9.5 所示的对话框。选择 Set the Transparency Value to the Current Background Color，然后单击 OK。

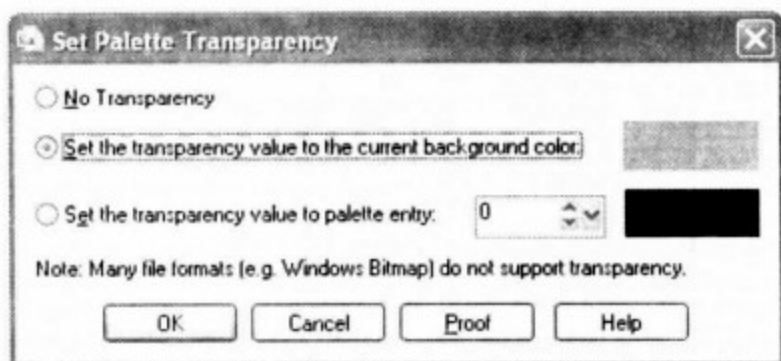


图 9.5

选择 Image/Palette/Set Palette Transparency 将打开该对话框，通常选择中间的单选按钮

4. 选择 Image/Palette/View Palette Transparency，图像的透明部分将变为灰色棋盘图案。
5. 可以使用任何画图工具对图像中透明背景色太少或太多的部分进行修饰。
6. 完成后，选择 File/Save As（或 File/Save Copy As），并将文件类型指定为 CompuServe Graphics Interchange (\*.gif)。

如果选择 Color/ View Palette Transparency（或单击如图 9.5 所示的 Proof 按钮），Paint Shop Pro 将以灰色棋盘图案显示图像中的透明部分。可以通过菜单 File/Preference/General Programm Preference/Transparency 修改网格的大小和颜色。

PNG 图像的透明度问题更复杂些，因为它不仅涉及一种透明颜色。使用 PNG 图像的技巧是，确保创建图像时背景本身是无色的。例如，在 Paint Shop Pro 中创建图像时，一定要将背景指定为透明的。这样，使用的任何有不同透明度的效果，如阴影，都将自动考虑背景是透明的。

## 9.5 创建自己的平铺背景

任何 GIF 或 JPEG 图像都可用作背景图片。然而，背景图像的顶部与底部、左边与右边能够无缝衔接时，网页效果最好。

如果你很聪明且有时间，可通过小心地剪切和粘贴，并对边缘进行修饰，将任何图像变为无缝衔接的拼贴。Paint Shop Pro 提供了一种更简单的方法，可自动将任何纹理变为无缝衔接的拼贴：只要裁剪出要作为拼贴的矩形区域，然后选择“Effects/Effects/Image Effects/Seamless Tiling（效果/图像效果/无缝平铺）”。Paint Shop Pro 裁剪出图像，使用复杂的自动处理将图像的边缘重叠，并使衔接的两边模糊。



图 9.6 显示了 Paint Shop Pro 中将石灰石照片转换为无缝拼贴时的 Seamless Tiling 对话框。得到的拼贴（用作图 9.7 所示网页的背景）无缝衔接，但仍保留自然石灰石的色泽和纹理。

图 9.6

Paint Shop Pro 自动将图像的任何部分变为易于平铺的背景图案

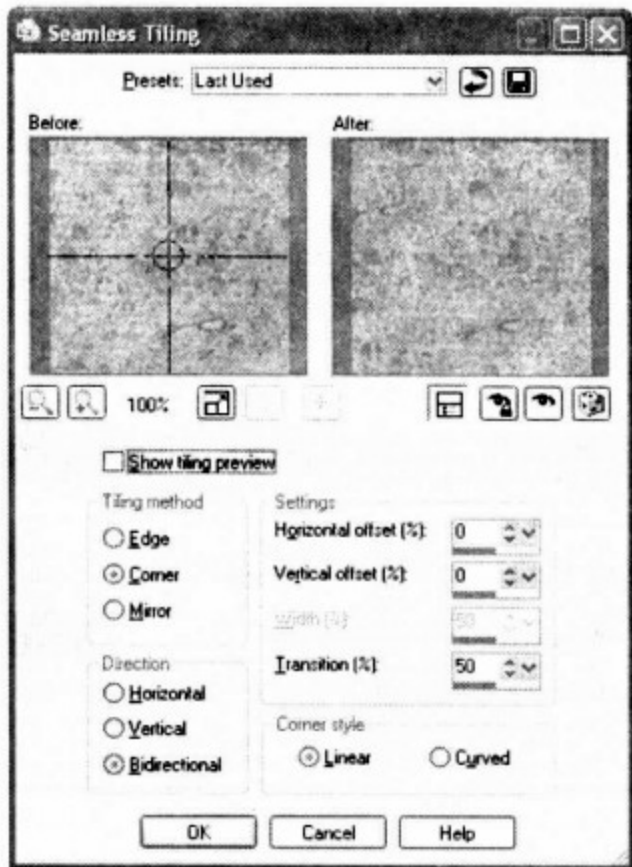
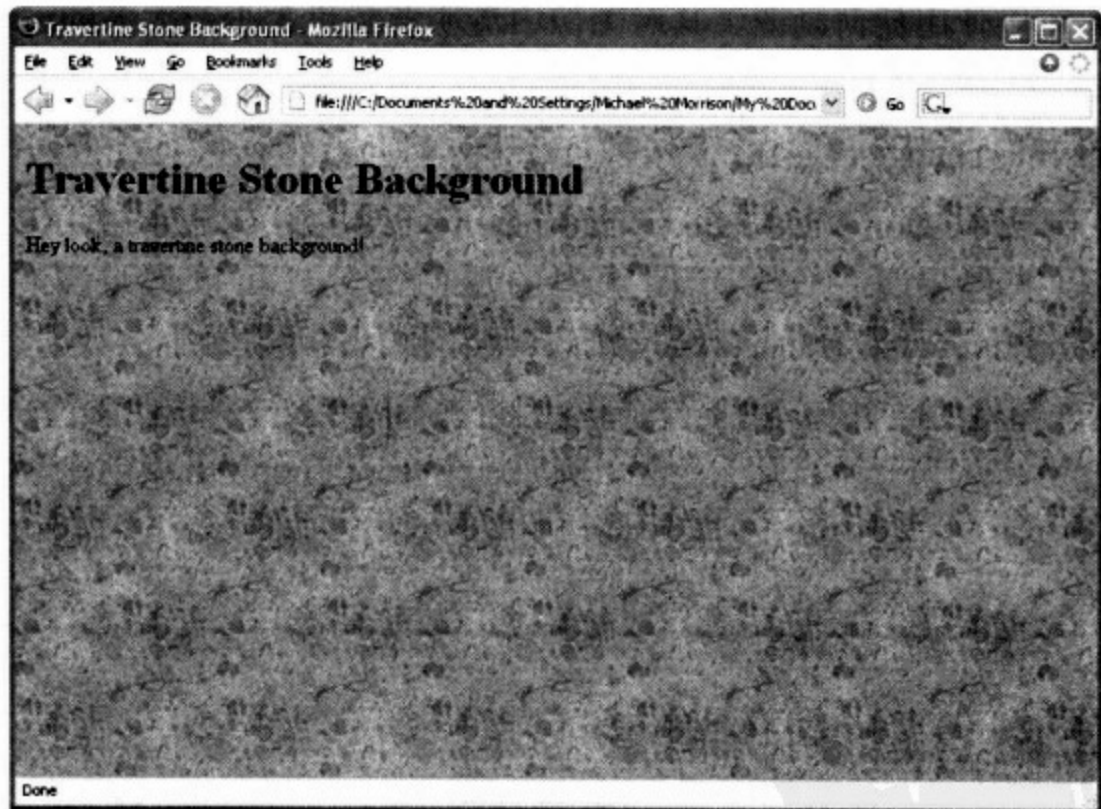


图 9.7

将图 9.6 创建无缝拼贴用作网页的背景图像



其他图像软件，包括 Photoshop（使用 Filter/Other/Offset，并选择 Wrap）、Kai's Power Tools 以及 Macintosh 软件 Mordant 和 Tilery，也有类似的特性。

▼

练习

- 下面是使用 Paint Shop Pro 创建背景拼贴的一些技巧。
- 如果有扫描仪或数码相机，可使用家里或办公室的纹理，如木质书桌的表面、室内

植物的叶子或衣物等。

- 将图像转换为拼贴前，对图像使用 Adjust/Blur/Blur More，这将使图像用作背景时效果更好。通常还要使用 Adjust/Brightness and Contrast，使背景的颜色有细微的变化。
- 选择要转换为拼贴的区域时，尽量选择两边的亮度一致的区域。否则，即使使用 Convert to Seamless Pattern 后，图像也不能无缝地拼贴，。
- 也可以使用 Paint Shop Pro 的画刷样式调板中的纸张纹理特性创建一些几乎自动的纹理。还可以选择 Adjust/Add Noise 或 Adjust/Remove Noise，然后再选择 Adjust/Blur/Blur，以创建纸张纹理。

**提示：**如果不能获得所需的图案，可看看 Internet 上许多提供无版权背景图像的网站，这些图像设计得很专业，且是免费的或价格很低。

**Did you know?**

## 9.6 总结

本章读者学习了如何设置网页的背景颜色和文本颜色，还了解到如何使平铺的背景图像显示在网页的背后、如何使前景图像部分透明从而让背景显示出来，以及如何创建用作背景的无缝平铺图像。

表 9.1 总结了本章介绍的<body>属性及相关的样式属性。

表 9.1	第 9 章介绍的<body>标签的属性与样式
标签/属性/样式	功能
<body>...</ body>	包含 HTML 文档的主体（文本与标签）
style="background-color:color"	设置网页的背景颜色；可以在子元素中覆盖
style="color:color"	设置网页所有文本的颜色；可以在子元素中覆盖
style="background-image:url(imageurl)"	设置网页的背景图像，通常是平铺的图像
style="background-repeat:repeat"	指定如何重复背景图像，可能的取值包括 repeat、repeat-x、repeat-y 和 no-repeat

## 9.7 问与答

**问：**网页浏览器不允许用户选择背景和文本颜色首选项吗？

**答：**允许，网页浏览器允许用户覆盖网页制作者设置的颜色。用户可能将蓝底白字网页显示为白底绿字或自己喜欢的颜色。但是实际上很少有人使用该选项，所以通常都以<body>标签指定的颜色显示。

**问：**我听说有 231 种“浏览器安全色”，网页应使用这些颜色，而不使用其他颜色。这是真的吗？

**答：**真实情况是这样的：有 231 种颜色对使用 256 色（8 位）视频模式计算机的用户显得不那么模糊（其他 25 种颜色用于菜单等元素）。有些网页制作者坚持使用这些颜色。但现在，真彩色或高彩色计算机显示器已经是标准配置，它们能以同样的清晰度显示所有颜色。所以，如果你的图像程序能够显示使用十六进制数表示的颜色值，可在网页中使用十六进制

数指定自定义颜色。也就是说，必须使用“浏览器安全色”的时代已经过去了。

**问：**我的背景图像在图像编辑程序中显示正常，但它在网页中显示时却出现奇怪的白色或有色的缝隙或点。这是为什么？

**答：**有两种可能：如果使用的背景图像是 GIF 文件，它的透明选项可能是打开的，使图像中的某种颜色（或 background-color 样式属性指定的颜色）变白。解决方法是使用图像程序打开文件，并关闭透明选项（在 Paint Shop Pro 中，选择“Image/Palette/Set Palette Transparency”，然后选择 No Transparency），然后重新保存文件。

如果 JPEG 图像、不透明 GIF 或 PNG 图像在网页中出现斑点，可能是因为网页浏览器的仿色功能。这是浏览器使用的一种方法，通过将相邻的颜色点混合在一起，来显示比系统设置更多的颜色。如果是这样，基本上没办法解决，虽然第 7 章提供了缓和这种问题的技巧。

## 9.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 9.8.1 测验

1. 如何将网页的背景设置为黑色，所有文本显示为亮绿色？
2. 如何使图像文件 texture.jpg 重复平铺在使用白色文本的网页的文本与图像后面？
3. 如果图像文件 elephant.jpg 是大象在纯白色背景前的照片。如何使背景透明，从而在网页中只显示大象？

### 9.8.2 答案

1. 将如下代码放在网页开头：

```
<body style="background-color:black; color:lime">
```

如下代码也有相同效果：

```
<body style="background-color:#000000; color:#00FF00">
```

- 2.

```
<body style="background-image:url(texture.jpg);color:white">
```

3. 在 Paint Shop Pro 中打开该图像文件；右击白色区域；选择 Image/Palette/Set Palette Transparency；如有必要接受减少颜色数；选择将背景色变成透明的。对所有没有变成透明的非白色点进行修饰，然后使用 File/Save As 将图像保存为 GIF 89a 格式。也可以裁剪出大象，然后将它粘贴到一个背景为透明的新图像中，再将其保存为 PNG 图像。



## 9.9 练习

1. 使用不寻常或令人意想不到的图像来创建背景。如果浏览自己收藏的数码照片，可能发现某张照片中的一片草地或其他材质可转变为很酷的背景图像。

2. 如果有用于基于 Web 的商品目录的照片，花些时间在照片周围的小心地绘制一种透明颜色（图像编辑器中的魔棒工具可能能够帮你自动完成这项任务）。也可以在 **Paint Shop Pro** 中使用 **Effect/3D Effect/Drop Shadow** 功能，在每件商品的后面或下面添加轻微的阴影，使其从背景中突显出来。



## 第 10 章

# 图像链接和图像映射

如果没有跳过太多内容且阅读了第 8 章，读者应该知道如何创建指向另一个文档的图像链接。如果你现在不太记得如何创建，提示一下，图像链接类似于这样：`<a href="gohere.htm"></a>`。

也可以将图像分为几个部分，它们分别指向不同的文档。这称为图像映射，任何图像都可以作为图像映射。医疗信息网站可能显示人体图像，每个人体部分指向提供相关建议的页面。世界地图可能允许用户单击不同的国家，以查看地区性信息。许多人使用图像映射来创建“导航栏”，将网站中每个页面的图标集成为一幅图像映射。

现代浏览器，如 Mozilla Firefox 和 Microsoft Internet Explorer，都支持两种实现图像映射的方法。但现在，所有图像映射都应该使用最新的方法实现，这就是客户端图像映射。没有理由再使用老式的服务器端图像映射，因为网页浏览器对客户端图像映射的支持已经有很多年了。

### 10.1 为什么不一定需要图像映射

关于图像映射，首先要说明的一点是，除非在很特殊的情况下，否则不需要使用图像映射。通常使用多幅图像，将它们排列在一起，每幅图像作为一个单独链接指向不同的页面，会比较简单，也更高效。

举个例子，假设有一幅图像可以分为不同的矩形区域。冰上曲棍球场便是这样的一个例子，用于说明国家曲棍球联盟（NHL）为 2005 年赛季修改的规则。图 10.1 显示了这样一幅图像。在网页上使用这个球场图像的一种方法是，使用图像映射将图像的特定区域链接到更详细的信息。

然而，这个例子更好的解决方法是使用图像编辑器，如 PhotoShop 或 Paint Shop Pro 将图像沿垂直方向剪切为几部分，然后将每个部分作为单独的网页图像。图像可以紧密排列在一起，看起来就像一幅完整的图像。这样做的优点是，页面可以与旧网页浏览器兼容，且代码更简单，因为不必使用数学方式描述图像的区域。图 10.2 显示了如何剪切上面的冰场图像，使每个链接区域成为单独的图像；这个图片实际上显示了球场的左半部分。

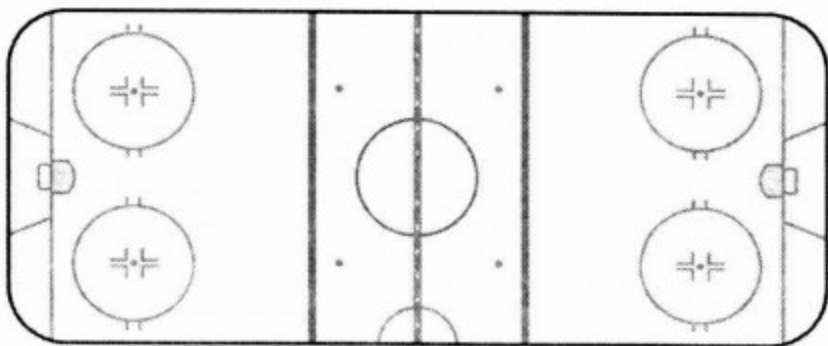


图 10.1  
冰上曲棍球场是一个使用图像映射的很好例子，它显示了如何将图像的区域映射到不同的网页

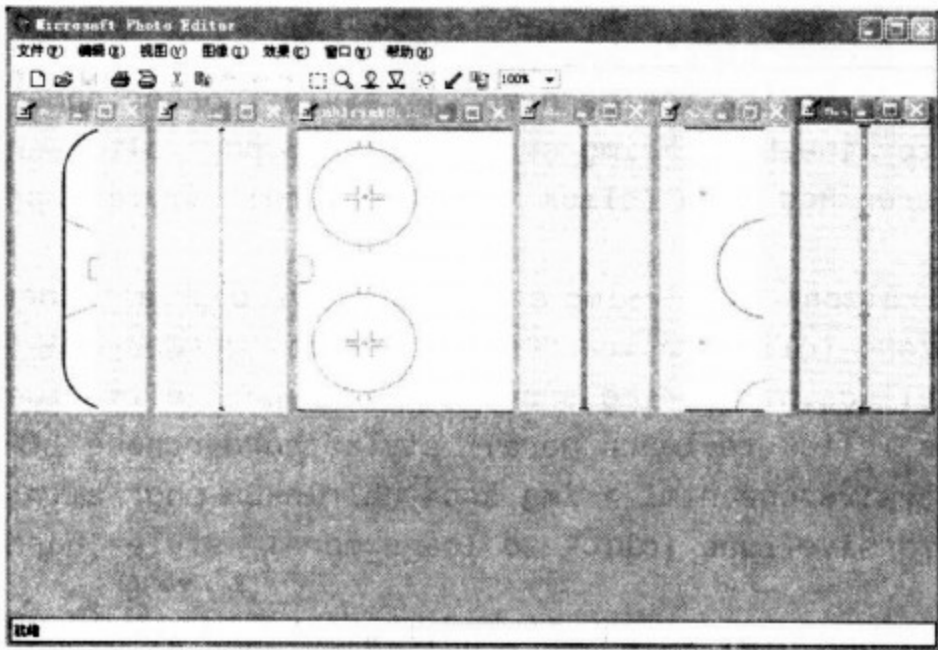


图 10.2  
为避免使用图像映射，将图 10.1 所示图像的左半部分剪切为六幅图像使用矩形选择工具剪切并粘贴)

**注意:**图 10.2 使用图像编辑器是 Microsoft 图片编辑器,它随 Microsoft Office 一起发布。这是一个很小的图片编辑器,但对于剪切图像来说——如在这个例子中——已经足够了。

By the  
Way

程序清单 10.1 列出了用于创建使用如图 10.2 所示的分块球场图像的网页的 HTML 代码。图 10.2 实际上只显示了球场左边的图像——球场右边还有一个类似的图像。

程序清单 10.1 使用分块图像模拟图像映射

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>NHL Rule Changes</title>
  </head>

  <body>
    <h1>NHL Rule Changes</h1>
    <p>
      Following is a graphical representation of some of the new NHL rule
      changes, and how they will affect the ice in 2005 and beyond.
    </p>
    <p style="text-align:center">
```

```

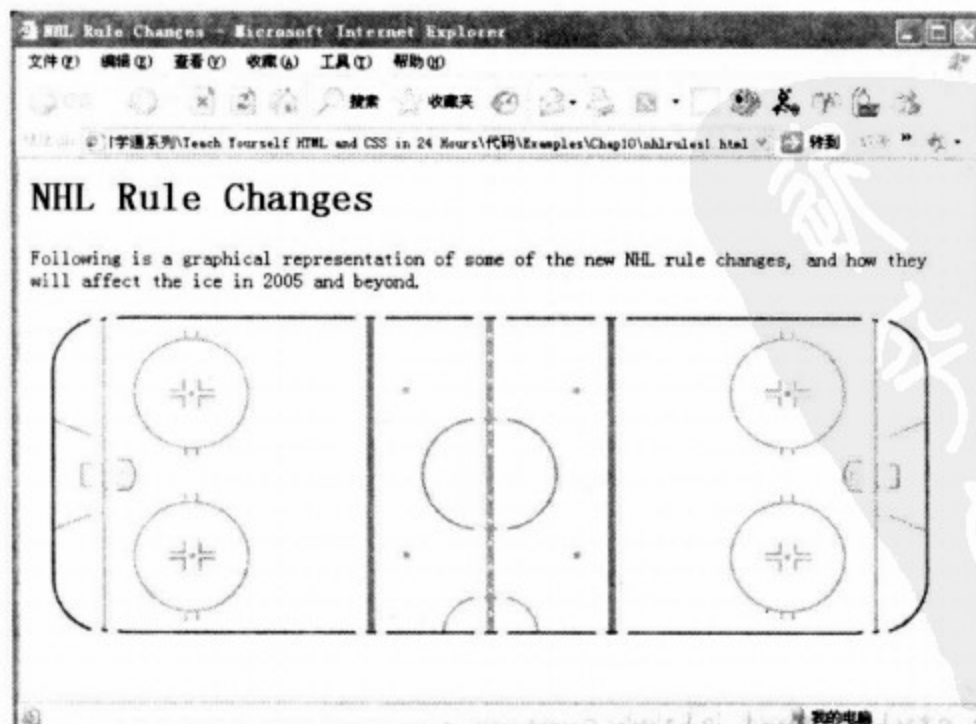
<a href="nhltrapzone.html"></a>
<a href="nhlgoallines.html"></a>
<a href="nhloffensivezone.html"></a>
<a href="nhlbluelines.html"></a>
<a href="nhlneutralzone.html"></a>
<a href="nhlcenterline.html"></a>
<a href="nhlneutralzone.html"></a>
<a href="nhlbluelines.html"></a>
<a href="nhloffensivezone.html"></a>
<a href="nhlgoallines.html"></a>
<a href="nhltrapzone.html"></a>
</p>
</body>
</html>

```

图 10.3 显示了这段代码的结果，它看上去可能和你期望的不太一样。

图 10.3

虽然图像块排列得当，但图像间的小空隙破坏了模拟图像映射的效果





这段代码的问题在于，<a>标签之间包含分行，这意味着图像之间存在分行。这样的分行结果是，页面上的图像之间出现空隙。解决方法是修改代码，使分行位于每个标签内，而不是相邻的标签之间，程序清单 10.2 显示了对这段代码进行修改，解决了分行问题的版本。

程序清单 10.2 重新安排代码中的分行，解决了页面上的空隙问题

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

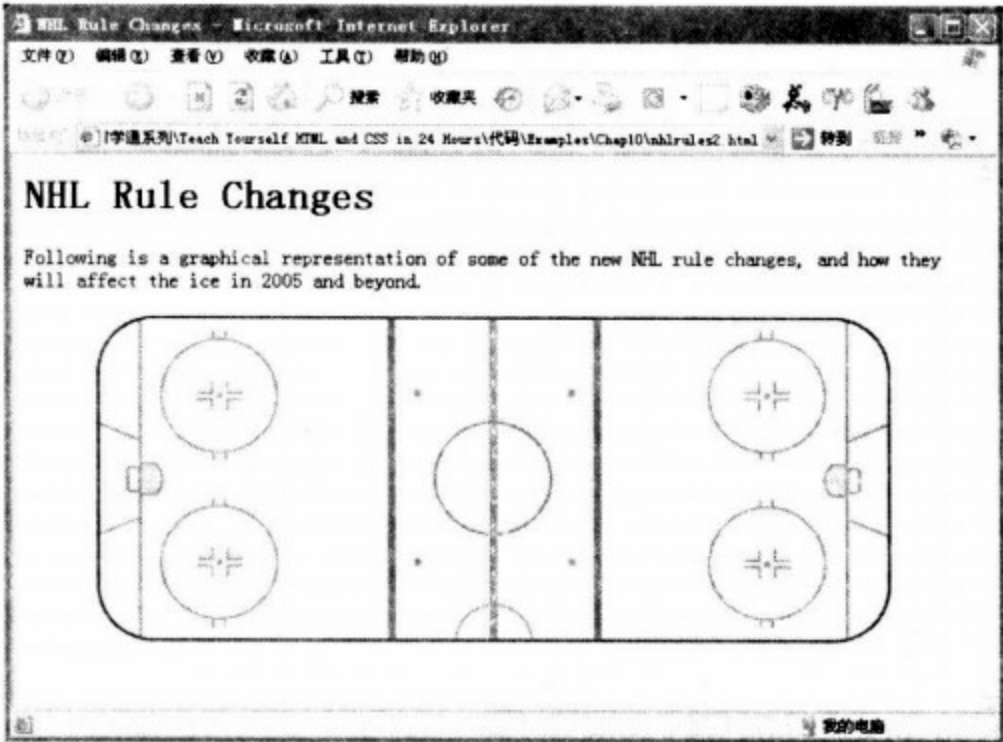
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>NHL Rule Changes</title>
  </head>

  <body>
    <h1>NHL Rule Changes</h1>
    <p>
      Following is a graphical representation of some of the new NHL rule
      changes, and how they will affect the ice in 2005 and beyond.
    </p>
    <p style="text-align:center">
      <a href="nhltrapzone.html"></a><a
      href="nhlgoallines.html"></a><a
      href="nhloffensivezone.html"></a><a
      href="nhlbluelines.html"></a><a
      href="nhlneutralzone.html"></a><a
      href="nhlcenterline.html"></a><a
      href="nhlneutralzone.html"></a><a
      href="nhlbluelines.html"></a><a
      href="nhloffensivezone.html"></a><a
      href="nhlgoallines.html"></a><a
      href="nhltrapzone.html"></a>
    </p>
```

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

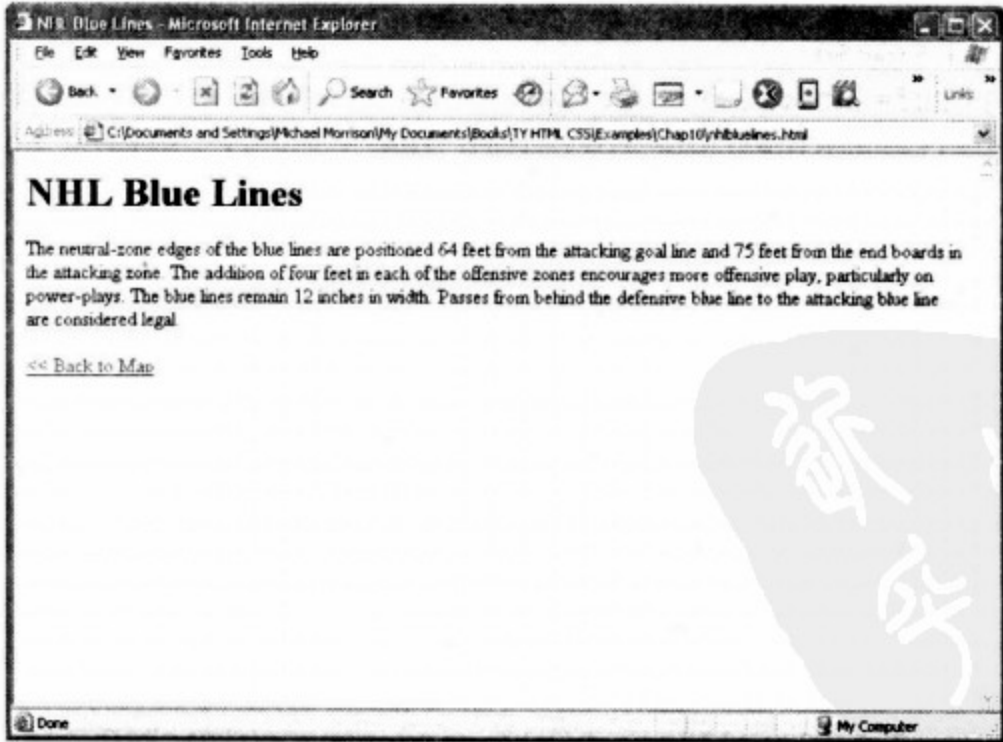
注意，这次没有在<a>标签之间或<img />标签之间加入任何空格或分行。再强调一次，这些标签之间的空格或分行将使得页面出现小空隙，这样将多幅图像组成大图像的效果将遭到破坏。图 10.4 显示了代码修改后的结果，这一次冰场的效果和期望的一样。

图 10.4  
可以用<img />标签和<a href>链接来创建这个网页，不一定要使用图像映射



将鼠标指向冰场，将显示每个图像块的 alt 文本，这可以提供与图像相关的提示。单击冰场的任何部分，将显示一个网页，提供影响冰场特定部分的 NHL 规则变化的详情。图 10.5 显示了影响表示曲棍球犯规区域的蓝线的规则变化。

图 10.5  
冰场图像的每个区域都链接到一个详细描述该区域的网页



证明可以不用图像映射而模拟图像映射后，提出了一个问题：什么时候使用图像映射？仅当要链接的图像部分很多或排列不规则，以至于难以将图像剪切为更小的图像时才使用。这种情况时有发生，因此，真的需要使用图像映射时就应知道如何创建。本章接下来的

部分将介绍如何创建图像映射。

## 10.2 映射图像中的区域

要创建图像映射，首先需要知道图像内要变为可单击链接的每个区域的数字像素坐标。这些可单击链接又称为区域（area）。确定区域坐标的一种简单方法是，在图像编辑器中打开图像，使用矩形选择工具在图像中选中一块矩形区域，然后在屏幕的底部就可以查看这块区域的坐标（见图 10.6）。按下鼠标时，屏幕底部的坐标指出了该矩形左上角和右下角的坐标。没有按下鼠标时，仅显示鼠标所处的 X 和 Y 坐标。



图 10.6  
可使用图像编辑器在图像中选择区域并确定区域的坐标

可以使用图 10.6 所示的整幅图像作为图像映射，链接到几幅图像，形成一个滑板公园的虚拟旅行。要实现这一点，首先需要决定图像的哪个区域要链接到哪个目标图像。区域可以是矩形、圆形和不规则多边形。图 10.7 是一个将图像划分为这些形状的例子。

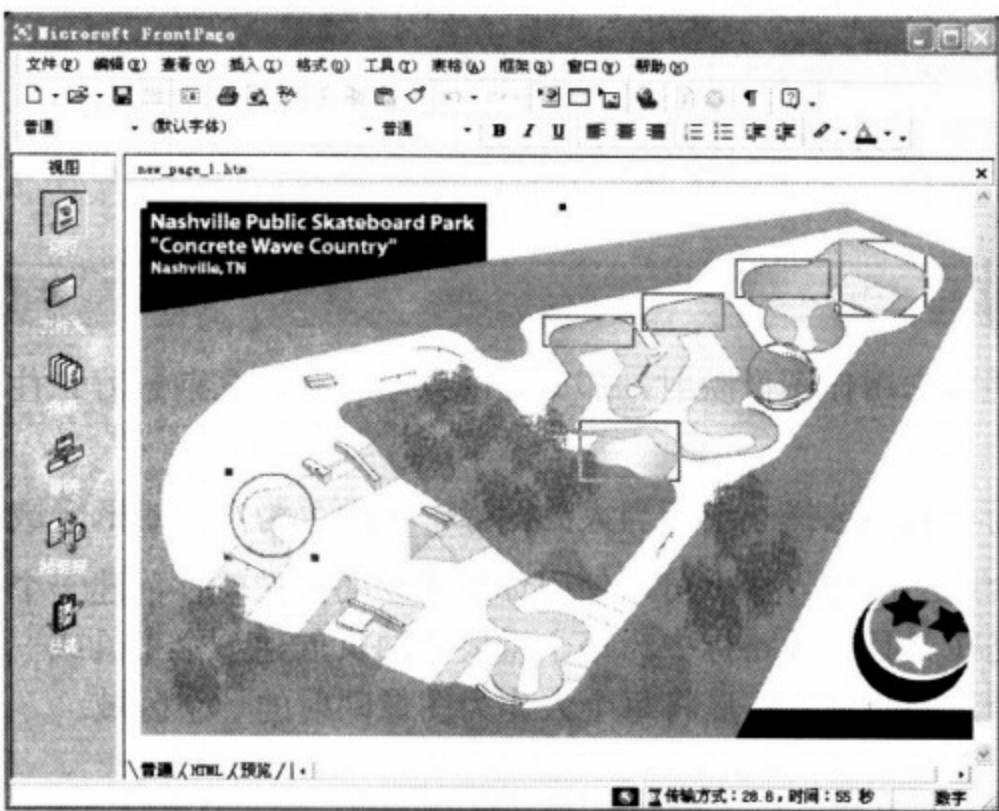


图 10.7  
Microsoft FrontPage 让用户能够使用鼠标在图像映射上绘制可单击的热点

**Did you know?**

**提示：**图形网页编辑器，如 Microsoft FrontPage、Macromedia Dreamweaver 和 Adobe GoLive，让用户能够以交互方式在图像上绘制热点（hotspot，见图 10.7），然后为用户生成必要的 HTML 代码。这是图形网页编辑器方便的地方。

还有些软件允许用户使用鼠标选择一个矩形，并自动将图像映射的坐标保存到一个文件中，用户可以将坐标剪切并粘贴到 HTML 代码中，但这有点麻烦。如果没有 FrontPage 或其他好的图形网页编辑器，可在通用图像软件或图像编辑器中获得像素坐标。

**By the Way**

**注意：**FrontPage 将自动生成图像映射的代码，但本课程的目标是教你学会如何从零开始为图像映射编写代码。因此我只是用 FrontPage 作为例子，说明如何获得图像中每个区域的坐标。也可以使用其他图像工具。

要创建图像映射，只要记下图 10.7 中每个矩形区域左上角和右下角的坐标，然后定位并记录三个圆的圆心和半径，以及两个不规则区域的每个顶点的坐标。如果使用的是自己的图像，现在只要记下这些坐标。下面将看到如何将这些坐标放到 HTML 文件中。

滑板公园图像映射区域的坐标如下。

- Spike Pipe（区域 1）：从 536 24 到 610 89 的矩形。
- Coach's Corner（区域 2）：从 461 43 到 535 78 的矩形。
- Barnstormer（区域 3）：从 342 170 到 422 218 的矩形。
- Mitchell's Meadow（区域 4）：从 392 69 到 460 114 的矩形。
- MIA Bowl（区域 5）：从 311 87 到 391 118 的矩形。
- Bubba Bowl（区域 6）：圆心位于 510 94、半径为 30 的圆形。
- Flume Zoom（区域 7）：圆心位于 499 135、半径为 25 的圆形。
- Orange Peel（区域 8）：圆心位于 98 243、半径为 28 的圆形。
- Horsey Hip（区域 9）：不规则多边形，有如下六个顶点：(323 134)、(309 154)、(343 166)、(358 139)、(345 119) 和 (311 116)。
- Devan's Divide（区域 10）：不规则多边形，有如下六个顶点：(227 229)、(254 243)、(273 267)、(247 281)、(206 278) 和 (203 244)。



如果你有自己的图像并将它用作图像映射，那么在继续学习本章之前最好记住如何创建图像映射。

- 对于初学者，最简单的是选择一个相当大的图像，大致分为几个矩形区域。
- 如果你手头没有合适的图像，那么可以使用你最喜欢的图像软件创建一个。一个简单而有用的方法是在工具栏或路标中为每个重要的网页创建一个工具或一个单词。或者使用包含多个人的照片，然后将每个人作为图像映射的一个区域。



## 10.3 创建客户端图像映射

记下坐标后，就可以开始创建 HTML 图像映射。图像映射都以下列 HTML 代码开始：

```
<map id="skatepark">
```

**注意：**在 XHTML 之前，<map> 标签中的 id 和 name 属性的作用是一样的，因此也可用下列代码：

```
<map name="skatepark">
```

然而，XHTML 标准要求使用 id 而不是 name，这就是我在本章（以及本书的其余部分）坚持使用 id 的原因。

**By the  
Way**

要记住，<map> 标签的 id 可用任意名称，但最好使用具有说明性的名称。如果 skateparkmap 不能描述你使用的图像，可修改它。

接下来是图像的每个区域各使用一个<area /> 标签。下面是滑板公园图像映射中使用的一个<area /> 标签：

```
<area shape="rect" coords="536, 24, 610, 89" href="skatepark_spikepipe.jpg"
alt="Spike Pipe" />
```

<area /> 标签有四个属性，可用于描述图像映射中的每个区域。

- shape 指明该区域为矩形（shape="rect"）、圆形（shape="circle"）还是不规则多边形（shape="poly"）。
- coords 给出区域的像素坐标。对于矩形，给出左上角的 x,y 坐标以及右下角的 x,y 坐标；对于圆形，给出圆心的 x,y 坐标和半径，以像素为单位；对于多边形，按连接顺序列出所有顶点的 x,y 坐标。
- href 指定区域链接的页面。可以像在普通<a href> 链接标签中那样使用任何地址或文件名。
- alt 让你能够提供与该区域形状相关联的一小段文字。当用户将鼠标指向该区域时，大多数浏览器（除 Firefox）将显示一个小窗口。这些文字为不理解图像映射的用户提供提示，虽然小却很重要。Firefox 在提供提示时，使用的是 title 属性而不是 alt 属性，这也就是为什么应尽量为图像同时提供两个属性。

图像映射中的每个可单击区域都必须描述为一个区域，也就是说典型的图像映射有一系列区域组成。编写完<area /> 标签后，图像映射的定义就完成了，因此最后加上</map> 结束标签。

创建图像映射的最后一步是，使用普通的<img /> 标签将真正的地图图像放在网页中。然而，有一个额外的 usemap 属性，如下所示：

```

```

指定 usemap 属性的值时, 使用<map>标签中 id 属性指定的名称 (别忘了符号#)。在程序清单中, 还使用了 style 属性, 使图像映射的边框不显示出来, 你自己的图像映射可以保留也可以不保留。

### By the Way

**注意:** 也可以将映射定义放在一个独立的文件中, 然后在 usemap 属性中包含该文件名, 如下所示:

```

```

例如, 如果网站的每个页面都使用图像映射, 可以将<map>标签和<area>标签放在一个页面中, 而不必在每个使用图像映射的网页都重复使用这些标签。

程序清单 10.3 列出了滑板公园图像映射网页的完整代码, 包括图像映射及其所有区域。

#### 程序清单 10.3 <map>标签和<area>标签定义了图像映射的区域

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Nashville Public Skatepark (Concrete Wave Country)</title>
  </head>

  <body>
    <h1>Nashville Public Skatepark (Concrete Wave Country)</h1>
    <p>
      Below is a map of Nashville's public skatepark, also known as
      Concrete Wave Country. Drag the mouse over the map and click to
      learn more about each area and obstacle.
    </p>
    <p style="text-align:center">
      <map id="skateparkmap">
        <area shape="rect" coords="536, 24, 610, 89"
          href="skatepark_spikepipe.jpg" alt="Spike Pipe" />
        <area shape="rect" coords="461, 43, 535, 78"
          href="skatepark_coachscorner.jpg" alt="Coach's Corner" />
        <area shape="rect" coords="342, 170, 422, 218"
          href="skatepark_barnstormer.jpg" alt="Barnstormer" />
        <area shape="rect" coords="392, 69, 460, 114"
          href="skatepark_mitchellsmeadow.jpg" alt="Mitchell's Meadow" />
        <area shape="rect" coords="311, 87, 391, 118"
          href="skatepark_miabowl.jpg" alt="MIA Bowl" />
        <area shape="circle" coords="510, 94, 30"
          href="skatepark_bubbabowl.jpg" alt="Bubba Bowl" />
      </map>
    </p>
  </body>
</html>
```

```
<area shape="circle" coords="499, 135, 25"
href="skatepark_flumezoom.jpg" alt="Flume Zoom" />
<area shape="circle" coords="98, 243, 28"
href="skatepark_orangepeel.jpg" alt="Orange Peel" />
<area shape="poly" coords="323, 134, 309, 154, 343, 166, 358, 139,
345, 119, 311, 116" href="skatepark_horsehip.jpg"
alt="Horsey Hip" />
<area shape="poly" coords="227, 229, 254, 243, 273, 267, 247, 281,
206, 278, 203, 244" href="skatepark_devansdivide.jpg"
alt="Devan' s Divide" />
</map>

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

**注意：**如果你注重细节的话，可能注意到这个网页是 XHTML 1.0 文档，而不像本书其他大部分例子那样是 XHTML 1.1 文档（看看代码的前几行）。这样做的原因是，有些浏览器（如 Internet Explorer）不支持 XHTML 1.1 中关于如何使用图像映射的小修改。这里说的是 usemap 属性，在 XHTML 1.1 中，映射名称前面不要求符号#。实际上，这在 XHTML 1.1 中是不允许的，这也就是我为什么将该网页设计成 XHTML 1.0 文档。符号#在 XHTML 1.0 中是允许的，因此，为满足当前网页浏览器的需要，同时提供有效的网页，我在这个例子中选择了 XHTML 1.0。

By the  
Way

图 10.8 显示了图像映射。注意，当鼠标在窗口底部的区域移动时，Microsoft Internet Explorer 将显示该区域的链接地址，就像普通链接一样。另外，将鼠标停放在区域上时，区域的 alt 文字将显示在图像映射上。



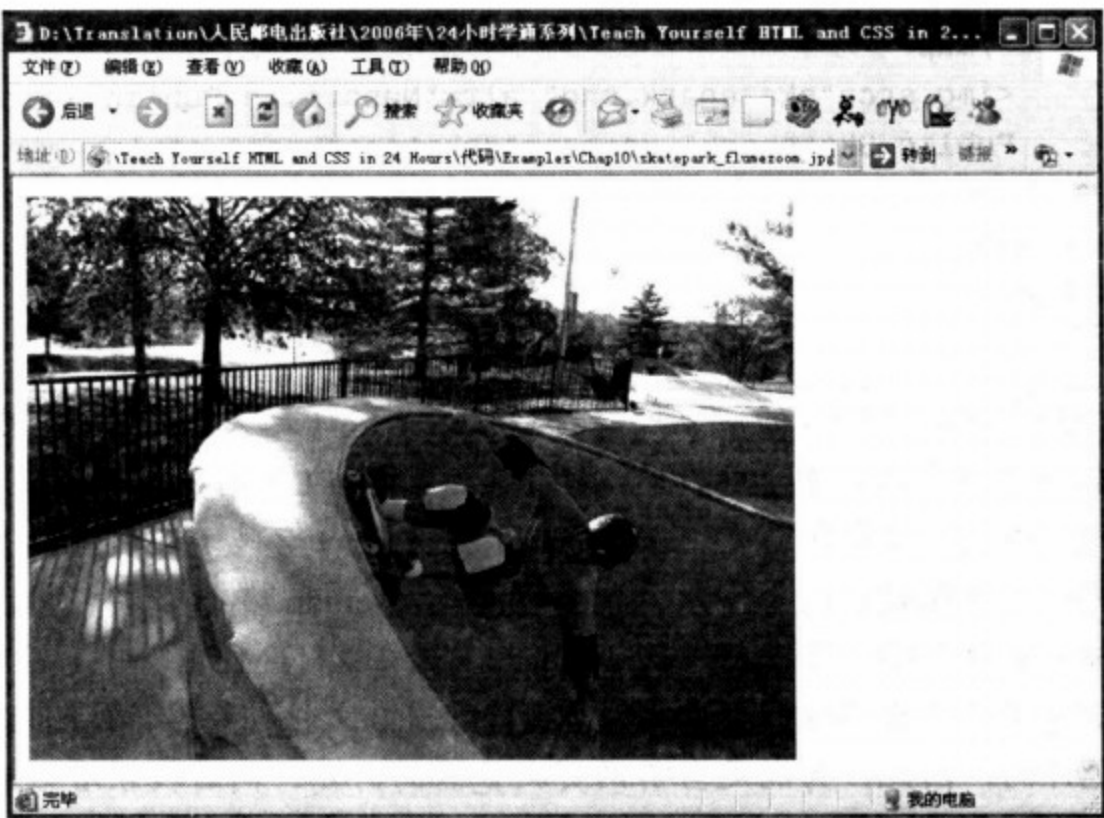
图 10.8  
程序清单 10.3 定义的图像映射在网页中的显示结果

Did you Know?

**提示：**你可能希望在图像映射底部包含文本链接，指向映射链接的页面。这样，使用老式网页浏览器的用户——或者不想等到图像加载完毕的用户——也能访问这些页面。

如果单击图 10.8 所示的 alt 文字，将在浏览器中打开图像 skatepark\_flumezoom.jpg。实际上，单击 Flume Zoom 后，将显示我朋友 Paul Sexton 在 Flume Zoom 滑行的图像，如图 10.9 所示。

图 10.9  
滑板公园图像映射中的每个区域分别指向一个滑板障碍



10.4 总结

本章解释了如何创建图像映射——指向多个地方的链接，根据在图像上单击的位置不同而链接到不同的地方——以及在有些情况下为什么以及如何避免使用它们。读者学习了如何在一幅图像中定义矩形、圆形以及不规则多边形的链接区域；还了解了图像映射虽然有专门的用途，但在有些情况下非常有用。使用得当的话，图像映射将是网页用户界面的重要组成部分。

表 10.1 总结了本章介绍的标签和属性。

表 10.1                      第 10 章介绍的标签和属性

标签/属性	功能
<img />	在文档中插入一幅图像
属性	
src=" 图像 URL "	图像的 URL
alt=" 描述 "	映射图像的文本描述，目的是在图像不能显示时替代图像
title=" 描述 "	区域的文本描述，可能显示在图像旁边的提示
usemap=" 名称 "	为客户端图像映射指定的图像映射名，用于与<map>及<area />



续表

标签/属性	功能
<map>...</ map>	客户端图像映射，由<img usemap="..." />引用。包含一个或多个<area />标签
<area />	在用户端图像映射中定义可单击的链接
属性	
shape=" 值 "	可单击区域的形状，这个属性有效的取值有 rect、poly 和 circle
cords=" 值 "	图像中可单击区域的坐标；其含义和设置随区域类型而异
href=" 链接 URL "	单击区域时应加载的 URL
alt=" 描述 "	区域的文本描述，目的是当图像不能显示时替代图像显示

## 10.5 问与答

问：我没有 Paint Shop Pro，且我的图像软件没有告诉我 x,y 坐标。我如何找出图像映射的坐标呢？

答：有一个诀窍，那就是可以使用网页浏览器来解决这个问题。使用 ismap 属性和<a> 标签将图像放在网页中，如下所示：

```
<a href="nowhere"></a>
```

通过网页浏览器查看该网页时，将鼠标指向图像，将在窗口底部的状态栏看到坐标。这是因为 ismap 属性用于老式服务器端图像映射，它将鼠标坐标传回 Web 服务器，以确定图像映射的哪部分被单击。

问：如果图像映射重叠了，将发生什么？

答：图像映射可以重叠，但要记住，确定采用哪个链接时，一个区域会比其他区域有更高的优先权。优先权取决于哪个区域最先在图像映射中列出。例如，如果图像映射中的第一个区域与第二个区域重叠，第一个区域有更高的优先权，也就是说如果单击这两个区域的重叠部分，将链接到第一个区域的链接。如果希望图像映射中有一个区域不链接到任何地方，可以使用重叠技巧故意使其不链接到任何地方：只要将这个“死”区域置于其他区域前，并将其 href 属性设置为""。

## 10.6 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 10.6.1 测验

1. 你的网页有一个名为 quarters.gif 的 200×200 像素图像。用户单击该图像左上角的 1/4 时，进入一个名为 topleft.html 的网页；单击右上角的 1/4 时，进入 topright.html；单击左下角的 1/4 时进入 bottom-left.html，而单击右下角的 1/4 时进入 bottomright.html。编写 HTML 代码，通过图像映射实现这些效果。

2. 如何不通过图像映射实现问题 1 的效果?

## 10.6.2 答案

1.

```
<map id="quartersmap">
  <area shape="rect"coords="0,0,99,99"href="topleft.html" alt="" />
  <area shape="rect"coords="100,0,199,99"href="topright.html" alt="" />
  <area shape="rect"coords="0,100,99,199"href="bottomleft.html" alt="" />
  <area shape="rect"coords="100,100,199,199"href="bottomright.html"
  alt="" />
</map>

```

2. 使用图像程序将图像剪切成四部分，并将它们分别以文件名 topleft.gif、topright.gif、bottomleft.gif 和 bottomright.gif 保存，然后编写如下代码：

```
<a href="topleft.html"></a>
<a href="topright.html"></a> <br />
<a href="bottomleft.html"></a>
<a href="bottomright.html"></a>
```

像上述 HTML 代码那样，在标签内分行，以避免图像间出现空隙。另外，width 和 height 属性是可选的，指定这两个属性旨在让网页浏览器能够更快地显示网页。

## 10.7 练习

1. 如果有网页包含简短的链接列表，看看能否将它们制作成有趣的图像映射。

2. 图像映射不仅是一些看上去重复的图标或按钮，通常还可以做得更有吸引力。可否给出与你的网站相关的可视化比喻，使得浏览你的网页更容易，且可能更有趣？以这种方式思考问题，可为将在第三部分处理的问题做好准备。

## 第三部分

### 充满创意的网页设计

第 11 章 使用表格组织和排列网页

第 12 章 使用 CSS 格式化网页

第 13 章 深入探讨样式表格式化

第 14 章 将样式表用于页面布局

第 15 章 创建打印友好的网页

第 16 章 使用框架创建多页面布局







# 第 11 章

## 使用表格组织和排列网页

要设计有创意的网页，最强大的工具之一是表格，它让你能够将文本和图像组织成多列和多行。本章将介绍如何创建 HTML 表格，控制网页中表格数据的间距、布局和外观。表格不仅可用于将信息组织为行和列，还可用于排列网页上的文本和图像。我还将解释为什么使用表格来组织网页并不总是最好的解决办法，并提供其他的方法。表格是一种将文本或图像排列为竖向列和横向行的方法。

### 练习

阅读本章课程时，考虑将文本组织为表格对改进网页布局有何帮助。请记住以下几点：

- 表格最明显的用途是用于组织表格信息，如由名称和数字组成的多列表；
- 如果文本和图像之间的关系复杂到使用<img align="left" />和<img align="right" />无法实现，可使用表格来处理；
- 表格可用于在文本或多个图像周围绘制边框；
- 需要将文本排成多列时，可使用表格。

如果网页满足上述条件之一，可按本章介绍的方法添加表格。本章最后的“练习”小节提供了这方面的具体建议。



### 11.1 创建简单表格

表格由多行信息组成，每行包含单元格。要创建表格，必须以<table>标签开始；当然，表格以</table>标签结束。如果要显示表格边框，使用 border 属性来指定边框的宽度，单位为像素。边框宽度为 0 或 none（或省略 border 属性）将隐藏边框，这在使用表格作为网页布局工具时很方便。

**Did you know?**

**提示：**有些样式属性可用于更详细地控制表格边框。例如，可设置边框宽度（border-width）、样式（border-style）和颜色（border-color）。这些属性是管用的，但必须将其应用于每个表格元素，因此比较麻烦。

指定<table>标签后，接下来需要使用<tr>标签。<tr>标签创建表格的一行，包含一个或多个单元格，最后以</tr>标签结束。要创建单元格，可使用<td>标签（<td>表示表格数据——table data）。表格信息放在标签<td>和</td>之间。单元格（cell）是一个矩形区域，可容纳任何文本、图像和 HTML 标签。表格中的每行至少包含一个单元格。行中的单元格组成列。

创建表格时，涉及一个基本标签：<th>，它与<td>标签类似，但表示单元格是表头的一部分。大多数网页浏览器将<th>单元格中的文本居中对齐并显示为粗体。

表格中的行可以包含任意数量的单元格，但每行应包含相等数量的列。程序清单 11.1 中的 HTML 代码使用前面提到的四个表格标签，创建了一个简单表格。图 11.1 显示了该页面在网页浏览器中的显示结果。

---

**程序清单 11.1 使用<table>、<tr>、<td>和<th>标签创建表格**


---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Hockey Stats</title>
  </head>

  <body>
    <table>
      <tr>
        <th>Season</th>
        <th>GP</th>
        <th>G</th>
        <th>A</th>
        <th>P</th>
        <th>PIM</th>
      </tr>
      <tr>
        <td>Summer 2005</td>
        <td>8</td>
        <td>4</td>
        <td>4</td>
        <td>8</td>
        <td>0</td>
      </tr>
      <tr>
        <td>Winter 2004</td>
        <td>24</td>
```

```
<td>14</td>
<td>14</td>
<td>28</td>
<td>2</td>
</tr>
<tr>
<td>Summer 2004</td>
<td>18</td>
<td>9</td>
<td>9</td>
<td>18</td>
<td>2</td>
</tr>
<tr>
<td>Spring 2004</td>
<td>19</td>
<td>7</td>
<td>17</td>
<td>24</td>
<td>0</td>
</tr>
</table>
</body>
</html>
```

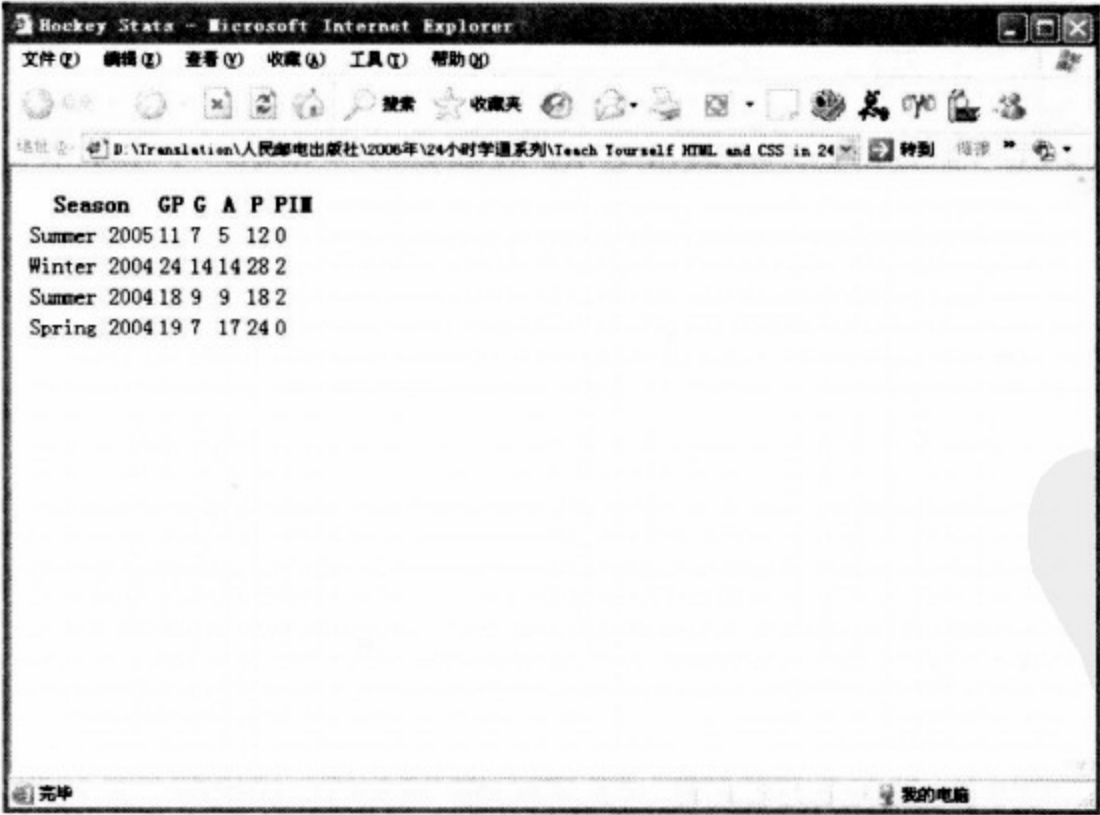


图 11.1  
程序清单 11.1 中的 HTML 代码创建一个五行六列的表格

**提示：**HTML 忽略单词与标签之间的空格。然而，如果像程序清单 11.1 那样使用空格来缩进<tr><td>标签，HTML 表格将更易于阅读（也不容易出现浪费时间的错误）。

*Did you know?*

这个例子中的表格包含曲棍球统计数据，这些数据很适合排列在行和列中。表头分别代表参加的比赛（Games Played, GP）、射门（Goals, G）、助攻（Assists, A）、得分（Points, P）和罚点球数（Penalties In Minutes, PIM）。在表格单元格中可以放入任何 HTML 元素。然而，一个单元格中的标签不会影响其他单元格，表格外的标签也不会影响表格中的信息。例如，请看下面的表格：

```
<div style="font-weight:bold">
  <table>
    <tr>
      <td style="font-style:italic">hello</td>
      <td>there</td>
    </tr>
  </table>
</div>
```

第 5 章介绍过，<div>标签用于包含页面中的信息块，因此可以将整块信息对齐并应用样式。在这个例子中，<div>标签用于包含表格，以演示表格不受外部标签的影响。单词 *there* 既不是粗体也不是斜体，因为表格外的 font-weight:bold 和前一个单元格中的 style:italic 样式都不会对它产生影响。然而，这个例子中的单词 *hello* 是斜体的。

要使单词 *hello* 和 *there* 都变成粗体，需要将表格的代码改成：

```
<table style="font-weight:bold">
  <tr>
    <td style="font-style:italic">hello</td>
    <td>there</td>
  </tr>
</table>
```

在这个例子中，两个单词都是粗体，且单词 *hello* 还是斜体。当然，可以不将样式应用于整个表格。如果需要的话，font-weight:bold 也可应用于单个单元格。

## 11.2 控制表格的大小

通常，表格及其单元格的大小会自动扩展，以适应其中包含的数据。然而，可以在 <table> 标签中指定 width 和 height 样式来控制整个表格的大小；也可以在每个 <td> 标签中指定 width 和 height 样式来控制每个单元格的大小。width 和 height 样式可用像素或百分比来指定。例如，下面的代码创建了一个宽 500 像素、高 400 像素的表格：

```
<table style="width:500px; height:400px">
```

**By the Way**

**注意：**在 XHTML 中，不提倡使用属性 width 和 height。这两个属性在网页浏览器中仍可用，但应尽量使用样式属性 width 和 height，这是 XHTML 中的正确方法。

要使第一个单元格占据整个表格宽度的 20%，第二个单元格占据整个表格宽度的 80%，



应使用如下所示的代码：

```
<table style="width:100%">
  <tr>
    <td style="width:20%">skinny cell</td>
    <td style="width:80%">fat cell</td>
  </tr>
</table>
```

注意，表格的大小指定为 100%，这将使表格与浏览器窗口等宽。使用百分比而不是像素时，表格将自动调整大小以适应浏览器窗口的大小，同时保持你所追求的美学平衡。在这个例子中，表格中的两个单元格将分别自动调整为表格宽度的 20%和 80%。

### 11.3 表格内部的对齐和跨度

默认情况下，放在表格单元格中的内容将水平左对齐，并垂直居中。可用样式属性 `text-align` 和 `vertical-align` 来控制表格单元格中的内容在水平方向和垂直方向的对齐方式。

这些属性可用于任何 `<tr>`、`<td>` 或 `<th>` 标签。应用于 `<tr>` 标签的对齐属性将应用于该行所有的单元格。如果表格很大，应将这些属性应用于 `<tr>` 而不是每个 `<td>` 或 `<th>` 标签，这样可以节省很多时间和精力。程序清单 11.2 中的 HTML 代码使用 `style="vertical-align:top"` 使文本出现在单元格的顶部。图 11.2 显示了结果。

程序清单 11.2 表格内的对齐、单元格间距、边框和背景颜色

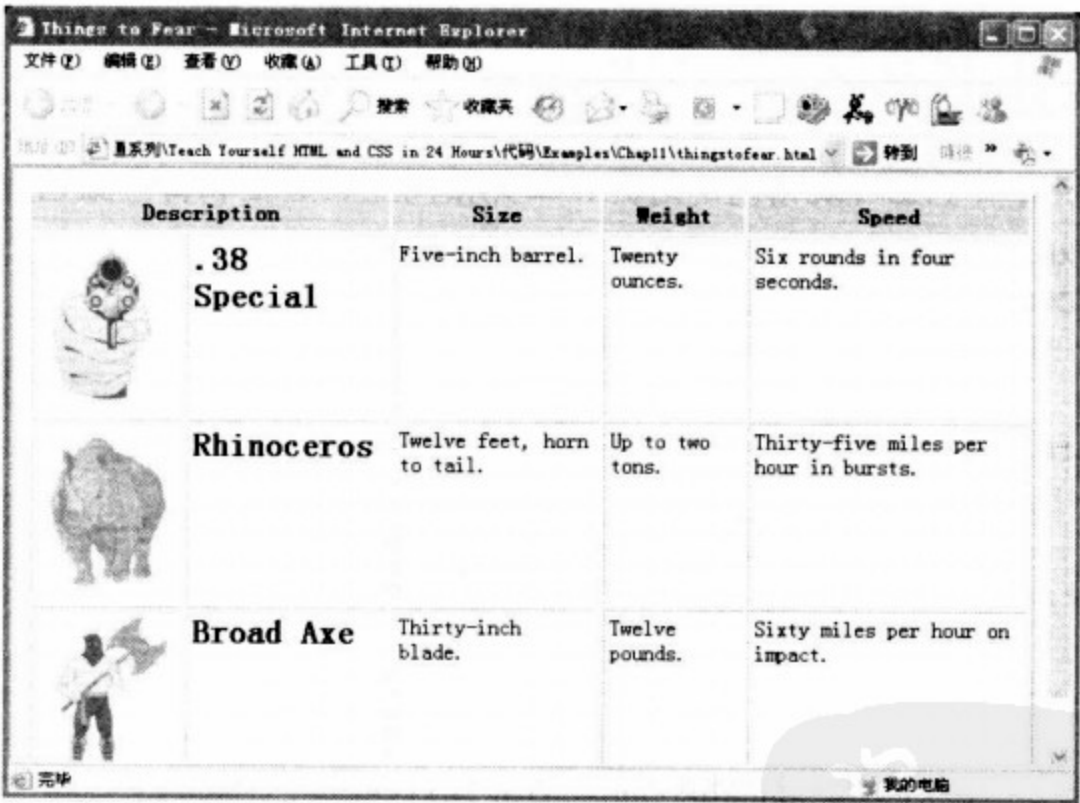
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Things to Fear</title>
  </head>

  <body>
    <table border="2" cellpadding="4" cellspacing="2">
      <tr style="background-color:silver">
        <th colspan="2">Description</th>
        <th>Size</th>
        <th>Weight</th>
        <th>Speed</th>
      </tr>
      <tr style="vertical-align:top">
        <td></td>
        <td><h2>.38 Special</h2></td>
        <td>Five-inch barrel.</td>
        <td>Twenty ounces.</td>
```

```
<td>Six rounds in four seconds.</td>
</tr>
<tr style="vertical-align:top">
  <td></td>
  <td><h2>Rhinoceros</h2></td>
  <td>Twelve feet, horn to tail.</td>
  <td>Up to two tons.</td>
  <td>Thirty-five miles per hour in bursts.</td>
</tr>
<tr style="vertical-align:top">
  <td></td>
  <td><h2>Broad Axe</h2></td>
  <td>Thirty-inch blade.</td>
  <td>Twelve pounds.</td>
  <td>Sixty miles per hour on impact.</td>
</tr>
</table>
</body>
</html>
```

图 11.2  
程序清单 11.2 中的 colspan 属性使左上角的单元格跨越多列



Did you Know?

**提示:**下面是最常用的 vertical-align 样式属性值: top、middle、bottom、text-top、text-bottom 和 baseline (用于文本)。这些属性值让你能够灵活地控制表格数据的垂直对齐方式。

在图 11.2 的顶部，一个单元格（Description）跨越两列。这是因为该单元格的<th>标签指定了属性 colspan="2"。也可用 rowspan 属性来创建跨越多行的单元格。

跨越（spanning）是使单元格扩展到占据表格中多行或多列的过程。Colspan 属性使单元格跨越多列；rowspan 使单元格跨越多行。

**提示：**创建含有跨越多行或多列的单元格的表格时，记住行和列的组织结构可能是最困难的事。通常一个很小的错误就能使整个表格排列错误。编写表格的 HTML 代码前，先在纸上画出表格，可以节省很多时间，避免造成困惑。

**Did you Know?**

也可以使用可视化的网页设计软件来排列表格中的行和列。这使表格的创建容易得多，只要选择的软件（如 Microsoft FrontPage 或 Macromedia Dreamweaver）能够创建格式良好、易于编辑的 HTML。

## 11.4 表格背景和间距

程序清单 11.2 中还有一些技巧我没有介绍。可以指定整个表格——以及表格中的每行或每个单元格——的背景，该背景可能与网页本身使用的背景不同。为此，可在标签 <table>、<tr>、<td>和<th>中指定样式 background-color 或 background-image，就像在<body>标签中一样（参见第 9 章）。例如，要使整个表格都使用黄色背景，可用<table style="background-color:yellow">或等价的<table style="background-color:#FFFF00">。

与 background-color 类似的是 background-image 属性，后者用于设置一个图像作为表格背景。要将图像 leaves.gif 设置为表格背景，可用<table style="background-image:url(leaves.gif)">。注意，图像文件名放在括号中，前面加上前缀 url，表明描述的是图像文件位置。

对表格的设置不仅可以使使用 style 属性。例如，可用 cellpadding 和 cellspacing 属性来控制表格边框的间距。cellspacing 属性设置表格边框之间和表格单元格之间的间距（以像素为单位）；cellpadding 属性设置单元格中的信息四周的间距（也是以像素为单位）。如果将 cellpadding 属性设置为 0，将使表格中的所有信息尽量接近表格边框，甚至接触到边框。cellpadding 和 cellspacing 属性让你能够全面控制表格的外观。

**注意：**虽然在 XHTML 中仍允许使用 cellpadding 和 cellspacing 属性，但 CSS 中存在与这两个属性等价的样式属性 padding 和 border-spacing。然而，当前的网页浏览器对这些样式属性的支持不一致，因此建议现在仍使用属性 cellpadding 和 cellspacing 来调整表格的间距。

**By the Way**

图 11.2 显示了背景颜色和间距的影响，图中的表头使用银色背景，表格中每个单元格的文本与图像离边框有一定的间距。

**提示：**可以将表格放在另一个表格的单元格中，这样内部的表格将拥有“父”表格的质量。换句话说，可以在表格中嵌套表格。

**Did you Know?**

使用嵌套表格可以设计许多有创意的网页布局。例如，如果要使一行文本显示在表格的左边，可以创建一个两列的表格，将文本放在一行，子表格放在另一行，如下所示：

```
<table>
  <tr>
    <td>To the right, you see all our telephone numbers.</td>
```

```
<td>
  <table border="1">
    <tr>
      <td>voice</td>
      <td>802-888-2828</td>
    </tr>
    <tr>
      <td>fax</td>
      <td>802-888-6634</td>
    </tr>
    <tr>
      <td>data</td>
      <td>802-888-3009</td>
    </tr>
  </td>
</tr>
</table>
</table>
```

注意到内部表格有边框，而外部表格没有。  
嵌套表格的一个缺点是，如果嵌套太深，浏览器渲染网页时可能很慢。如果没有很好的理由，嵌套的级别不应超过三级，尽量控制在两级以内。

### 11.5 使用表格设计有创意的网页布局

表格的一种常见用途是将文本和数字排列为表格。网页设计人员认识到表格的用途不限于此，还用于排列整个网页。例如，没有边框的表格可用于排列文本和图像。看一下程序清单 11.3，这个网页现在使用了表格进行更有效的布局。结果如图 11.3 所示。

**By the Way**

**注意：**虽然使用看不见的表格进行网页布局是一种很流行也很有用的方法，但从技术上说，万维网联盟（World Wide Web Consortium，W3C）——负责 Web 标准的组织——并不提倡这样做。W3C 提倡的方法是使用样式表而不是表格进行网页布局。虽然我基本上同意 W3C，然而，表格对网页布局非常有用，不可能完全不使用表格。不要担心，我将在第 14 章介绍如何将样式表用于网页布局。样式表的功能比表格强大，应将其用于设计复杂的网页。然而，表格适用于简单的布局以及使用样式表不方便实现的布局。举个例子，对于三列的页面使用样式表很难正确地编码。

程序清单 11.3 使用表格更有效地组织多列网页布局

```
<table cellpadding="5">
  <tr>
    <td colspan="3">
      <div style="font-family:verdana, arial; font-size:18pt; font-weight:bold">
        16 - Terry Lancaster</div>
```

```

    </td>
</tr>
<tr style="height:10px">
    <td>
        
    </td>
    <td style="vertical-align:top">
        <div style="font-family:verdana, arial; font-size:12pt; color:navy">
            <span style="font-weight:bold">Nickname:</span> Big T<br />
            <span style="font-weight:bold">Position:</span> RW<br />
            <span style="font-weight:bold">Height:</span> 6' 3"<br />
            <span style="font-weight:bold">Weight:</span> 195<br />
            <span style="font-weight:bold">Shoots:</span> Left<br />
            <span style="font-weight:bold">Age:</span> 40<br />
            <span style="font-weight:bold">Birthplace:</span>
            <a href="http://en.wikipedia.org/wiki/Nashville%2C_Tennessee">
                Nashville, TN</a>
        </div>
    </td>
    <td style="vertical-align:top">
        <div style="font-family:verdana, arial; font-size:12pt; color:navy">
            <span style="font-weight:bold">Favorite NHL Player:</span>
            <a href="http://www.nhl.com/players/8448091.html">Brett Hull</a><br />
            <span style="font-weight:bold">Favorite NHL Team:</span>
            <a href="http://www.nashvillepredators.com/">Nashville Predators</a>
            <br /><span style="font-weight:bold">Favorite Southern Fixin:</span>
            <a href="http://southernfood.about.com/od/potatorecipes/r/blbb442.htm">
                Skillet Fried Potatoes</a><br />
            <span style="font-weight:bold">Favorite Meat and Three:</span>
            <a href="http://www.hollyeats.com/Swetts.htm">Swett' s</a>
            (<a
href="http://maps.google.com/maps?q=2725+clifton+ave,+nashville,+tn"
rel="external">map</a>)
            <br />
            <span style="font-weight:bold">Favorite Country Star:</span>
            <a href="http://www.patsycline.com/">Patsy Cline</a><br />
            <span style="font-weight:bold">Favorite Mafia Moment:</span>
            "<a href="mcmplayer_chale.html">Chet</a> finishing the game with his
            eyelid completely slashed through."
        </div>
    </td>
</tr>
<tr>
    <td colspan="3">
        <table style="width:100%; text-align:right; font-family:verdana, arial;
font-size:11pt; color:navy" border="1" >
            <tr style="background-color:navy; color:white">
                <th style="text-align:left">Season</th>

```



```
<th>GP</th>
<th>G</th>
<th>A</th>
<th>P</th>
<th>PIM</th>
<th>PPG</th>
<th>SHG</th>
<th>GWG</th>
</tr>
<tr style="background-color:white">
  <td style="text-align:left">Summer 2005</td>
  <td>11</td>
  <td style="width:75px">7</td>
  <td>5</td>
  <td>12</td>
  <td>0</td>
  <td>0</td>
  <td>0</td>
  <td>0</td>
</tr>
<tr style="background-color:#EEEEEE">
  <td style="text-align:left">Winter 2004</td>
  <td>24</td>
  <td>14</td>
  <td>14</td>
  <td>28</td>
  <td>2</td>
  <td>0</td>
  <td>0</td>
  <td>5</td>
</tr>
<tr style="background-color:white">
  <td style="text-align:left">Summer 2004</td>
  <td>18</td>
  <td>9</td>
  <td>9</td>
  <td>18</td>
  <td>2</td>
  <td>0</td>
  <td>0</td>
  <td>2</td>
</tr>
<tr style="background-color:#EEEEEE">
  <td style="text-align:left">Spring 2004</td>
  <td>19</td>
  <td>7</td>
  <td>17</td>
  <td>24</td>
```

```

        <td>0</td>
        <td>0</td>
        <td>0</td>
        <td>1</td>
    </tr>
</table>
</td>
</tr>
<tr>
    <td colspan="3">
        <div style="font-family:verdana, arial; font-size:12pt">
            <a href="mailto:l&#97;ncastert@musiccitym&#97;fi&#97;.com?subject=
            Fan Question&amp;body=What' s your secret?">Contact Terry.</a>
        </div>
    </td>
</tr>
</table>
```

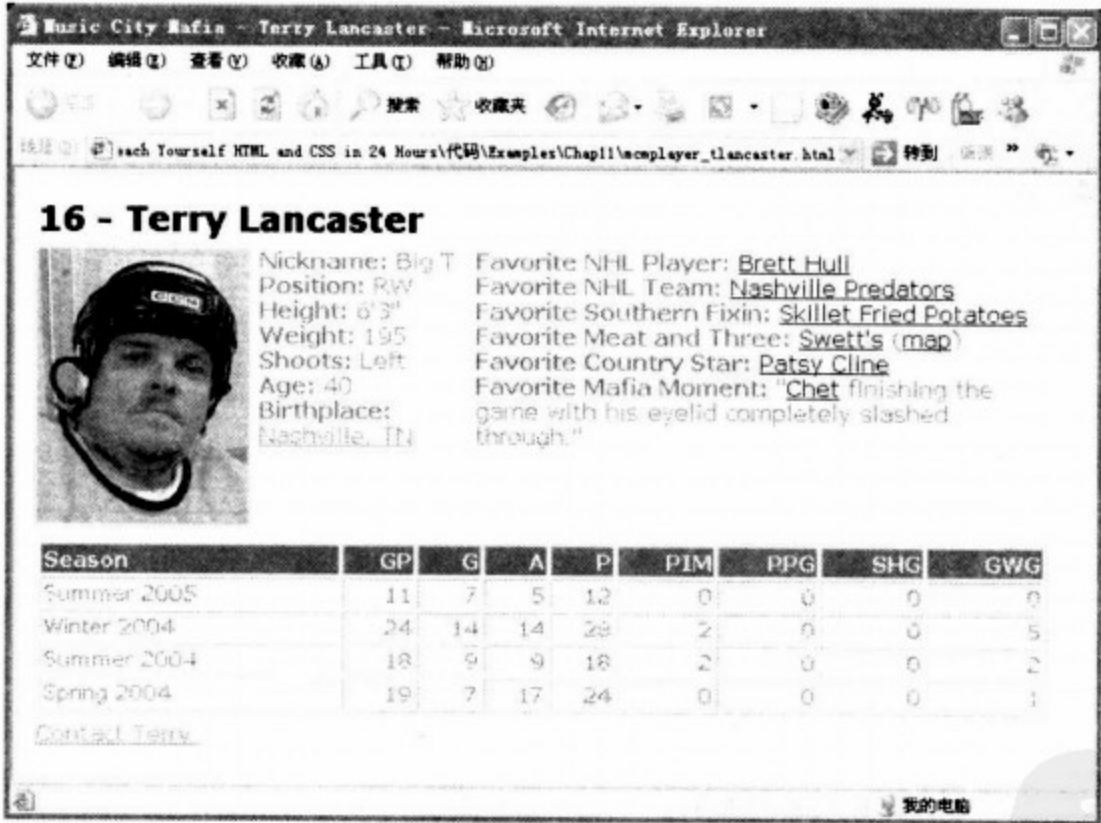


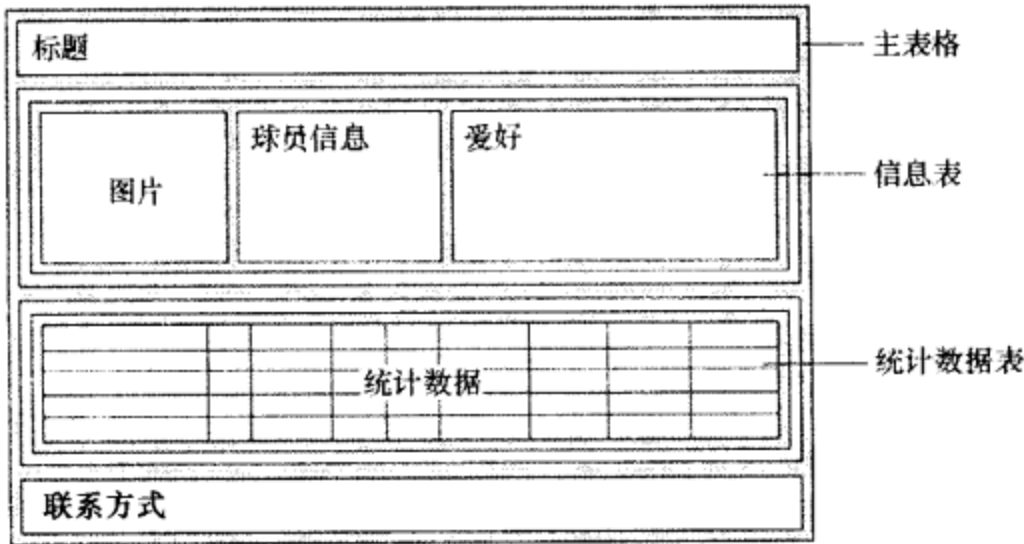
图 11.3  
HTML 表格让你能够更灵活地控制网页布局

这个程序清单包含大量的代码，但其中大部分读者都见过。这些是曲棍球队员网页之一的代码，只是这里使用了表格来实现更有效的网页布局。我在创建表格时，将表格边框设置为可见的，这样可保证信息排列得当。在将这个表格放在最后的网页中之前，我删除了<table>标签中的 border="1"属性，使边框不可见。

程序清单 11.3 和图 11.3 实际上有三个不同的表格。第一个表格将图像和文本排列成四行。第二个表格占据了其中的第二行，提供了三列信息。最后一个表格是曲棍球统计表，占据了主表格中的一行。

如果想象不出曲棍球队员表格的布局，请参见图 11.4，它说明了该页面中表格之间的关系。

图 11.4  
将不包含实际内容的  
表格排列呈现出来，有助于设计表格布局



**Did you know?** **提示：** 关于如何使用表格设计有创意的布局，可参考 <http://www.sampublishing.com/> 上的示例 “The Site of the '90s”。读者的网站看上去可能比 LOOK 网站要保守一点——不过有些读者可能有更疯狂的想法。

## 11.6 总结

本章读者学习了如何将文本和图像排列成由行和列组成的表格。学习了用于创建表格的三个基本标签以及用于控制表格的对齐方式、间距与外观的众多可选属性和样式。还看到多个表格可一起使用，一个表格可嵌套到另一个表格中，从而实现更多样化的布局。

表 11.1 总结了本章介绍的标签与属性。

表 11.1 第 11 章介绍的 HTML 标签与属性	
标签/属性	功能
<table>...</table>	创建表格，可以包含任意数量的行（<tr>标签）
属性	
border="宽度"	指定表格边框宽度，单位为像素，如果 border="0"或省略该属性，边框将不显示
cellspacing="间距"	表格单元格之间的间距，单位为像素
cellpadding="填充距离"	表格单元格边框同其内容的间距，单位为像素
style="width:宽度"	表格在网页中的宽度，可以是像素数或网页宽度的百分比
style="height:高度"	表格在网页中的高度，可以是像素数或网页高度的百分比
style="background-color:颜色"	表格与没有背景颜色的单元格的背景颜色
style="backgroundimage:url (图像 url) "	表格与没有背景图像的单元格的背景图像（如果也指定了背景颜色，背景颜色将透过图像的透明区域显示出来）
<tr>...</tr>	定义表格的一行，可以包含一个或多个单元格（<td>标签）
属性	
style="text-align:对齐方式"	该行中单元格内容的水平对齐方式，可能的值为 left、right 和 center
style="vertical-align:对齐方式"	该行中单元格内容的垂直对齐方式，可能的值为 top、middle 和 bottom
style="background-color:颜色"	该行中所有没有背景图像的单元格的背景颜色
style="backgroundimage:url (图像 url) "	该行中所有没有背景图像的单元格的背景图像
<td>...</td>	定义表格中的一个数据单元格

续表

标签/属性	功能
属性	
style="text-align: 对齐方式"	该单元格内容在水平方向上的对齐方式，可能的值为 left、right 和 center
style="vertical-align: 对齐方式"	该单元格内容在垂直方向上的对齐方式，可能的值为 top、middle 和 bottom
rowspan="行数"	该单元格跨越的行数
colspan="列数"	该单元格跨越的列数
style="width: 宽度"	该列单元格的宽度，可以是像素数或表格宽度的百分比
style="height: 高度"	该行单元格的宽度，可以是精确的像素或表格高度的百分比
style="background-color: 颜色"	该单元格的背景颜色
style="backgroundimage:url ( 图像 url) "	在这个单元格中显示的背景图像
<th>...</th>	定义表头单元格（可接受的属性与样式同<td>一样）

## 11.7 问与答

问：我制作了一个大表格，加载这个网页时，很长时间都没有任何显示，为什么会这样？

答：因为网络浏览器需要计算出表格中每部分的大小才能将它显示出来，所以复杂的表格需要一段时间才能显示在屏幕上。对表格中的每个图像指定 width 和 height 可稍微提高速度。在<table>和<td>标签中使用 width 属性也会有所帮助。

问：我注意到网上许多网页的表格，有一个单元格是可以改变的，而其他单元格保持不变。如何做到这一点？

答：这些网页使用了框架（frame）而不是表格。框架与表格有点相似，但每个框架包含一个独立的 HTML 网页，且可独立于其他框架进行更新。新的浮动框架（iframe）实际上可以放在表格中，因此，虽然创建它们的 HTML 不同，但看起来像常规表格。你将在第 16 章学习如何创建和使用框架。

问：XHTML 没有新的表格标签吗？为什么没有提及这些标签？

答：最新的 XHTML 标准确实引入了一些新表格标签，但本书没有讨论。这些扩展的主要用途是提供一些现在的网页浏览器还不完全支持的高级特性，例如，本身有滚动条的表格以及方便有视力障碍的用户阅读的表格。如果你对这些特性感兴趣，可在 W3C 网站（<http://www.w3.org/>）找到有关这些新标签的信息。新标签不会直接影响表格在现有网页浏览器中的显示。

别担心，无论现在还是将来，新标签都不会使本章介绍的表格标签过时。这些标签将像现在一样继续发挥作用。

## 11.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 11.8.1 测验

1. 如何创建包含两列并排文本的网页？
2. 你认为为问题 1 创建的列靠得太近，如何在它们之间加入 30 像素的间距？
3. 编写 HTML 代码创建如下图所示的表格。



### 11.8.2 答案

1. 使用如下表格：

```
<table>
  <tr>
    <td style="vertical-align:top">
      First column of text goes here...
    </td>
    <td style="vertical-align:top">
      Second column of text goes here...
    </td>
  </tr>
</table>
```

2. 将 `cellspacing="30"` 加入到 `<table>` 标签中；也可以使用 `cellpadding="15"` 在每列边缘加入 15 像素的间距。

3.

```
<table border="5">
  <tr>
```



```
<td rowspan="3">A</td>
<td colspan="3">B</td>
</tr>
<tr>
<td>E</td>
<td>F</td>
<td rowspan="2">C</td>
</tr>
<tr>
<td colspan="2">D</td>
</tr>
</table>
```

## 11.9 练习

1. 可以使用一个只有一个单元格且带有边框的表格在网页任何文本四周绘制一个矩形。将这个表格嵌套到另一个双列表格中，就可以在网页的左边或右边创建旁注。同时使用正文和旁注在印刷品中非常常见，你可能也想在网页这样使用。尝试在网页中使用这样的布局。

2. 你有没有这样的网页：不同的访问者可能对其中不同的信息感兴趣？使用表格呈现两列文本，每列都有表头（也可能包含图像）。这样，每个人感兴趣的信息都将在网页显示时就出现在顶端。



## 第 12 章

# 使用 CSS 格式化网页

在前一章，读者学习了如何使用表格将信息排列为行和列。表格作为网页布局工具曾被大量用于 Web，但随着样式表的出现，表格的这种用途已被摒弃。表格仍很有用，但主要用于其原始目的，也就是将信息排列为行和列。除取代表格作为网页布局机制外，样式表也是指定字体、控制边距、指定边框和调整网页外观的首选方法。

样式表背后的概念很简单：创建一个独立的样式表文档，以指定字体、颜色、背景和其他确定网站外观的特征。然后链接需要使用该样式表的所有网页，而不是在每个文档中都重复指定这些样式。决定修改字体或颜色主题时，只要修改一两个样式表，就可以一次性修改所有网页。因此，样式表是一组格式化指令，能够同时控制众多 HTML 页面的外观。

样式表的作用不仅是将网页内容同格式分离。CSS 样式表能够设置许多以前没有大量工作不可能实现的格式化特征，这包括精确的字体控制、字间距和行间距、页面边距和边框以及对非欧洲语言和字符的扩展支持。样式表还能够以人们熟悉的英寸、毫米、点和 `picas` 等来指定大小和其他度量值。你还可以使用样式表正确地控制图像和文本在网页上的位置。

总之，样式表将面向纸张的复杂的出版机制引入了 Web。样式表确实实现了这一点。

**By the Way**

**注意：**如果有 3 个或更多的网页使用类似的格式和字体，可能想在阅读本章时为它们创建一个样式表。即使不想创建完整的样式表，将样式直接应用于网页中的 HTML 元素也很有帮助。

### 12.1 理解 CSS

样式表背后的技术称为 CSS，即级联样式表（Cascading Style Sheets）。CSS 是一种定义样式（如字体、颜色和位置）的语言，用于描述如何格式化和显示网页中的信息。CSS 样式可直接存储在 HTML 网页中，也可存储为独立的样式表文件。不管以哪种方式保存，样式表都包含将样式应用于特定元素类型的样式规则。在外部使用时，样式表规则放在外部样式表文档中，文件扩展名为 `.css`。

样式规则是可应用于网页中元素，如文本段落或链接的格式化指令。样式规则由一个或多个样式属性及其值组成。内部样式表直接放在网页中，外部样式表保存在独立的文档中，网页通过一个特殊标签链接外部样式表——稍后将介绍这个标签。

名称 CSS 中的“级联 (cascading)”表示样式表规则应用于 HTML 文档元素的方式。具体地说，CSS 样式表中的样式形成一个层次结构，更具体的样式覆盖通用样式。样式规则的优先级由 CSS 根据这个层次结构决定，从而实现级联效果。如果上述解释还不是很清楚，可将 CSS 的级联机制看成一种类似继承的关系，在这种继承关系中，父亲的特征传递给子女，但子女有更特殊的特征；基样式规则适用于整个样式表，但可被更具体的样式规则覆盖。

用一个简单例子可能更能够解释清楚。看下面的代码，你是否能够说出文本的颜色：

```
<div style="color:green">
  This text is green.
  <p style="color:blue">
    This text is blue.
  </p>
  <p>
    This text is still green.
  </p>
</div>
```

在这个例子中，颜色 green 通过样式属性 color 应用于<div>标签。因此<div>标签中的文本是绿色的。由于<p>标签是<div>标签的子标签，因此绿色的文本样式传递给<p>中的文本。然而，第一个<p>标签覆盖了 color 样式，将它改为蓝色。最后的结果是，第一段文本是蓝色的，第二段文本仍为传递而来的绿色。

**注意：**你可能注意到了，我在本章和下面几章中大量使用了术语“元素”。元素就是网页中的信息（内容），如图像、段落或链接。标签用于对元素进行编码，可以将元素看成标签及其内部的描述信息（属性、文本、图像等）。

**By the  
Way**

与许多 Web 技术一样，CSS 已经发展很多年了。CSS 最初的版本称为 Cascading Style Sheets Level 1 (CSS1)，创建于 1996 年。后来的 CSS2 标准创建于 1998 年，虽然这似乎是很久以前了，但直到最近 CSS2 仍得到所有网页浏览器的支持。由于 CSS 的强大功能，浏览器的支持始终是一个问题。幸运的是，浏览器最后支持了大部分功能，现在你可以安全地使用 CSS2 样式表，不用担心太多事情。因此，当我在本书中谈到 CSS 时，指的是 CSS2。

**警告：**即使在最新的网页浏览器中，仍有一些 CSS2 特性没有得到正确支持。我将在本书接下来的几章中指出这些特性及相关的解决方法。

**Watch  
Out!**

在 <http://www.w3.org/Style/CSS/> 上可以找到关于样式表语言 CSS1 和 CSS2 的完整参考，还可以找到各个 CSS 版本都包含哪些样式。本章接下来将解释如何以与最新的网页浏览器兼容的方式使用这些参考文档中的信息。

## 12.2 一个基本样式表

虽然样式表的功能很强大，但要创建它们也可以很简单。请看图 12.1 和图 12.2 所示的网页。

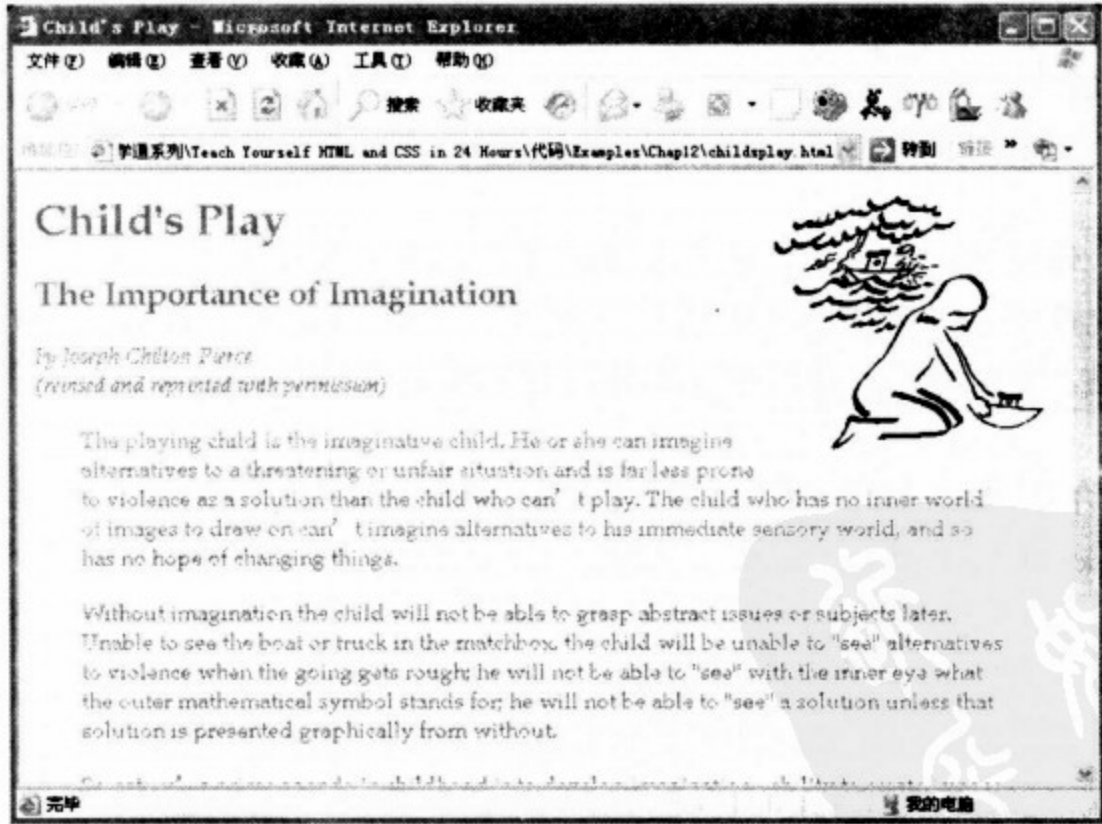
图 12.1

这个网页使用样式表来调整文本和背景的外观和间距



图 12.2

这个网页使用与图 12.1 相同的样式表，因此两个网页的风格相同



这两个网页的一些视觉属性是相同的，可将其放在一个样式表中：

- 它们的正文都使用 Book Antiqua 字体，标题使用 Prose Antique 字体；
- 它们都使用 parchment.jpg 作为屏幕背景图像；
- 所有文本的颜色都是栗色（在彩色屏幕上，在本书中不是）；

- 它们的边距都比较大，正文都缩进；
- 它们的行间距都很大；
- 脚注都居中，且字体很小（在图中看不到）。

其中的有些属性，如文本颜色、背景图像和居中的小字体，都可以通过（摒弃的）普通 HTML 标签来实现。其他属性，如行间距和大边距，使用标准 HTML/XHTML 无法实现。现在，所有这些属性都可以用样式表轻松地实现。

程序清单 12.1 是一个指定这些属性的样式表。

程序清单 12.1 外部样式表可以指定任意数量网页的属性

```
body {
    font-size:12pt;
    font-family:Book Antiqua;
    color:maroon;
    line-height:16pt;
    margin-left:10pt;
    margin-right:10pt;
    background:url(parchment.gif);
}

p {
    margin-left:24pt;
    margin-right:24pt;
}

h1{
    font:24pt Prose Antique, Lucida Handwriting;
    font-weight:bold;
    line-height:30pt;
}

h2{
    font:18pt Prose Antique, Lucida Handwriting;
    font-weight:bold;
    line-height:22pt;
}

a {
    text-decoration:none;
}

a:link {
    color:red;
}

a:visited {
```



```

    color:red;
}

a:active {
    color:black;
}

div.byline {
    font-style:italic;
}

div.footnote {
    font-size:9pt;
    line-height:12pt;
    text-align:center
}

```

一开始你可能觉得这些代码很多，然而，如果仔细看，将看到每行代码其实没有多少信息。将每个样式规则放在一行上，是相当标准的做法，使样式表更易于阅读。谈到代码可读性，可能你看到这段样式表代码的第一反应是，它并不像普通 HTML 代码。CSS 使用自己的语言来指定样式表。

当然，样式表中也有一些熟悉的 HTML 标签。你可能猜到了，这个样式表中的 `body`、`p`、`h1`、`h2` 和 `div` 指的是将应用该样式表的 HTML 文档中对应的标签。每个标签名后的大括号内是对如何显示标签中内容的说明。

在这个例子中，可能的话所有 `body` 文本应显示为 12 点，使用 Book Antiqua 字体，颜色为栗色，行间距为 16 点。页面边距为 10 点，背景为相对 URL 中的 `parchment.jpg` 图像。

### By the Way

**注意：**这个样式表使用了样式属性 `background` 来设置背景图像。另外还有 `background-color`、`background-image`、`background-repeat`、`background-position` 属性，可用于设置背景的各个方面。只设置平铺背景图像或不透明的背景颜色时，可使用 `background` 或更具体的 `background-image`、`background-color` 样式。

以 `<p>` 标签开始的任何段落左右两边（边距）都将缩进 24 点。在 `<h1>` 或 `<h2>` 标签内的文本将显示为粗体，字体为 Prose Antique，大小分别为 30 点和 22 点。如果用户没有安装 Prose Antique 字体，将使用 Lucida Handwriting 字体。

在程序清单 12.1 中，每个数字后的 `pt` 表示 point（点，1 英寸相当于 72 点）。也可以用英寸（`in`）、厘米（`cm`）、像素（`px`）或字母 `m` 的宽度（`em`）等作为样式表中的度量单位。

你可能注意到了，程序清单中的每个样式规则都以分号（`;`）结束。分号用于分隔样式规则。在每条样式规则后加上分号已成为一种惯例，这样便于在样式规则后加上另一条规则。

### By the Way

**注意：**使用样式表可以将字体大小指定为任意值，但有些显示设备和打印机可能不能正确处理超过 200 点的字体。

要在 HTML 文档中链接样式表，可在文档的<head>部分包含一个<link />标签。程序清单 12.2 是图 12.1 所示页面的 HTML 代码，它包含下面的<link />标签：

```
<link rel="stylesheet" type="text/css" href="styles.css" />
```

这行代码假定样式表存储在该 HTML 文档所在文件夹下的文件 styles.css 中。只要网页浏览器支持样式表——现在所有浏览器都支持，样式表指定的属性就将应用于网页的内容，而不需要特殊的 HTML 格式化代码。这与 XHTML 的最终目标相符，XHTML 旨在将网页内容与显示这些内容使用的具体格式分离。

**程序清单 12.2** 图 12.1 所示网页并没有包含格式化代码，因为所有格式化工作都留给了外部样式表

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Features of a Waldorf Classroom</title>
    <link rel="stylesheet" type="text/css" href="styles.css" />
  </head>

  <body>
    <h1>Distinguishing Features of a Waldorf School Classroom</h1>
    <div class="byline">
      by Scott Dorwart and P.J. Long
    </div>
    <p>
      There are more than 500 independent Waldorf schools in 32 countries.
      Though each one is unique, a visitor would observe several common
      characteristics which distinguish the Waldorf approach to learning.
      Most notable among these would be how the curriculum directly responds
      to each phase of child development.
    </p>
    <p>During the elementary years, it is the educator's task to
    transform all that the child needs to know about the world into the
    language of the imagination—encouraging wonder, curiosity,
    reverence, and a love of learning. This principle lies at the heart of
    the most common features which distinguish Waldorf from other forms of
    education:</p>
    <ol>
      <li>All learning proceeds from a wholistic vision of the child.</li>
      <li>Lessons derive from an arts-based, integrated curriculum.</li>
      <li>Expectations for learning are developmentally appropriate.</li>
      <li>The teacher remains with the class as the children grow.</li>
      <li>No grades are assigned to students' work.</li>
      <li>Behavior is managed through creative, positive discipline.</li>
```

```

    <li>The classroom provides a cooperative social environment.</li>
</ol>
<div class="footnote">
    <hr />
    <span style="font-style:italic">Scott Dorwart is a carpenter and father
    of two children who attend the Green Mountain Waldorf School. P.J. Long
    is a psychotherapist and mother of two.</span>
    <hr />
    <a href="http://netletter.com/GMWS/HHH.htm">Head, Heart, Hands: A
    Waldorf Family Newsletter</a><br />
    Published by <a href="http://netletter.com/GMWS/GMWS.htm">The Green
    Mountain Waldorf School</a>
    <hr />
</div>
</body>
</html>

```

### By the Way

**注意：**在大多数网页浏览器中，可以打开.css文件，并选择记事本或其他文本编辑器来查看该文件，从而查看样式表中的样式规则。要确定.css文件的文件名，可查看链接该文件的HTML源代码。要编辑自己的样式表，只要使用文本编辑器就可以了。

虽然CSS在现代网页浏览器中得到广泛支持，但在以前的浏览器中并非如此。另外，并不是每个浏览器对CSS的支持都没有缺陷。要比较主要浏览器在支持CSS方面的差别，可参考网站 [http://www.westciv.com/style\\_master/academy/browser\\_support/](http://www.westciv.com/style_master/academy/browser_support/)。

程序清单 12.2 中的代码很有趣，它没有包含任何格式化代码。换句话说，没有HTML代码用于指示文本和图像应如何显示——没有颜色、字体，什么格式都没有指定。然而，由于链接了外部样式表 style.css，网页是经过仔细格式化的。这种方式的优点在于，当网站包含许多网页且要求网页的风格一致时，使用这种方式将很容易。另外还有一个优点，就是可以将页面的样式分离出来，存储在一个独立的文档（样式表）中，这样在一个地方修改将影响所有页面。

## 12.3 CSS 样式初步

现在你已经具备了CSS样式表的基础知识，知道样式表如何根据样式规则来描述网页信息的外观。接下来的几章将介绍一些最重要的样式属性，并教你开始在自己的样式表中使用CSS。

CSS 包括各种样式属性，用于控制字体、颜色、对齐方式和边距，这些只是网页样式的一些方面。CSS 中的样式属性分为两大类：

- 布局属性；
- 格式化属性。

布局属性包括影响网页中元素位置的属性。例如，布局属性可以控制内容的宽度、高度、边距、填充和对齐方式，甚至可以控制内容在网页中的准确位置。最后一点仅使用 HTML 代码是无法实现的。

### 12.3.1 布局属性

CSS 布局属性用于决定内容在网页上如何排列。最重要的布局属性之一是 `display`，它描述元素相对于其他元素如何显示。`display` 属性有 4 个可能的取值。

- `block`: 元素显示在下一行，作为一个新段落。
- `list-item`: 元素显示在下一行，前面有列表元素标记（项目符号）。
- `inline`: 元素显示在当前段落内。
- `none`: 元素隐藏不显示。

将元素在网页上的显示想象成占据一块矩形区域，就更容易理解 `display` 属性：`display` 属性控制该矩形区域如何显示。例如，`block` 值使元素显示在下一行中，而 `inline` 使元素显示在之前的内容后。`display` 是几个能够在大多数样式规则中使用的几个样式属性之一。下面的例子演示了如何设置 `display` 属性：

```
display: block;
```

**注意：**`display` 属性依赖于相对位置的概念，相对位置指元素相对于网页中其他元素进行定位。CSS 也支持绝对位置，将元素放在网页中特定的位置，而不考虑其他元素。你将在第 14 章学到这两种定位。

**By the  
Way**

`width` 和 `height` 属性控制元素矩形区域的大小。同许多与大小相关的 CSS 属性一样，`width` 和 `height` 属性的值可使用几种不同的度量单位：

- `in`（英寸）；
- `cm`（厘米）；
- `px`（像素）；
- `cm`（厘米）；
- `pt`（点）；
- `mm`（毫米）。

在同一个样式表中，可以使用不同的度量单位，但类似的样式属性最好使用一致的单位。例如，字体大小属性的度量单位统一用点，尺寸统一用像素。下面是一个以像素为单位设置元素宽度的例子：

```
width: 200px;
```

### 12.3.2 格式化属性

CSS 格式化属性用于控制网页内容的外观，而不是物理位置。最常用的格式化属性之一

是 `border`，它用于在元素四周建立一个可见的方框或部分边框。下列 `border` 属性提供了描述元素边框的方法：

- `border-width`：边框的宽度；
- `border-color`：边框的颜色；
- `border-style`：边框的样式；
- `border-left`：边框的左边；
- `border-right`：边框的右边；
- `border-top`：边框的顶部；
- `border-bottom`：边框的底部；
- `border`：边框的所有边。

`border-width` 属性用于指定边框的宽度，通常以像素为单位，如下列代码所示：

```
border-width:5px;
```

属性 `border-color` 和 `border-style` 用于设置边框的颜色和样式。下面是一个如何设置这两个属性的例子：

```
border-color:blue;  
border-style:dotted;
```

`border-style` 属性可设置为下列值之一。

- `solid`：单实线。
- `double`：双实线。
- `dashed`：虚线。
- `dotted`：点线。
- `groove`：沟线（groove）。
- `ridge`：脊线（ridge）。
- `inset`：内陷边框。
- `outset`：外凸边框。
- `none`：无边框。

`border-style` 属性的默认值为 `none`，这就是为什么除非将 `border` 属性设置为其他类型，否则元素没有边框的原因。最常见的 `border` 样式是 `solid` 和 `double`。

### By the Way

**注意：**`border-style` 属性的默认值为 `none`，这一规则有个例外，就是当图像放在 `<a>` 标签内作为链接图像时。在这种情况下，默认自动设置实线边框。这就是为什么经常见到链接的图像使用样式 `boder-style:none`，将自动设置的边框去掉。

属性 `border-left`、`border-right`、`border-top` 和 `border-bottom` 让你能够单独为元素的每一边



设置边框。如果要使四周的边框都显示一致，只要使用 `border` 属性就可以了，后面接着下列样式：`border-width`、`border-style` 和 `border-color`。下面的例子使用 `border` 属性将边框设置为两条红线，宽度总共为 10 像素：

```
border:10px double red;
```

虽然元素边框的颜色由 `border-color` 属性设置，但元素内部区域的颜色由属性 `color` 和 `background-color` 设置。`color` 属性设置元素中文本的颜色（前景色），`background-color` 属性设置文本后面的背景色。下面的例子将这两个属性设置为预定义的颜色：

```
color:black;
background-color:orange;
```

也可以像在 HTML 中一样，通过指定十六进制或 RGB（红、绿、蓝）十进制值，为这些属性指定自定义的颜色：

```
background-color:#999999;
color:rgb(0,0,255);
```

**注意：**十六进制颜色在第 9 章详细介绍。

**By the Way**

控制网页内容的对齐和缩进也不麻烦，这是通过 `text-align` 和 `text-indent` 属性来实现的，如下列代码所示：

```
text-align:center;
text-indent:12px;
```

将元素对齐并缩进后，可能还要设置其字体。下列字体属性可用于设置与字体相关的各种参数。

- `font-family`：字体集。
- `font-size`：字体大小。
- `font-style`：字体样式（正常或斜体等）。
- `font-weight`：字体的粗细（细、中等、粗等）。

`font-family` 属性指定一个有优先级的字体集名称列表。使用有优先级的列表而不是一个值，旨在防止特定系统中没有安装指定的字体。`font-size` 属性使用度量单位——通常是点——来指定字体大小。最后，`font-style` 属性设置字体样式，`font-weight` 设置字体的粗细。下面是一个使用这些字体属性的例子：

```
font-family: Arial, sans-serif;
font-size: 36pt;
font-style: italic;
font-weight: medium;
```

知道许多关于样式属性及如何使用样式属性的知识后，再回过头去看一下程序清单 12.1，

看看是否能够更好地理解这些代码。

下面再复习一下该样式表用到的样式属性，帮助读者更好地理解样式属性。

- **font**: 让你能够一次指定许多字体属性。可以指定一个字体名称列表，名称之间用逗号分开；如果不能使用第一种字体，将尝试第二种，依此类推。也可以包含 **bold** 或 **italic** 及字体大小。每个字体属性都可以用单独的 **font-family**、**font-size**、**font-weight:bold** 和 **font-style:italic** 指定。
- **line-height**: 在出版界也称为 **leading**。它设置文本行的高度，通常以点为单位。
- **color**: 设置文本颜色，使用标准的颜色名称或十六进制颜色编码（详情请参见第 9 章）。
- **text-decoration**: 可用于删除链接的下划线——只要将该属性设置为 **none**。另外还支持 **underline**、**italic** 和 **line-through**。将样式应用于链接将在下一章更详细地介绍。
- **text-align**: 将文本向左 (**left**)、向右 (**right**) 或居中 (**center**) 对齐，同时使用一个 **justify** 值对每行文本进行排版。
- **text-indent**: 从左边距缩进指定的距离。可用 **px**、**in**、**cm**、**mm**、**pt** 和 **pc** 等单位来指定缩进距离，也可用页面宽度的百分比（如 20%）。
- **margin**: 将左边距和右边距设置为相同，这个值可用度量单位或页面宽度的百分比来指定。如果要分别设置左右边距，可分别使用 **margin-left** 和 **margin-right**。在第 14 章将学到更多关于这些样式属性的知识。
- **background**: 在文本后面显示颜色或图像，通过指定颜色或 **url(address)**，其中 **address** 指向背景图像文件。注意，这不仅可用于 **<body>** 标签，也可用于任何标签或文本块 (**span**)，以突出相应的区域。样式属性 **background-color** 和 **backgroundimage** 可实现相同的效果。

## 12.4 使用样式类

这是一本“自学”书，因此读者不必参加课程来学习如何创建优秀的网页样式，但需要学习什么是样式类。需要使网页中的有些文本与其他文本不同时，可以创建自定义 HTML 标签。定义的特殊格式文本称为样式类。样式类是一组自定义的格式化说明，可用于网页中的任何元素。

在演示样式类前，先解释一些 CSS 术语。首先，CSS 样式属性是一个可分配值的具体样式，如 **color** 或 **font-size**。通过选择符 (**selector**) 将样式规则及其值同网页中的元素关联起来。选择符用于标识网页中要应用样式的标签。下面是一个例子，在这个简单的样式规则中包含选择符、属性和值。

```
h1 { font:36pt Courier; }
```

在这行代码中，**h1** 是选择符，**font** 是样式属性，而 **36pt Courier** 是值。选择符很重要，因为它意味着该字体设置将应用网页中所有 **h1** 元素。但你可能想挑出一些 **h1** 元素，这时该怎么办？答案是使用样式类。

假设要在文档中使用两种不同的 **<h1>** 标签。可以在样式表中添加下列 CSS 代码，为每种

标签创建一个样式类：

```
h1.silly { font:36pt Comic Sans; }
h1.serious { font:36pt Arial; }
```

注意，这些选择符在 h1 之后包含一个点号 (.)，然后是一个描述性类名。在 HTML 页面中需要选择这两个样式类之一时，可使用 class 属性，如下所示：

```
<h1 class="silly">Marvin's Munchies Inc.</h1>
<p>
  Text about Marvin's Munchies goes here.
</p>
<h1 class="serious">MMI Investor Information</h1>
<p>
  Text for business investors goes here.
</p>
```

在 HTML 代码中引用样式类时，只要在元素的 class 属性中指定类名。在这个例子中，如果在网页开头包含一个指向样式表的<link />标签，且用户安装了 Comic Sans 字体，Marvin's Munchies Inc.将显示为 36 点的 Comic Sans 字体；而 MMI Investor Information 将显示为 36 点的 Arial 字体。

如果要创建可应用于任何元素的样式表，而不仅仅应用于标题或其他特定标签，该如何办呢？可以将样式类与<div>标签相关联（第 5 章介绍过，该标签将文本放在类似于文本段落的块中）。

使用 div 选择符加上点号以及样式类名和样式说明，可创建自定义的 HTML 标签。这种标签可同时控制任意数量的字体、间距和边距设置。要在网页中应用自定义标签，可使用<div>标签和 class="创建的类名"。

例如，程序清单 12.1 中的样式表包括下面的样式类说明：

```
div.footnote {
  font-size:9pt;
  line-height:12pt;
  text-align:center;
}
```

**提示：**你可能注意到了，样式规则包含多个属性时，样式编码将稍有不同。对于只包含一个属性的样式规则，通常属性与规则放在同一行，如下所示：

```
div.footnote { font-size:9pt; }
```

然而，样式规则包含多个样式属性时，如果每个属性都单独一行，代码将更易于阅读和理解，如下所示：

```
div.footnote {
  font-size:9pt;
  line-height:12pt;
  text-align:center;
}
```

**Did you  
Know?**

在程序清单 12.2 中，使用下面的标签应用该样式表：

```
<div class="footnote">
```

在程序清单 12.2 中，<div>标签和结束标签</div>之间的文本内容都显示为 9 点并居中对齐，行间距为 12 点。

使用样式类的优点在于，它们将样式代码同网页分离，从而将 HTML 代码用于网页内容，而不是这些内容在屏幕上如何显示。然后，可以集中精力来调整样式表，控制内容如何显示。很少的样式表代码就能对整个网站产生很大的影响。这使得网页更易于维护和操作。

## 12.5 内部样式表和内联样式

在有些情况下，你可能希望指定只用于一个网页的样式，在这种情况下，可将样式表放在标签<style>和</style>内，直接包含在 HTML 文档中。以这种方式使用的样式表必须出现在 HTML 文档的 head 中。不需要<link />标签，在其他网页中不能引用该样式表（除非将其复制到该文档的 head 部分）。这种样式表称为内部样式表。

程序清单 12.3 是一个如何指定内部样式表的例子。

程序清单 12.3 演示如何使用内部样式表的小型网页

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Some Page</title>

    <style type="text/css">
      div.footnote {
        font-size:9pt;
        line-height:12pt;
        text-align:center;}
    </style>
  </head>

  <body>
    ...
    <div class="footnote">
      Copyright 2006 Acme Products, Inc.
    </div>
  </body>
</html>
```

---

在这段代码中，div.footnote 样式类是在一个内部样式表中指定的，后者位于页面的开头。这样，该样式类可用于这个页面的正文。实际上，在页面正文中，它用于格式化版权标记。

创建只在一个页面中使用多次的样式规则时，内部样式表很方便。然而，在有些情况下，可能需要将样式应用于特定的元素。这称为内联（inline）样式规则，让你能够为页面的一小部分，如单个元素指定样式。例如，可以在<p>、<div>或<span>标签内使用 style 属性创建并应用样式规则。这种样式称为内联样式，因为是在 HTML 代码中指定的。

**注意：** <span>和</span>实际上是伪标签（dummy tag），本身并没有什么功能，只是指定添加的 style 属性将应用的范围。标签<div>和<span>之间唯一的区别是，<div>是块元素，包含分行，而<span>不包含。因此，要修改不分行的句子或段落中的文本样式，必须使用<span>。

**By the Way**

下面是一个 style 属性的例子：

```
<p style="color:green">
  This text is green, but <span style="color:red">this text is red.</span>
  Back to green again, but...
</p>
<p>
  ...now the green is over, and we're back to the default for this page.
</p>
```

这段代码使用<span>标签来说明如何在内联样式规则中应用颜色样式属性。实际上，这个例子中的<p>标签和<span>标签都将颜色属性用作内联样式。很重要的一点是，color:red 样式属性将覆盖标签<span>和</span>之间的文本的 color:green 样式。在第二段，不使用这两个 color 属性，因为这是一个新段落，使用的是整个页面的默认颜色。

**注意：** 为使 Sams Publishing 网站上所有网页的风格一致，我创建了一个样式表，网站的每个页面都使用该样式表。为读者方便并启发读者，我将这个样式表复制到 HTML 页面中，从同一个 zip 文件可以获得它。

**By the Way**

## 12.6 样式表快速复习

你已经理解了如何将 CSS 样式应用于网页，下面是三种主要的方法。

- 外部样式表：将一组 CSS 样式规则保存为一个独立的.css 文件，在每个页面开头通过<link />标签指定该文档。
- 内部样式表：通过<style>标签将一组 CSS 样式规则放在文档中。这些样式规则只对其所在的页面有效。
- 内联样式表：通过 style 属性直接将一个或多个 CSS 样式规则放在标签内。这些样式规则只对其所在的标签有效。

可以看到，如何应用 CSS 样式规则有很大的灵活性。如果要创建包含多个网页并需要确保风格类似的网站，强烈建议使用外部样式表；如果要在单个网页中组织并重用相同的样式，则内部样式表是不错的选择；最后，如果要对页面中特定的元素应用独特的样式，使用内联



样式表将很方便。

## 12.7 总结

本章读者学习了样式表可同时控制多个 HTML 网页的外观，它还能够极其精确地控制 HTML 元素的字体、间距和位置。读者还了解到，几乎所有 HTML 标签都可以加入 style 属性，从而控制 HTML 网页的任何部分，而不需要引用独立的样式表文档。

表 12.1 总结了本章讨论的标签。关于可放在<style>标签或 style 属性后的选项的具体细节，请参阅 [www.w3c.org](http://www.w3c.org) 中的 CSS1 和 CSS2 样式标准。

表 12.1 第 12 章介绍的 HTML 标签与属性

标签/属性	功能
<style>...</style>	使得可以在文档中包含内部样式表，用于<head>和</head>之间
属性	
type="内容类型"	Internet 内容类型，对于 CSS 样式表，总是 text/css
<link />	链接外部样式表（或其他文档类型），用于文档的<head>部分
属性	
href="url"	样式表的地址
type="内容类型"	Internet 内容类型，对于 CSS 样式表，总是 text/css
rel="样式表"	链接类型，对于样式表，总是 stylesheet
<span>...</span>	除提供一个放置样式或其他属性的位置外，不起什么作用。类似于<div>...</div>，但不产生分行
属性	
style="style"	包含内联样式说明，可用于<span>、<div>、<body>和其他大部分 HTML 标签中

## 12.8 问与答

问：如果网页链接了一个样式表，后者指定所有文本为蓝色，但在网页的某处有一个<span style="font-color:red">标签，则这个标签内的文本将是蓝色还是红色？

答：红色。局部内联样式总是优先于外部样式表。放在网页开头的<style>与</style>之间的样式也优先于外部样式表（但不优先于网页后面的内联样式）。这是我在本章开头提到过的级联效果。可以将级联效果看为首先从外部样式表开始，然后被内部样式表覆盖，再被内联样式表覆盖。

问：一个网页可以链接多个样式表吗？

答：当然可以。例如，可使用一个样式表指定字体，另一个样式表指定边距和间距——只要分别用一个<link />包含这两个样式表。从技术上说，CSS 标准要求网页浏览器允许用户在多个<link />标签提供的多个样式表中进行选择。然而，实际上所有主流网络浏览器只是包含每一个样式表。实际上，链接多个样式表更常使用@import 命令。下面是一个通过@import 导入多个样式表的例子：

```
@import url (styles1.css) ;
@import url (styles2.css) ;
```

与<link />标签一样，@import 命令必须放在网页的开头。你将在第 15 章进一步学习这个方便的小命令，那时你将学习如何创建专门用于打印网页的样式表。

## 12.9 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 12.9.1 测验

1. 创建一个样式表，指定标题使用 30 点的 Arial 字体，颜色为蓝色；其他所有文本使用 10 点的 Times Roman 字体（或默认的浏览器字体），颜色为蓝色，行间距为两倍行间距。
2. 假设在问题 1 中创建的样式表为 corporate.css，如何将它应用于名为 intro.html 的网页？

### 12.9.2 答案

1.

```
h1 { font:30pt blue Arial; }  
body { font: 10pt blue; }
```

2. 将如下的标签放在 intro.html 文档的<head>与</head>标签之间：

```
<link rel="stylesheet" type="text/css" href="corporate.css" />
```

## 12.10 练习

1. 为你的网站开发一个标准样式表，并在所有网页中链接该样式表（对于不需要使用这个样式表的网页，使用内部样式表或内联样式）。如果为某家公司工作，很可能为打印材料指定字体和样式，按这些规范设置公司的网页。

2. 研究 <http://www.w3.org/Style/CSS/> 上关于样式表的官方规范，尝试使用一些本章没有提及的高级样式属性。

## 第 13 章

# 深入探讨样式表格式化

---

前一章介绍了样式表的基础知识，演示了如何运用样式表。本章将深入探讨前一章没有涉及的一些 CSS 高级主题，这些知识对创建网页很有帮助。CSS 是一种非常强大的技术，让网页设计人员能够更容易设计具有良好视觉效果网页。读者在本章学习的技巧不仅可增加 CSS 知识，还可让你更好地理解如何将样式表用于实际网页中。

### 13.1 使用 CSS 格式化文本的高级技巧

你已经学习了如何使用几个控制网页文本格式的 CSS 样式属性。除这些常用文本格式化属性外，还有些不那么常用的属性也值得学习，这可增长读者的 CSS 知识。

首先是 `font-variant` 样式属性，用于将 `small-caps`（小型大写字母）效果应用于文本。这种文本效果使段落中的所有文本都使用大写字母；应用 `small-caps` 时首字母比其他字母要大。

下面是一个使用 `font-variant` 样式属性的例子：

```
<p style="font-variant:small-caps">  
  This paragraph is in small caps.  
</p>
```

`font-variant` 属性可能取值包括 `none`（默认值）和 `small-caps`。图 13.1 说明了 `small-caps` 文本在网页中如何显示。

另外两个有趣的文本格式化属性是 `letter-spacing` 和 `word-spacing`，这两个属性可用于修改字母间距和词间距。这两个属性都使用标准的 CSS 度量单位，如像素（`px`）。

下面的例子演示了如何使用样式属性 `letter-spacing` 在段落的字母之间增加五个像素：

```
<p style="letter-spacing:5px">  
  This paragraph has its letters spaced apart by 5 pixels.  
</p>
```

如果要控制词间距而不是字母间距,可参见下面使用 word-spacing 在单词间添加 10 像素的例子:

```
<p style="word-spacing:10px">
  This paragraph has its words spaced apart by 10 pixels.
</p>
```

这两个例子的结果如图 13.1 所示。

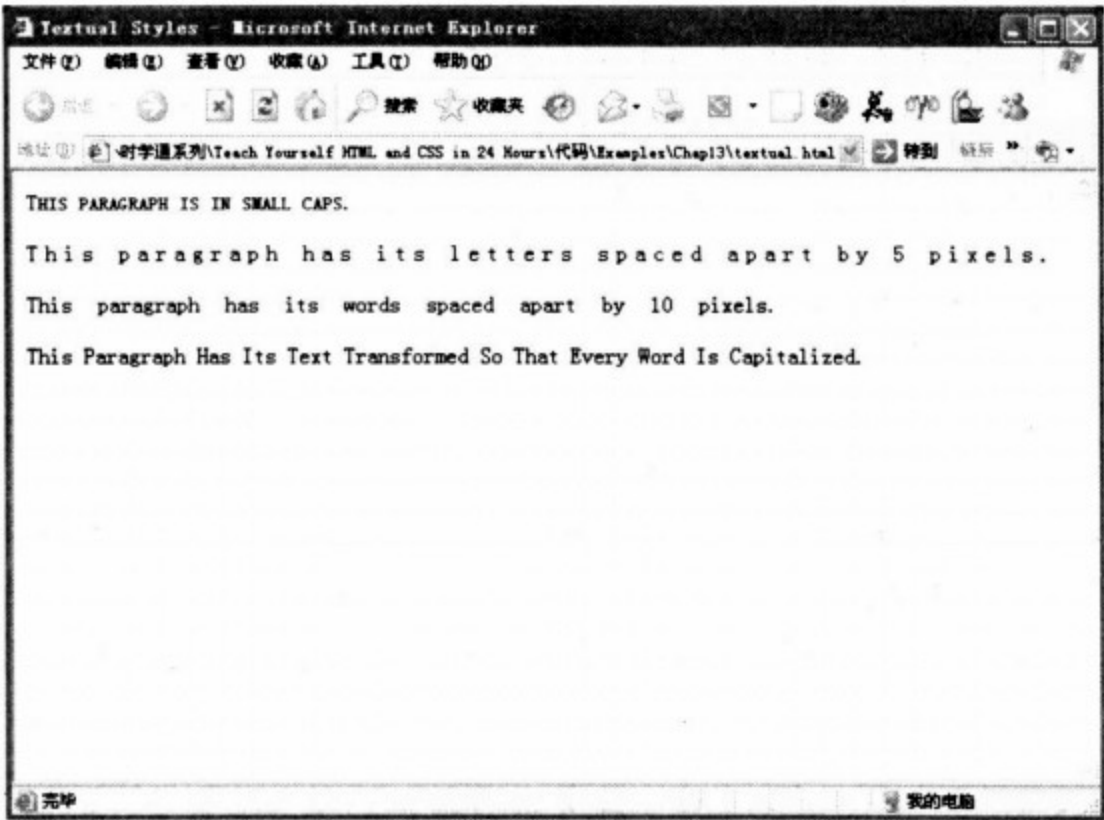


图 13.1  
高级的 CSS 样式属性可修改文本的大小写和间距

这里要介绍的最后一个文本格式化样式属性不像前几个那么常用;然而,这个样式属性在适用的情况下也很有用。这就是 text-transform 属性,用于自动转换段落文本的大小写。该属性的可能取值包括 none (默认值)、uppercase (大写)、lowercase (小写) 和 capitalize (首字母大写)。uppercase 和 lowercase 这两个值分别表示所有文本大写和小写,而 capitalize 表示每个单词的首字母大写。

下面的例子演示了如何使用 text-transform 属性来转换段落文本,使每个单词的首字母大写:

```
<p style="text-transform:capitalize">
  This paragraph has its text transformed so that every word is capitalized.
</p>
```

图 13.1 也显示这个例子的结果。

程序清单 13.1 列出了图 13.1 所示的高级文本格式化页面的完整代码。

程序清单 13.1 CSS 可以全面地控制文本的格式化

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

```

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Textual Styles</title>
  </head>

  <body>
    <p style="font-variant:small-caps">
      This paragraph is in small caps.
    </p>
    <p style="letter-spacing:5px">
      This paragraph has its letters spaced apart by 5 pixels.
    </p>
    <p style="word-spacing:10px">
      This paragraph has its words spaced apart by 10 pixels.
    </p>
    <p style="text-transform:capitalize">
      This paragraph has its text transformed so that every word is capitalized.
    </p>
  </body>
</html>

```

请记住，可以混合使用这些样式属性，也可以将其同本书介绍的其他与字体或文本相关的属性搭配使用。实际上，在 13.5 节中将介绍如何混合使用这些文本格式化属性，实现有趣的文本效果。

### By the Way

**注意：**CSS 标准定义的样式属性比现在可以使用的属性多得多。例如，从技术上说，font-stretch 和 text-shadow 属性属于标准的一部分，但在主要的网页浏览器中都没有得到支持。font-stretch 属性将文本往一个方向拉伸，而 text-shadow 属性添加投影——只需应用样式，而不需要复杂的图像编辑！

## 13.2 修改链接的外观

读者肯定见过有些链接的响应方式与默认行为不同，链接默认地显示为带下划线的蓝色文本（在大多数网页浏览器中）。在链接的网页或资源被打开后，大多数浏览器将链接改为紫色，以表明用户是否单击过链接。

CSS 包含一种根据链接状态修改链接外观的机制。为应用样式，可将链接的状态分为如下四种。

- Link: 链接尚未访问过。
- Hover: 鼠标指向链接。
- Active: 链接被激活，如用鼠标单击。
- Visited: 链接被访问过。

每种链接状态对应一个 CSS 伪类（pseudoclass），伪类是应用于元素状态的一种特殊样



式类。伪类的独特之处在于，指定它们要使用冒号 (:) 而不是点号 (.)。应用于链接的伪类包括 link、hover、active 和 visited。要应用这些伪类样式，需将其附加在元素后，如 a:link、a:hover、a:active 和 a:visited。

下面的例子说明了如何设置 a:link 伪类的样式，它决定了正常链接的外观：

```
a:link {
  color:#19619A;
  font-weight:bold;
  text-decoration:none;
}
```

可以看到，通过 text-decoration:none 删除了链接的下划线，而 font-weight:bold 将链接的字体设置为粗体。#19619A 表示淡蓝色。这个样式规则稍后将用于新版本的曲棍球队员样式表中。

**休息一下：**有关如何应用链接样式以获得不同的链接效果的示例，可参考 Wired News 网站 <http://www.wired.com/>。Wired News 网站将链接显示为带下划线的文本，但在用于将鼠标指向链接时，链接为红色而不是蓝色。访问过的链接显示为紫色。

**Coffee**

下面的代码演示了如何修改前一个链接样式，使鼠标指向时链接的颜色不同：

```
a:hover {
  background-color:gold;
  font-weight:bold;
  text-decoration:none;
}
```

也可以修改链接活动时及访问后的外观。然而，如果只要修改鼠标指向链接的外观，只要将样式类 a:active 和 a:visited 设置为与 a:link 类匹配。

**提示：**这里有一个技巧，如果 a:visited 的颜色与页面背景色相同，已访问过的链接将不可见。这给网页提供了“智能”——只提供用户没有访问过的链接。另一方面，有些用户可能不这样认为，因为这样不知道链接到哪里去了。因此，使用这个技巧要谨慎。

**Did you**

## 13.3 CSS 代码注释

与 HTML 代码类似，CSS 代码也允许包含对样式表的描述。然而，与 HTML 不同的是，CSS 注释使用不同的方式。具体地说，CSS 注释放在 /\* 和 \*/ 之间。因此，下面是一条简单的 CSS 注释：

```
/* This is a CSS comment. */
```

CSS 注释必须以 /\* 开始，以 \*/ 结束。另外，CSS 注释只能用于样式表（内部样式表或外

部样式表), 而不能用于内联样式。

**By the Way**

**注意:** 如果使用过编程语言 C、C++ 或 Java, 应该很熟悉 CSS 注释格式 /\* \*/——CSS 从这些语言中借鉴了注释语法。

程序清单 13.2 对前一章的 Waldorf School 网页的样式表进行了修改。该样式表版本与前一个版本类似, 只是增加了一些注释, 对样式的含义进行解释。

#### 程序清单 13.2 CSS 注释可用于解释样式规则的含义

```
/* Make all body text maroon-colored 12-point Book Antiqua
   with 16-point vertical spacing between lines of text
   and 10-point margins. Use parchment.gif as the background. */
body {
    font-size:12pt;
    font-family:Book Antiqua;
    color:maroon;
    line-height:16pt;
    margin-left:10pt;
    margin-right:10pt;
    background:url(parchment.gif);
}

/* Indent paragraphs */
p {
    margin-left:24pt;
    margin-right:24pt;
}

/* Make headings Prose Antique bold with generous line spacing.
   If user doesn't have Prose Antique, use Lucida Handwriting. */
h1 {
    font:24pt Prose Antique, Lucida Handwriting;
    font-weight:bold;
    line-height:30pt;
}

h2 {
    font:18pt Prose Antique, Lucida Handwriting;
    font-weight:bold;
    line-height:22pt;
}

/* Don't underline links, and make all links red.
   Make links flash black when activated. */
a {
    text-decoration:none;
}
```

```

a:link {
    color:red;
}

a:visited {
    color:red;
}

a:active {
    color:black;
}

/* Format footnotes as 9-point Book Antiqua, and center them. */
div.footnote {
    font-size:9pt;
    line-height:12pt;
    text-align:center
}

```

这段代码演示了注释在 CSS 样式表中的用途，以及如何使用注释更详细地描述每个样式规则的意义。

## 13.4 CSS 技巧

虽然本章已介绍新样式表的一些有趣的地方，但还有一些独特的 CSS 特性没有介绍。接下来的几个小节将介绍一些非常方便的 CSS 技巧。

### 13.4.1 使用通用类

到目前为止，本书每个 CSS 样式规则都依赖于基标签指定规则的选择符部分。例如，要创建一段粗体文本，使用如下所示的样式规则：

```
p.bold { font-weight:bold; }
```

在这个例子中，在规则中的点号左边使用了元素 `p`。要创建可应用于任何元素的样式规则，可省略选择符左边的内容，从而创建一个通用样式类。通用样式类可应用于支持样式规则包含的属性的任何元素。下面的例子创建一个通用 `bold` 样式类：

```
.bold { font-weight:bold; }
```

这个通用样式类与前一个样式类相似，只是没有使用元素 `p`。这意味着现在可将该样式类应用于任何支持 `font-weight` 属性的元素。下面是一个例子：

```

<p class="bold">
  This paragraph is bold.
</p>
<p>

```

```
This <span class="bold">word</span> is bold.
</p>
```

在这个例子中，第一段将显示为粗体，因为 `class` 属性被设置为 `bold`。然而，第二段除 `<span>` 标签外的文本都将显示为常规字体，`<span>` 标签的 `class` 属性也被设置为 `bold`。因为 `bold` 是通用样式类，因此可用于 `<p>`、`<span>` 和其他标签。

### 13.4.2 指定多个选择符

创建样式规则时可使用的另一个技巧是，为单个规则指定多个选择符。这样可消除不必要的重复代码，使样式表代码更紧凑。请看下面的例子：

```
a:link, a:visited, a:active {
  color:#19619A;
  font-weight:bold;
  text-decoration:none;
}
```

在这个例子中，`a:link`、`a:visited` 和 `a:active` 这三个样式类都设置为相同的样式规则。如果不使用多个选择符，必须将该样式规则重复三次，这显然是不必要的麻烦。

下面是另一个例子，旨在确保读者理解如何使用多个选择符：

```
h1, h2 { text-align:center; font-style:italic; }
```

在这行代码中，元素 `h1` 和 `h2` 使用同一种样式，使其文本都居中对齐，字体为斜体。通过使用多个选择符，可简化样式表代码，使其更易于维护。换句话说，如果要修改元素 `h1` 和 `h2` 的样式规则，只要编辑一个规则即可。

### 13.4.3 混合样式类

最后一个重要的样式表技巧是使用混合样式表。具体地说，可以创建应用于特定元素的自定义样式表，这种技巧也可用于伪类。前面讲过，伪类是与元素状态相关联的内置类，如 `a:link` 和 `a:hover`。有时候网页可能需要创建几种不同的链接，在这种情况下可以创建多个链接样式类。

要理解如何使用混合样式类，请看下面的代码：

```
a:link, a:visited, a:active {
  color:#19619A;
  font-weight:bold;
  text-decoration:none;
}

a:hover {
  background-color:gold;
}
```

```
font-weight:bold;
text-decoration:none;
}
```

读者应该熟悉这段代码，因为它包含前面介绍过的一些链接样式。可以看到，这些样式类使链接以粗体显示且没有下划线。如果要使用类似的链接，但使用斜体而不是粗体，该如何办呢？可使用下面的代码：

```
a.emphasis:link, a.emphasis:visited, a.emphasis:active {
    color:#19619A;
    font-style:italic;
    text-decoration:none;
}

a.emphasis:hover {
    background-color:gold;
    font-style:italic;
    text-decoration:none;
}
```

设置为 `emphasis` 类的链接将显示为斜体而不是粗体。

## 13.5 重新设计曲棍球队员示例

本章介绍了 CSS 样式表的一些有趣的地方，现在要将其中的一些新知识应用到实际示例中。程序清单 13.3 列出了曲棍球队员网页的 HTML 代码，但使用一个外部样式表对网页内容进行格式化。

程序清单 13.3 曲棍球队员网页现在使用外部样式表

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Music City Mafia - Terry Lancaster</title>
    <link rel="stylesheet" type="text/css" href="player.css" />
    <script type="text/javascript" src="external.js"></script>
  </head>

  <body>
    <div class="title">16 - Terry Lancaster</div>
    <div>
      <div class="image">
        
      </div>
```



```

<div class="info">
  <div>
    <span class="label">Nickname:</span> Big T<br />
    <span class="label">Position:</span> RW<br />
    <span class="label">Height:</span> 6' 3"<br />
    <span class="label">Weight:</span> 195<br />
    <span class="label">Shoots:</span> Left<br />
    <span class="label">Age:</span> 40<br />
    <span class="label">Birthplace:</span>
    <a href="http://en.wikipedia.org/wiki/Nashville%2C_Tennessee">Nashville,
    TN</a>
  </div>
</div>
<div class="favorites">
  <div>
    <span class="label">Favorite NHL Player:</span>
    <a href="http://www.nhl.com/players/8448091.html">Brett Hull</a><br />
    <span class="label">Favorite NHL Team:</span>
    <a href="http://www.nashvillepredators.com/">Nashville Predators</a>
    <br />
    <span class="label">Favorite Southern Fixin:</span>
    <a href="http://southernfood.about.com/od/potatorecipes/r/blbb442.htm">
    Skillet Fried Potatoes</a><br />
    <span class="label">Favorite Meat and Three:</span>
    <a href="http://www.hollyeats.com/Swetts.htm">Swett' s</a>
    (<a href="http://maps.google.com/maps?q=2725+clifton+ave,+nashville,+tn"
    rel="external">map</a>)
    <br />
    <span class="label">Favorite Country Star:</span>
    <a href="http://www.patsycline.com/">Patsy Cline</a><br />
    <span class="label">Favorite Mafia Moment:</span>
    "<a href="mcmplayer_chale.html">Chet</a> finishing the game with his
    eyelid completely slashed through."
  </div>
</div>
</div>
<div>
  <table class="stats" border="1">
    <tr class="heading">
      <th class="season">Season</th>
      <th>GP</th>
      <th>G</th>
      <th>A</th>
      <th>P</th>
      <th>PIM</th>
      <th>PPG</th>
      <th>SHG</th>
      <th>GWG</th>
    </tr>
  </table>
</div>

```

```
</tr>
<tr class="light">
  <td class="season">Summer 2005</td>
  <td>11</td>
  <td>7</td>
  <td>5</td>
  <td>12</td>
  <td>0</td>
  <td>0</td>
  <td>0</td>
  <td>0</td>
</tr>
<tr class="dark">
  <td class="season">Winter 2004</td>
  <td>24</td>
  <td>14</td>
  <td>14</td>
  <td>28</td>
  <td>2</td>
  <td>0</td>
  <td>0</td>
  <td>5</td>
</tr>
<tr class="light">
  <td class="season">Summer 2004</td>
  <td>18</td>
  <td>9</td>
  <td>9</td>
  <td>18</td>
  <td>2</td>
  <td>0</td>
  <td>0</td>
  <td>2</td>
</tr>
<tr class="dark">
  <td class="season">Spring 2004</td>
  <td>19</td>
  <td>7</td>
  <td>17</td>
  <td>24</td>
  <td>0</td>
  <td>0</td>
  <td>0</td>
  <td>1</td>
</tr>
</table>
</div>
<div class="contact">
```

```
<a href="mailto:l&#97;ncastert@musiccitym&#97;fi&#97;.com?subject=
Fan Question&body=What's your secret?">Contact Terry.</a>
</div>
</body>
</html>
```

### By the Way

**注意：**external.js 是程序清单 13.3 引用的 JavaScript 文件，在第 3 章讨论如何使用 JavaScript 让目标链接打开新浏览器窗口时介绍过。要使网页是有效的 XHTML 文档，必须使用 JavaScript 代码，因为 XHTML 不支持使用 target="\_blank" 在新窗口中打开链接。

读者应该熟悉这段代码的大部分，因为它们与在本书中出现过多次的代码类似。这段代码甚至比前几个曲棍球队员网页版本还简单，因为它删除了所有的 CSS 内联样式。这是因为现在这个网页使用了样式类，而样式类是在外部样式表中定义的。

曲棍球队员网页使用的外部样式表名为 player.css，其代码如程序清单 13.4 所示。

**程序清单 13.4** 外部样式表 player.css 包含几个样式类，对曲棍球队员网页的内容进行详细格式化

```
body {
    font-family: Verdana, Arial;
    font-size: 12pt;
    color: navy;
}

div {
    padding: 3px;
}

div.title {
    font-size: 18pt;
    font-weight: bold;
    font-variant: small-caps;
    letter-spacing: 2px;
    color: black;
}

div.image {
    float: left;
}

div.info, div.favorites {
    float: left;
    vertical-align: top;
}
```

```
table.stats {
  width:100%;
  clear:both;
  text-align:right;
  font-size:11pt;
}

.label {
  font-weight:bold;
  font-variant:small-caps;
}

tr.heading {
  font-variant:small-caps;
  background-color:navy;
  color:white;
}

tr.light {
  background-color:white;
}

tr.dark {
  background-color:#EEEEEE;
}

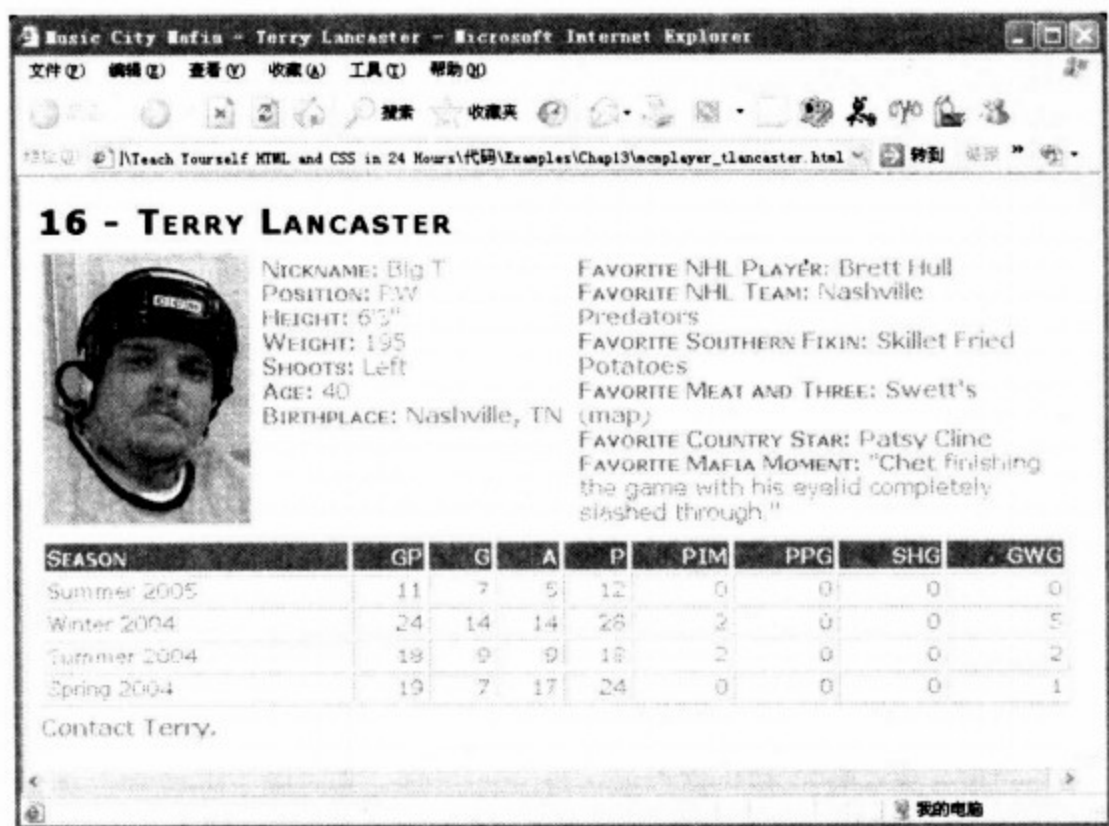
th.season, td.season {
  text-align:left;
}

a:link, a:visited, a:active {
  color:#19619A;
  font-weight:bold;
  text-decoration:none;
}

a:hover {
  background-color:gold;
  font-weight:bold;
  text-decoration:none;
}
```

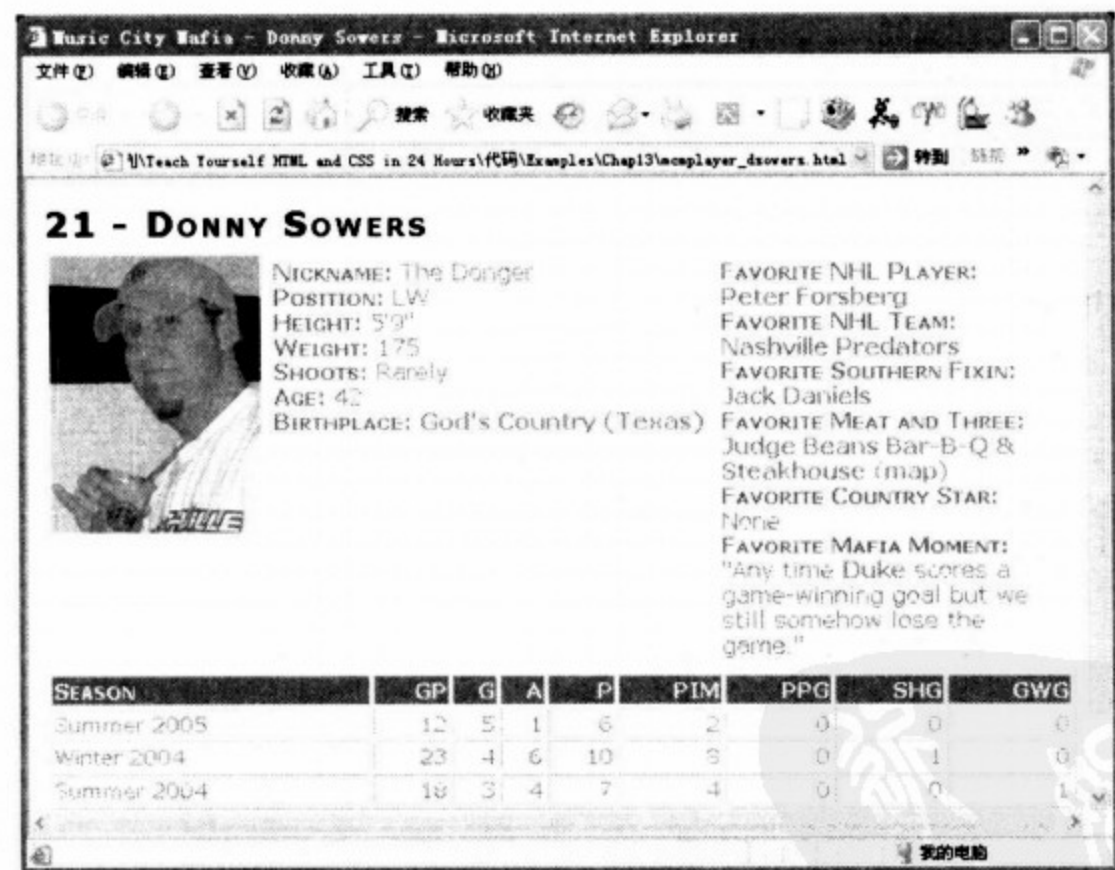
这个样式表旨在将曲棍球队员网页各个部分的信息分离并分别指定格式。另外，样式要一致，可应用于多个队员的网页，使其风格一致。图 13.2 显示了队员 Terry Lancaster 的网页，这个网页的格式全部在外部样式表 `player.css` 中指定。

图 13.2  
曲棍球队员网页的  
样式全部由外部样  
式表指定



如果不将 player.css 应用于其他网页，然后查看结果，就不能体现外部样式表的强大功能。图 13.3 显示了使用同一个样式表的另一个队员的网页。

图 13.3  
由于使用外部样式  
表 player.css，该  
曲棍球队员网页的  
样式与图 13.2 相同



这个例子说明了应用于多个网页时外部样式表的强大功能。不仅如此，如果认真研究样式表 player.css，将发现它用了本章介绍的许多 CSS 高级技巧。

### 13.6 总结

本章接着前一章没有介绍的地方，继续探索样式表的功能。首先介绍了一些读者未见过



的新文本格式化样式属性；然后介绍了如何使用 CSS 伪类修改链接的风格；接下来介绍 CSS 注释，读者了解了如何在 CSS 代码中添加注释。然后介绍了一些 CSS 技巧，其中大部分对读者创建样式表都很有用。这一章最后将学到的大部分 CSS 知识应用到一个实际网页中。

## 13.7 问与答

问：我曾经看到一些样式表，其中的样式规则定义以#开始。这是什么？我可以在样式表中使用时吗？

答：以#开始的样式规则称为 ID 类，能够创建通过 id 属性应用于元素的样式规则。可以创建和使用 ID 类，但不鼓励这样做，因为这可能导致多个元素的 id 属性设置为同一个值，而这是 XHTML 不允许的。XHTML 要求同一网页中的每个 id 属性都是惟一的，也就是说，不可以在多个元素中使用同一个 ID 类。知道这些后，最好使用常规样式类，然后通过 class 属性将其应用于多个元素。

问：有时设置 a:hover 伪类后，我注意到当将鼠标指向链接时，浏览器中的网页会闪烁和抖动。这是为什么？

答：事实上，可以很容易地设置一个 a:hover 伪类，要求浏览器在鼠标指向时不断重画网页。发生这种情况的典型原因是链接改变了字体大小，而有时可能是将文本从常规变成粗体。因为链接的整体尺寸因鼠标指向它而改变，这意味着浏览器必须重新计算网页中在链接后的每个部分的位置，然后重画结果。解决的方法是不要在鼠标指向链接上时改变链接文本的大小。

## 13.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 13.8.1 测验

1. 如何创建显示为小型大写字母的标题？
2. 如何创建样式规则，使网页中所有链接都显示为红色——指向模式除外，在这个模式下链接变成绿色。
3. 如何在 CSS 代码中插入注释？
4. 通用样式类的作用是什么？

### 13.8.2 答案

1.

```
<h1 style="font-variant:small-caps">My Heading</h1>
```

2.

```
a:link, a:visited, a:active {  
    color:red;  
}
```

```
a:hover {  
    color:green;  
}
```

3. 只需将注释文本放在/\*和\*/之间。

4. 通用样式类可应用于任何元素，而不是特定类型的元素。要使用通用样式类，元素必须支持该样式类中的样式规则，除此之外，通用样式类没有其他限制。很容易识别样式表中的通用样式类，因为它们都以一个句点(.)打头。

## 13.9 练习

1. 修改曲棍球队员样式表 `player.css` 中的一些样式规则。特别注意这些修改如何使每一个队员网页变为不同的样子。

2. 使用本章介绍的一些 CSS 样式技巧创建样式表（内部或外部的）。例如，创建一些有趣的链接伪类，使网页中的链接闪烁。



## 第 14 章

# 将样式表用于页面布局

在前两章，读者学习了如何使用默认的 CSS 定位方法（称为相对定位）来指定 HTML 内容的位置。在相对定位中，网页元素按前后顺序显示，每个元素都显示在之前的元素之后。这是本书到目前为止介绍过的每个 HTML 文档定位元素的方法。尽管相对定位有一些优点，但样式表提供了另一种方法，让你能够对网页元素的位置进行更多的控制。

本章探讨样式表控制的 HTML 定位。读者将学习如何进行相对定位和绝对定位，以及如何使元素重叠、如何控制元素的重叠方式。还将学习如何调整网页元素之间的边距和间隔。到本章结束时，读者将认识到为什么现在表格作为网页布局机制已经过时——你将认识到样式表的功能太强大了，不能不用它。

### 14.1 定位

前面提到过，相对定位是 HTML 使用的默认定位类型。可以将相对定位理解为棋盘上棋子的排列：棋子从左到右排列，达到边缘时移到下一行。样式属性 `display` 的值为 `block` 的元素自动放在下一行，而为 `inline` 的元素自动放在同一行，位于前一个元素后面。举个例子，`<p>` 和 `<div>` 标签是块元素，而 `<span>` 标签是内联元素。

**注意：**相对定位是 CSS 使用的默认定位方法，因此，如果没有指定样式规则的定位，将默认为相对定位。

**By the Way**

CSS 还支持另一种定位，称为绝对定位，因为它可以精确地设置 HTML 内容在网页中的位置。虽然绝对定位让你能够明确指定元素的位置，但位置仍是相对于网页中父元素的。换句话说，绝对位置让你能够指定元素的矩形局域相对于父元素区域的位置，这一点与相对定位截然不同。

与现实生活中的大部分事情一样，自由往往伴随着责任。你拥有指定元素在网页中的位置的自由，但当元素占据了其他元素的位置时，这就导致元素重叠的问题。你可以指定元素的绝对位置，从而导致元素重叠。在这种情况下，CSS 依靠 `z` 索引（`z-index`）来决定哪个元

素在上方，哪个元素在下方。你将在本章稍后更详细地学习 z 索引。现在，先来看看如何控制样式规则使用相对定位还是绝对定位。

样式规则使用的定位类型（相对或绝对）是由 `position` 属性决定的，该属性有两个可能的值：`relative` 和 `absolute`。指定定位类型后，可用下列属性来提供具体的位置。

- `left`: 左边的位置偏移。
- `right`: 右边的位置偏移。
- `top`: 顶部的位置偏移。
- `bottom`: 底部的位置偏移。

你可能认为这些位置属性只适用于绝对位置，但实际上它们对两种定位都适用。在相对定位下，元素的位置用相对于元素原始位置的偏移来指定。因此，如果将 `left` 属性设置为 `25px`（像素），元素的左边将从原来的（相对）位置向右移 25 像素。而对于绝对定位，应用样式的元素位置是相对于父元素指定的。因此，如果在绝对定位下将 `left` 属性设置为 `25px`，元素的左边将距离父元素的左边 25 像素。而如果将 `right` 属性设置为同样的值，元素的右边将距离父元素的右边 25 像素。

看一个例子可以帮助理解定位。请看程序清单 14.1 中一个简单网页的代码。

**程序清单 14.1 一个非常简单的网页，依靠外部样式表进行网页布局**

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Color Blocks</title>
    <link rel="stylesheet" type="text/css" href="colors_rel.css" />
  </head>

  <body>
    <div class="one">One</div>
    <div class="two">Two</div>
    <div class="three">Three</div>
    <div class="four">Four</div>
  </body>
</html>
```

---

这个网页并不是很有趣，但很好地说明了相对定位和绝对定位之间的区别。注意，代码中有几个 `div` 元素，每个 `div` 元素都使用不同的样式类，因此使用不同的样式规则。在将样式表应用于这个网页前，先看看图 14.1 显示的结果。

我知道，这是一个非常普通的网页；但它很好地说明了样式表对单调的内容可有多大的影响，下面你将看到这一点。程序清单 14.2 包含了用于这个网页的样式表，它使用相对定位将网页的内容放在彩色矩形中。



图 14.1  
如果不使用样式表，Color Blocks 网页将非常单调

程序清单 14.2 colors\_rel.css 样式表使用相对定位

```
div {
    position:relative;
    width:250px;
    height:100px;
    border:10px single black;
    color:black;
    text-align:center;
}

div.one {
    background-color:red;
}

div.two {
    background-color:green;
}

div.three {
    background-color:blue;
}

div.four {
    background-color:yellow;
}
```

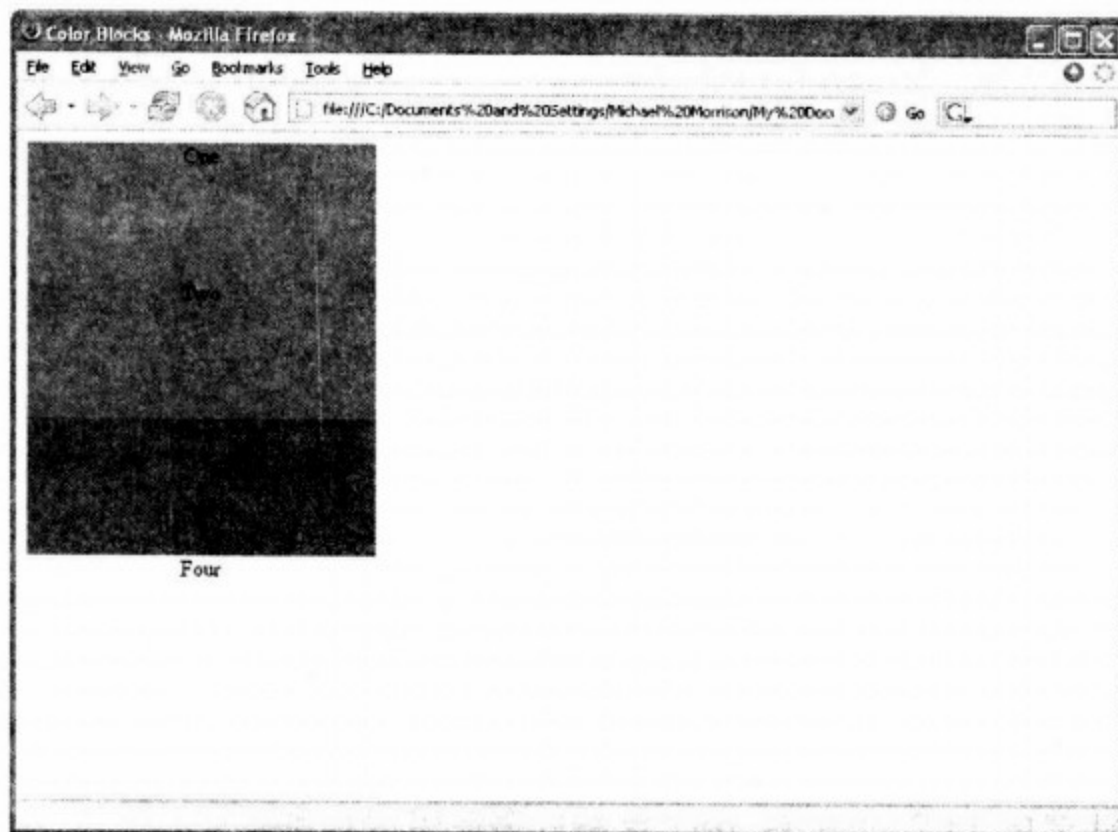
这段代码首先将 div 元素的样式属性 postion 设置为 relative。因为其他样式规则都继承 div 样式规则，因此它们也都继承了它的相对定位。实际上，其他样式规则惟一的不同



在于背景颜色。图 14.2 显示了 Color Blocks 网页使用相对定位样式表时显示在 Firefox 中的样子。

图 14.2

采用一个使用相对定位的样式表显示 Color Blocks 网页



注意，上图中 div 元素是按顺序显示的，因为使用了相对定位。要使网页的显示更有趣，可以改为使用绝对定位，显式地指定元素的位置。程序清单 14.3 包含了用于 Color Blocks 网页的使用绝对定位来排列元素的样式表。

#### 程序清单 14.3 样式表 colors\_abs.css 使用绝对定位

```
div {
    position: absolute;
    width: 250px;
    height: 100px;
    border: 10px solid black;
    color: black;
    text-align: center;
}

div.one {
    background-color: red;
    left: 0px;
    top: 0px;
}

div.two {
    background-color: green;
    left: 75px;
    top: 25px;
}
```

```
div.three {
  background-color:blue;
  left:150px;
  top:50px;
}

div.four {
  background-color:yellow;
  left:225px;
  top:75px;
}
```

这个样式表将 `position` 属性设置为 `absolute`，要使样式表使用绝对定位必须这样设置。另外，每个派生的 `div` 样式规则都设置了 `left` 和 `top` 属性。然而，这些规则的位置都设置成使元素互相重叠，如图 14.3 所示。

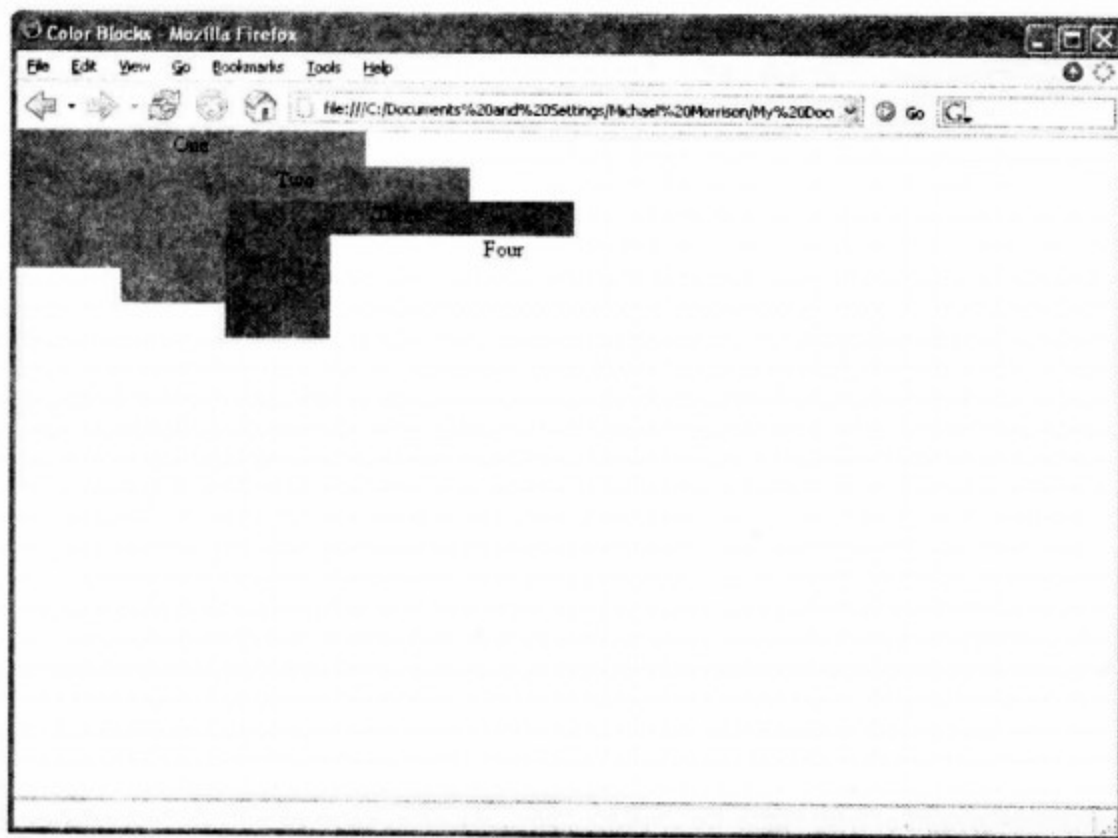


图 14.3

引用的样式表使用绝对定位时，Color Blocks 网页的显示结果

该图表明，绝对定位让你能够准确地指定元素的放置位置。它还表明排列元素使其相互重叠很简单。读者可能想知道元素重叠时浏览器如何知道哪个元素在上面，请往下阅读以找到答案。

## 14.2 控制元素的堆叠顺序

在有些情况下，肯定需要仔细地控制元素在网页中堆叠顺序。`z-index` 样式属性让你能够设置元素的堆叠顺序。`z-index` 这个名称看上去有些奇怪，它表示除计算机屏幕上水平和垂直的两维外，指向计算机屏幕的第三维。也可以将 `z-index` 看作是一本杂志在一摞杂志中的相对位置。靠近顶部的杂志的 `z-index` 比下面的杂志大；同样，堆叠元素时，`z-index` 值较大的元素在 `z-index` 值较小的元素上面。

z-index 属性用于设置一个数值，指示样式规则的相对 z-index。z-index 值仅在与样式表中的其他样式规则比较时才有意义，即设置单个样式规则的 z-index 属性没有意义。另一方面，如果为应用于重叠元素的多个样式规则设置 z-index，则 z-index 较大的元素将显示 z-index 较小的元素上面。

**By the Way**

**注意：**不管样式规则设置的 z-index 是多少，应用该样式规则的元素都将显示在父元素上面。

程序清单 14.4 列出了 Color Blocks 网页引用的样式表的另一个版本，它设置 z-index 来改变元素的堆叠顺序。

程序清单 14.4 样式表 colors\_zcss 改变 Color Blocks 网页中元素的 z-index

```
div {
    position: absolute;
    width: 250px;
    height: 100px;
    border: 10px solid black;
    color: black;
    text-align: center;
}

div.one {
    background-color: red;
    z-index: 0;
    left: 0px;
    top: 0px;
}

div.two {
    background-color: green;
    z-index: 3;
    left: 75px;
    top: 25px;
}

div.three {
    background-color: blue;
    z-index: 2;
    left: 150px;
    top: 50px;
}

div.four {
    background-color: yellow;
    z-index: 1;
    left: 225px;
```

```
top:75px;  
}
```

与程序清单 14.3 中的代码相比，这段代码惟一的不同是，每个带编号的 div 样式类增加了 z-index 属性。注意，第一个 div 的 z-index 设置为 0，这使它成为最下层的元素，而第二个 div 的 z-index 最大。图 14.4 说明了 z-index 如何影响网页的显示。

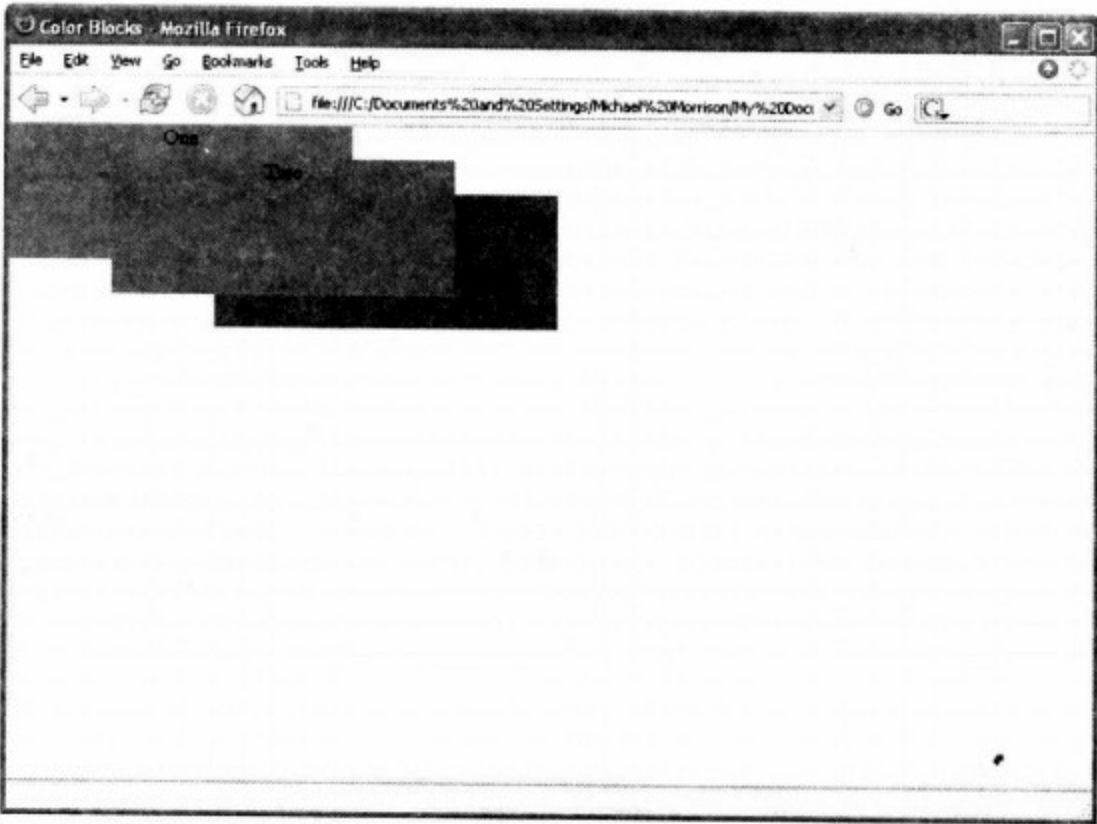


图 14.4  
使用修改了 z-index 的样式表时 Color Blocks 网页的显示结果

该图表明，z-index 样式属性可以准确地控制元素的堆叠顺序。

为说明 z-index 样式属性可以影响任何 HTML 内容，请看图 14.5。由于使用了绝对定位和 z-index 属性，该网页中的图像重叠得以堆叠。

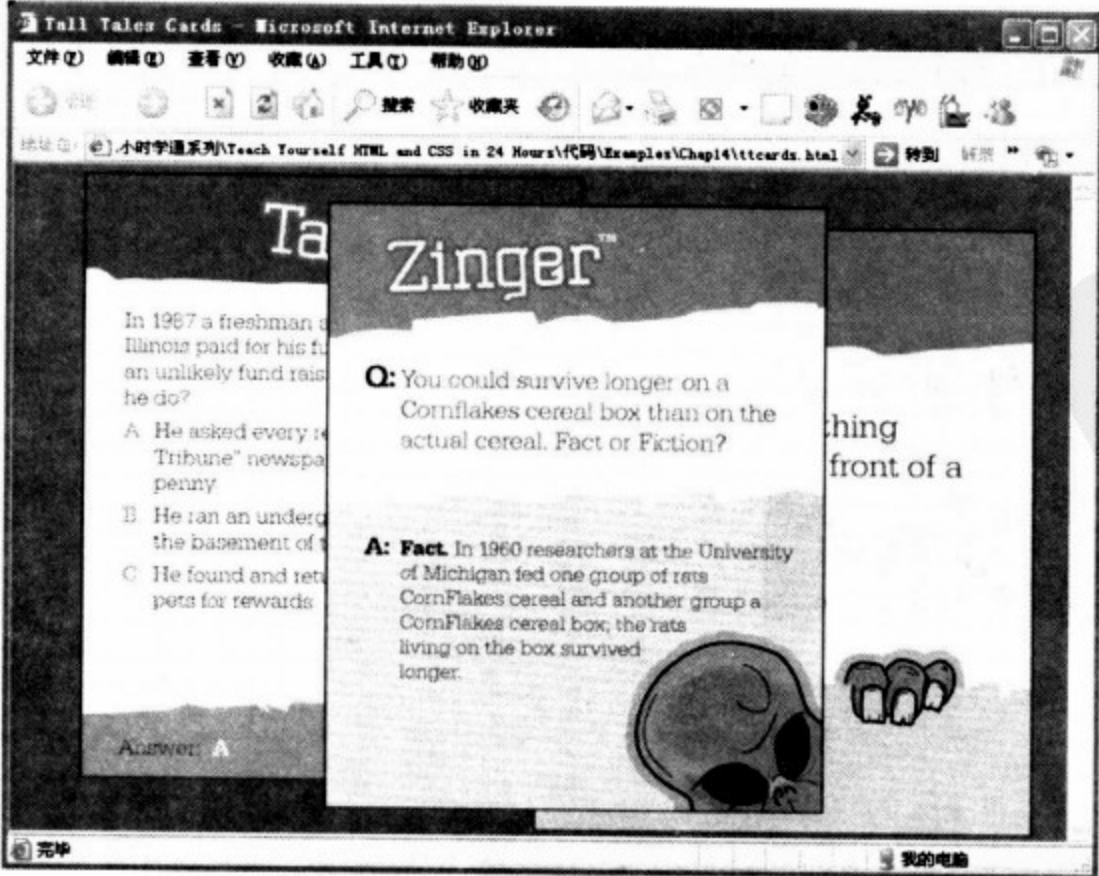


图 14.5  
对包括图像在内的块元素，可使用绝对定位并设置 z-index

该图表明，网页中的任何内容都可以重叠。

### 14.3 使用边距 (margin)

在本书网页的左边和右边，可以看到文本段落和页面边缘之间有一定的空白，这就像书中的边距一样。样式表边距让你能够网页中元素矩形区域周围添加空白。下面是设置边距的样式属性。

- `margin-top`: 设置上边距。
- `margin-right`: 设置右边距。
- `margin-bottom`: 设置下边距。
- `margin-left`: 设置左边距。
- `margin`: 在一个属性中设置上、下、左、右边距。

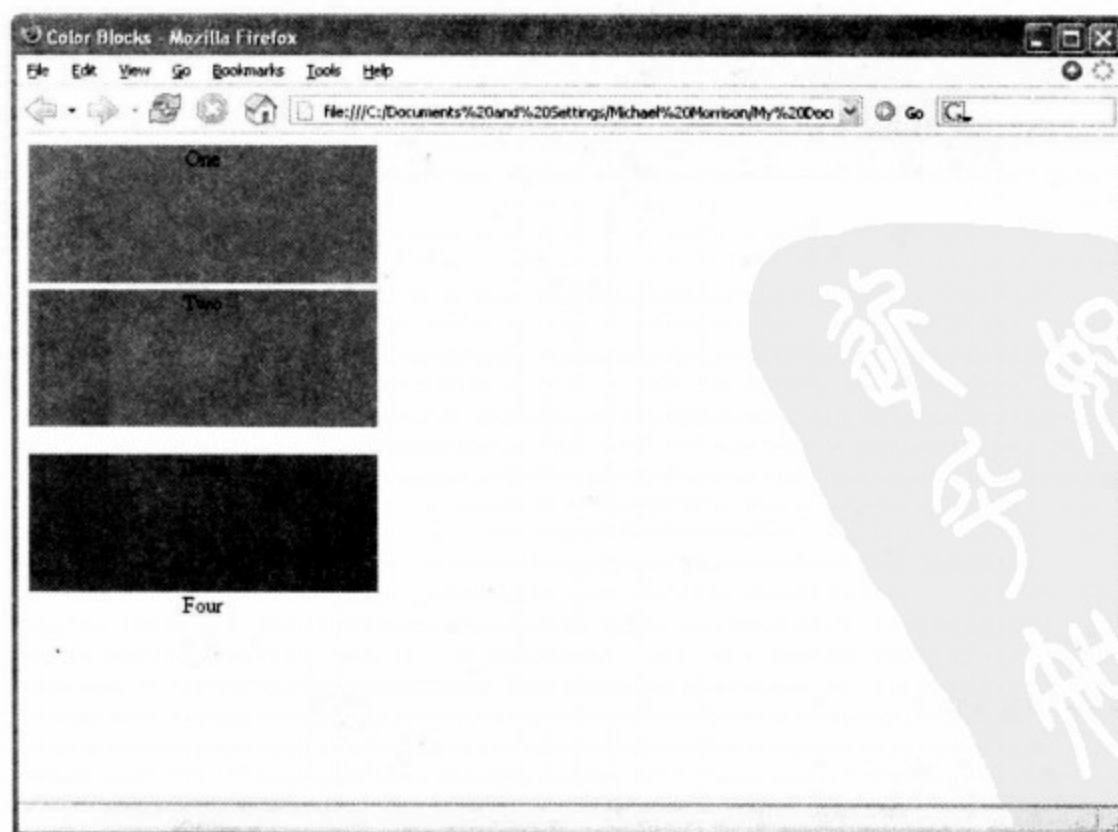
可以使用单独的边距属性来指定边距，也可以只使用 `margin` 属性来指定。如果要将边距设置为百分比，别忘了百分比是基于整个网页而不是元素的大小的。因此，如果将 `margin-left` 属性设置为 25%，元素的左边距将是整个页面宽度的 25%。下列代码演示了如何设置本课使用的 Color Blocks 网页中的一种颜色块的上、下边距：

```
div.two {
    background-color:green;
    margin-top:5px;
    margin-bottom:20px;
}
```

在这个例子中，上边距设置为 5 像素，下边距设置为 20 像素。这段代码的结果如图 14.6 所示。

图 14.6

应用的样式表设置了一种颜色块的上、下边距时 Color Blocks 网页的显示结果





该图说明了第二个 `div` 元素的上、下边距。边距并没有侵占任何颜色块的空间，这些颜色块都保持原来的大小。换句话说，边距只是显示在元素四周。

如果要为样式规则的设置所有边距，只要使用 `margin` 属性，这可以简化代码。这个属性比较灵活，它提供了三种不同的方法来指定样式规则的边距。这些方法随着设置属性时使用多少个值而异：

- 一个值：所有边距的大小。
- 两个值：上、下边距和左、右边距（顺序是先上下后左右）。
- 四个值——上、下、左、右边距（顺序是上、下、左、右）。

下面的例子将样式规则的垂直边距（上、下）设置为 5 像素，将水平边距（左、右）设置为 10%：

```
margin:5px 10%;
```

在这行代码中，上、下边距都设置为 5 像素，左、右边距都设置为 10%。当然，如果要使代码更清晰，使用下列设置也可达到相同的效果：

```
margin:5px 10% 5px 10%;
```

## 14.4 为安全起见而使用填充 (padding)

与边距类似，填充用于给元素增加额外的空间。然而，填充与边距不同，它在元素的矩形区域内增加空间，而不是在矩形区域外。举个例子，如果为一个宽 50 像素、高 30 像素的元素创建了一条样式规则，然后将该规则的填充设置为 5 像素，则余下的内容区域将变为 40×20 像素。另外，因为元素填充显示在内容区域内，因此使用与元素内容相同的样式，包括背景颜色。

可使用填充属性之一来指定样式规则的填充，这些属性与边距属性很像。下面是可用于为样式规则的设置填充的填充属性。

- `padding-top`：设置上填充。
- `padding-right`：设置右填充。
- `padding-bottom`：设置下填充。
- `padding-left`：设置左填充。
- `padding`：通过单个属性设置上、下、左、右填充。

与边距一样，可以使用单独的填充属性来设置样式规则的填充，也可以只用 `padding` 属性。填充也可用度量单位或百分比来表示。下面的例子说明了如何设置样式规则的左、右填充，在元素内容的左、右两边空 10 像素：

```
padding-left:10px;  
padding-right:10px;
```

与边距一样，可用一个属性（`padding` 属性）来设置元素的所有填充。与 `margin` 属性一

样，可使用三种方法来设置填充属性。下面的例子将样式规则的垂直（上、下）填充设置为 12 像素，将水平（左、右）填充设置为 8 像素：

```
padding:12px 8px;
```

下面是一个更清晰的实现相同效果的例子，它指定了所有填充值：

```
padding:12px 8px 12px 8px;
```

## 14.5 确保内容对齐

网页中的内容并不总是填满它所在的矩形区域，因此，控制内容的对齐通常很有帮助。即使矩形区域内的文本有多行，仍可使用对齐，因为你可能希望文本左、右或居中对齐。有两个样式属性可用于控制网页内容的对齐：`text-align` 和 `vertical-align`。第 5 章简要地介绍过这两个属性，深入研究 CSS 布局样式时可深入介绍它们。

`text-align` 样式使元素在其定界区域内水平对齐，其取值可以是 `left`、`right`、`center` 或 `justify`。`justify` 使元素两端对齐。下面的例子使用 `text-align` 属性使虚构的网页广告居中对齐：

```
div.ad {
  width:275px;
  margin-bottom:10px;
  border:5px double black;
  color:black;
  background-color:yellow;
  text-align:center
}
```

在这条样式规则中，定义的最后一个是样式属性将 `text-align` 样式设置为 `center`，从而使元素 `div.ad` 在其父元素内居中对齐。如果该元素的父元素是网页，它将位于网页中央。

`vertical-align` 属性与 `text-align` 属性类似，只是用于使元素在垂直对齐。`vertical-align` 属性指定元素如何与其父元素（在有些情况下，是与网页中元素的当前行）对齐。“当前行”指的是显示在元素内的元素所处的垂直位置，换句话说，是内联元素。如果几个内联元素显示在同一行，可将其垂直对齐方式设置为相同，使它们垂直对齐。一个这样的例子是一行并排显示的图像——使用 `vertical-align` 属性可让它们垂直对齐。

下面是 `vertical-align` 属性的可能取值。

- `top`：将元素的顶部与当前行对齐。
- `middle`：将元素的中心与当前行对齐。
- `bottom`：将元素的底部与当前行对齐。
- `text-top`：将元素的顶部与父元素的顶部对齐。
- `baseline`：将元素的基线与父元素的基线对齐。
- `text-bottom`：将元素的底部与父元素的底部对齐。

- `sub`: 将元素作为父元素的下标。
- `super`: 将元素作为其父元素的上标。

下面的例子演示了如何使用 `vertical-align` 属性使文本垂直居中对齐:

```
div.ad {
  width:275px;
  margin-bottom:10px;
  border:5px double black;
  color:black;
  background-color:yellow;
  text-align:center;
  vertical-align:middle
}
```

这段代码表明,要修改样式规则使元素垂直对齐很简单。在这个例子中,元素 `div.ad` 与父元素在居中对齐。

## 14.6 管理文本流

前面讨论了“当前行”的概念,它指的是用于放置网页元素的不可见行。这一行与网页中的元素流有关;它在元素在网页中依次排列时起作用。元素流的一部分是文本流。当文本与其他元素(如图像)混合时,控制文本如何沿其他元素绕排很重要。

下面是一些可用于控制文本流的样式属性。

- `float`: 决定文本如何沿元素绕排。
- `clear`: 禁止文本沿元素绕排。
- `overflow`: 在元素太小而不能包含所有文本时控制文本的溢出方式。

`float` 属性用于控制文本如何沿元素绕排。它可设置为 `left` 或 `right`。这两个值决定元素相对于浮动文本应如何放置。因此,如果将图像的 `float` 属性设置为 `left`,图像将放在文本的左边。

使用 `clear` 属性可防止文本出现在元素旁边,`clear` 属性可设置为 `none`、`left`、`right` 或 `both`。`clear` 属性的默认值是 `none`,表示排列文本时不考虑元素;`left` 导致文本不出现在元素左边,除非网页左边没有元素;同样,`right` 值表示文本出现在元素右边;`both` 值表示文本出现在元素两边。

`overflow` 属性处理溢出文本,溢出文本指的是在元素的矩形区域内放不下的文本;如果将元素的 `width` 和 `height` 设置得太小,就可能发生这种问题。`overflow` 属性可设置为 `visible`、`hidden` 或 `scroll`。`visible` 自动扩大元素使矩形区域可容纳溢出文本,这是 `overflow` 属性的默认值。`Hidden` 保持元素的大小不变,使溢出文本不可见;`scroll` 可能是最有趣的值,它使元素添加滚动条,以便可以通过滚动看到所有文本。

## 14.7 利用样式表模板

前面提过多次,样式表让你能够将网页内容(文本、图像和链接)同其显示方式分开。

这就意味着 HTML 代码可将重点放在信息上，然后根据样式表中指定的规则将这些信息显示在屏幕上。因此，通过使用正确的样式表，只要做少量的工作就可以创建有趣的网页。

样式表模板是可复用的样式表，用于解决特定问题，如让你能够创建新闻稿或博客。虽然第 21 章将介绍如何使用在线博客服务来创建博客，但我认为，学习如何只使用一个样式表模板和一些 HTML 代码创建简单博客也是不错的。

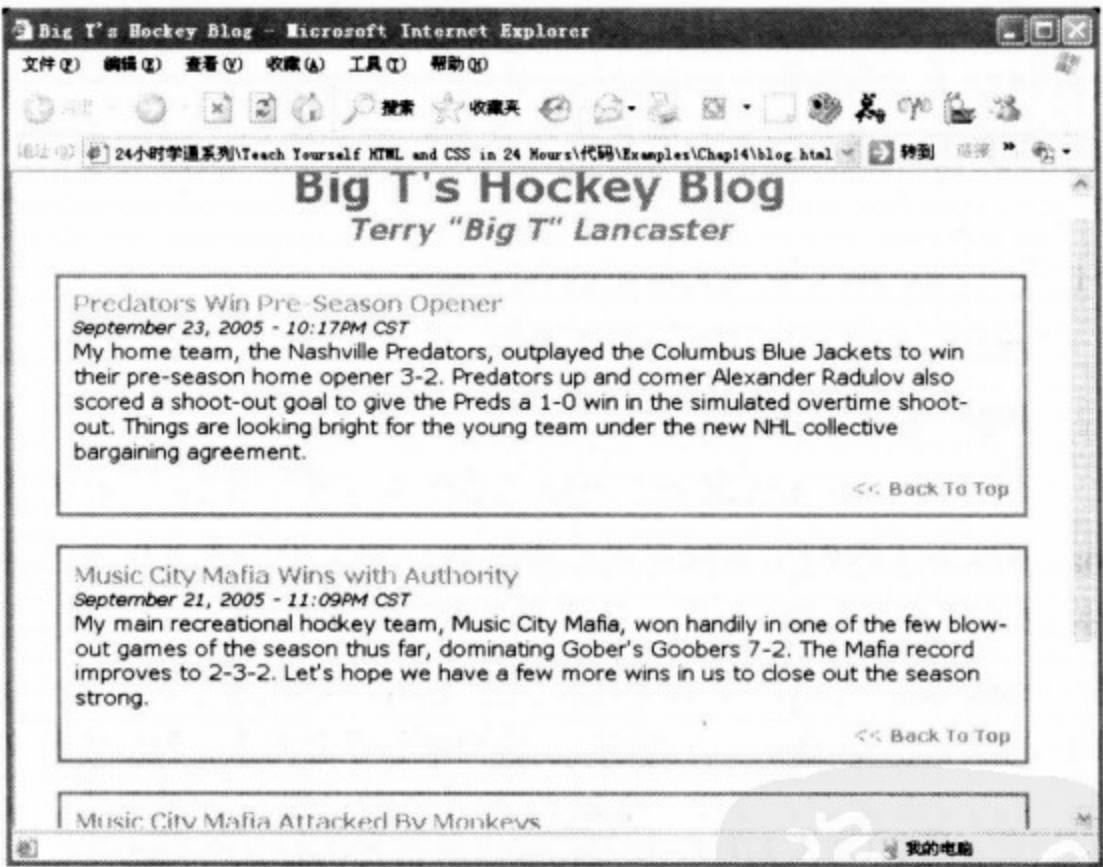
By the Way

注意：如果你不熟悉博客，我来解释一下，“博客”（blog）代表网络日记，它指的是一种特殊网页，包含由某人撰写或针对某个主题的一系列文章。政治博客现在很流行，也有关于其他各种主题的博客。

首先，在最简单的情况下，博客是一系列简短的文章，有点像短小的新闻故事，理解这一点很重要。每篇博客通常都由题目、日期和时间以及实际文本组成。博客按降序显示，即最新的（最近的）博客显示页面的最前面。

图 14.7 是一个简单的博客页面，这个网页的样式完全由外部样式表指定。下面你将看到该博客使用的样式表。

图 14.7  
只用一些 HTML 代码和一个外部样式表就能创建简单的博客



Coffee Break

休息一下：要查看其他博客的样子，可访问 <http://www.blogwise.com/>。这个网站是一个全球博客目录，可按地区或主题来浏览。

程序清单 14.5 列出了这个博客的 HTML 代码。注意，HTML 代码只包含博客的内容，并不关心它在网页浏览器中如何显示。

程序清单 14.5 这个网页的布局和格式化完全依赖于样式表 blog.css

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
```

```

"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Big T's Hockey Blog</title>
    <link rel="stylesheet" type="text/css" href="blog.css" />
  </head>

  <body>
    <div class="pagetitle" id="top">Big T's Hockey Blog</div>
    <div class="author">Terry &quot;Big T&quot;; Lancaster</div>

    <div class="entry">
      <div class="title">Predators Win Pre-Season Opener</div>
      <div class="datetime">September 23, 2005 - 10:17PM CST</div>
      <div class="text">My home team, the Nashville Predators, outplayed the
      Columbus Blue Jackets to win their pre-season home opener 3-2. Predators
      up and comer Alexander Radulov also scored a shoot-out goal to give the
      Preds a 1-0 win in the simulated overtime shoot-out. Things are looking
      bright for the young team under the new NHL collective bargaining
      agreement.</div>
      <div class="link">
        <a href="#top">&lt;&lt; Back To Top</a>
      </div>
    </div>

    <div class="entry">
      <div class="title">Music City Mafia Wins with Authority</div>
      <div class="datetime">September 21, 2005 - 11:09PM CST</div>
      <div class="text">My main recreational hockey team, Music City Mafia, won
      handily in one of the few blow-out games of the season thus far,
      dominating Gober's Goobers 7-2. The Mafia record improves to 2-3-2. Let's
      hope we have a few more wins in us to close out the season strong.</div>
      <div class="link">
        <a href="#top">&lt;&lt; Back To Top</a>
      </div>
    </div>

    <div class="entry">
      <div class="title">Music City Mafia Attacked By Monkeys</div>
      <div class="datetime">September 18, 2005 - 10:43PM CST</div>
      <div class="text">The Mafia boys were attacked by a rogue pack of wild Ice
      Monkeys tonight, and the result wasn't pretty. The 5-1 loss saw few bright
      moments, with one of them being a goal by Chet that was set up by Duke
      early in the game. It was downhill from then on, as the Monkeys proceeded
      to frustrate and aggravate the Mafia. Retribution plans are already
      underway!</div>
      <div class="link">

```



```

        <a href="#top">&lt;&lt; Back To Top</a>
    </div>
</div>
</body>
</html>

```

这段代码表明，该博客使用 7 个样式类来组织其内容。

- **pagetitle**: 博客的标题，以大字体居中显示在页面顶部。
- **author**: 博客的作者，居中显示在页面顶部的标题下面。
- **entry**: 用于保存一篇博客文章的博客级别的元素。
- **title**: 博客文章的标题。
- **datetime**: 博客文章的日期和时间。
- **text**: 博客文章的正文。
- **link**: 博客文章中让用户返回博客页面开头的链接。

仔细观察程序清单 14.5 和图 14.7，可看出这些样式表是如何映射到博客页面内容的。如果看不出来，程序清单 14.6 可帮助理解。

#### 程序清单 14.6 样式表 blog.css 可作为可复用的博客网页模板

```

body {
    background-color:#FFFPCA;
}

div {
    font-family:Verdana;
}

div.pagetitle {
    text-align:center;
    font-size:24pt;
    font-weight:bold;
    color:#19619A;
}

div.author {
    text-align:center;
    font-size:16pt;
    font-weight:bold;
    font-style:italic;
    color:#19619A;
}

div.entry {
    background:#EEEEEE;
    border:3px solid #19619A;
    margin:20px;
}

```

```
padding:10px;
}

div.title {
  text-align:left;
  font-size:12pt;
  font-weight:bold;
  color:#19619A;
}

div.datetime {
  text-align:left;
  font-size:10pt;
  font-style:italic;
}

div.text {
  text-align:left;
  font-size:11pt;
}

div.link {
  text-align:right;
  font-size:10pt;
  margin-top:10px;
}

a { font-weight:bold }
a:link {
  color:#19619A;
  text-decoration:none;
}

a:visited {
  color:#19619A;
  text-decoration:none;
}

a:hover {
  color:#3D9DE9;
  text-decoration:none;
}

a:active {
  color:#19619A;
  text-decoration:none;
}
```

---

这个样式表包含前面提到的样式类的所有细节，还有一些样式规则有助于实现博客效果。首先，`body` 选择符用于指定整个网页的背景颜色。然后指定了所有 `div` 元素都使用的字体（`Verdana`），即整个网页都使用相同的字体。

接下来指定具体的样式类，首先是 `div.pagetitle` 和 `div.author`。样式类 `div.entry` 很有趣，它设置了边框和更浅的背景色，以突出网页中的每篇博客文章。其他博客样式类都使用不同的字体设置来改进博客的布局和可读性。

样式表 `blog.css` 中的最后几条样式规则是针对链接的，指定了将鼠标指向和单击时，链接将如何变化。`text-decoration` 样式属性删除链接的下划线，`color` 属性用于将鼠标指向链接时突出链接。

可稍微修改样式表 `blog.css`，将其用来创建自己的博客。当然，如果要创建更优美、更专业的博客，可以参阅第 21 章。

## 14.8 总结

本章继续探索 CSS 样式表，主要介绍如何利用样式表来处理网页布局。首先学习了相对定位和绝对定位的区别，以及如何将它们用于元素定位；然后，学习了其他 CSS 定位特性，如 `z-index`、边距、填充和内容对齐；接下来还学习了一些巧妙的样式属性，让你能够控制网页中的文本流；最后，演示了如何为简单博客创建可复用的样式表模板。

## 14.9 问与答

问：如何选择相对定位和绝对定位？

答：关于相对定位与绝对定位的选择，虽然没有固定的指导方针，但总体思路是，绝对定位只用于需要精确控制内容的位置时。因为绝对定位在定位内容时让你能够精确到像素，而相对定位在内容如何定位上的不可预测性更大。这不并不是说相对定位不能将网页元素定位好，只是说绝对定位更精确。当然，这也使得绝对定位在屏幕大小改变时更易受影响，而这是你无法控制的。

问：如果两个相互重叠的元素没有指定 `z-index`，如何知道哪个元素将显示在方面？

答：如果相互重叠的元素没有指定 `z-index`，在网页中较后出现的元素将显示在上面。这一点很容易记住，只要想象一下浏览器从 HTML 文档逐一读取元素并绘制它们；后面读取的元素将绘制在前面读取的元素上面。

## 14.10 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 14.10.1 测验

1. 绝对定位与相对定位的区别是什么？

2. 使用什么 CSS 样式属性来控制元素的堆叠顺序?
3. 编写 HTML 代码, 使句子 “What would you like to” 从在浏览器窗口的左上角开始显示, 使 “THROW TODAY?” 以大字体显示在窗口左上角下方 80 像素和右边 20 像素处。

### 14.10.2 答案

1. 在相对定位中, 内容根据网页的走向显示, 在 HTML 代码中, 每个元素显示在它前面的元素后面; 而绝对定位让你能够设置网页内容的精确位置。

2. 使用 z-index 控制元素的重叠顺序。

3.

```
<span style="position:absolute; left:0px; top:0px">
What would you like to</span>
<h1 style="position:absolute; left:80px; top:20px">
THROW TODAY?</H1>
```

## 14.11 练习

1. 修改前一章的曲棍球队员样式表, 使用绝对定位将每个元素定位于网页的精确位置。
2. 通过添加背景图像、修改颜色和调整字体, 进一步改进本章的博客样式表。

## 第 15 章

# 创建打印友好的网页

如果使用过在线地图工具，如 MapQuest 或 Google Maps，肯定有过需要打印网页的经历。网页并不只是完全为在屏幕上浏览而设计的。你可能没有认识到这一点，但是有时候是可能要为需要打印网页脱机阅读的用户特别设计并提供打印友好（print-friendly）的网页版本的。实际上，CSS 使得创建可根据浏览方式不同而改变外观的网页非常简单。本章将介绍如何创建这样的网页。

### 练习

在本章的学习过程中，考虑自己的网页打印出来是否好看。然后考虑如何修改这些网页，使其打印出来的效果更好。下面是一些要考虑的因素。

1. 网页是否使用了与文本反差强烈的背景图像或背景颜色？这种网页可能因为背景而难以打印，因此可考虑在打印版本中不使用背景并使用黑色文本。
2. 网页是否包含许多链接？如果是，可考虑在打印时改变链接的外观，使链接不那么突出。别忘了，在纸上不能单击链接！
3. 最后，网页中的图像是否都是必要的？色彩丰富的图像在大多数打印机上都很费墨，因此可考虑在打印时删除一些图像。



### 15.1 如何成为打印友好的网页

在前面的“练习”中简要地介绍过该主题，但还需要探讨如何使网页易于打印。首先，有些网页已经是打印友好的。如果网页使用白色背景和黑色文本，包含的图像很少，就无需考虑设计打印友好的版本。然而，如果网页的背景为黑色，包含动态链接和许多图像，这样的网页对大多数打印机来说是不友好的。

考虑使网页更易于打印时，要记住的主要一点是介质的限制。换句话说，什么使打印的



网页与显示在计算机屏幕上的网页不同？一个明显的区别是大小——打印的网页大小是固定的，通常是 8.5×11 英寸，而计算机屏幕有多种大小。除大小外，打印的网页也有颜色限制，即使在彩色打印机上也如此。当用户只对网页的文本感兴趣时，很少用户会浪费油墨来打印彩色背景。

大多数用户对网页中的主要内容文本外的内容不感兴趣。例如，请看图 15.1，该图是我个人网站上的一篇博客文章作为普通网页显示的样子。

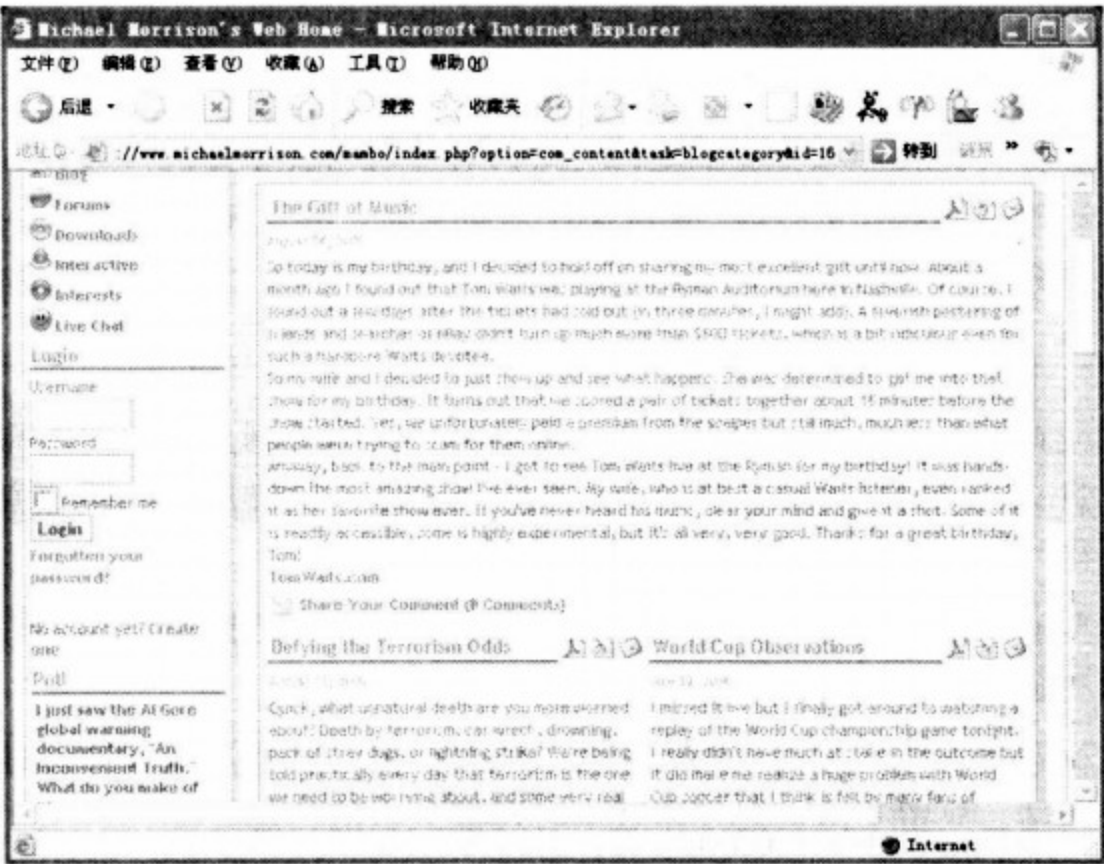


图 15.1  
这个网页不是打印友好的，因为它在主要的文章外使用了许多菜单和其他内容

如果注意看图中网页的右上角，可以看到上面有一个打印图标，让你能够浏览该网页的打印友好版本。图 15.2 显示了这篇博客文章的打印友好版本，它更适合打印。



图 15.2  
网页的打印友好版本，它将文章中的文本分离出来以便单独打印

从上图可以看出，打印友好的版本对原来的网页有很大改进，至少从打印机的角度看是这样的。所有导航菜单、围绕页面起修饰作用的其他元素以及背景图像都被删除。

为了让读者理解创建打印友好的网页时需要特别考虑的因素，下面列出了至少应该考虑的一些修改。

- 删除页面背景，使打印的网页背景为白色。
- 将文本颜色改为黑色：文本可有颜色，但最好使用黑色。
- 保证字体足够大，使打印出来的网页易于阅读——可能需要针对不同的字体大小进行测试。

### By the Way

**注意：**有些网页设计人员甚至建议将面向打印的网页字体改为 Serif，打印时使用这种字体更容易阅读。我对此持保留意见，因为我通常更喜欢在打印时保持网页的核心风格，即在打印时使用与网页在屏幕上显示的版本相同的字体。

- 删除链接格式或转换为基本的下划线链接。我认为在打印的网页中保留特殊的链接外观真的没有意义，但有些网页设计者喜欢保留下划线，这样用户能够知道原来的网页中包含链接。
- 删除所有不必要的图像。这通常包括对表达网页内容不重要的图像，如导航按钮、大多数广告和动画。

除这些建议外，在标题下添加一行关于网页作者、网页 URL 和网页版权等信息也可能有用。这些是用户离开网站、只留下网页的打印版本时可能失去的信息。

要记住，我并不是建议将所有这些修改都应用于网页。这里只是说明如何使网页打印友好，这样在创建面向打印机的样式表时就可以做得更好。可创建只在打印网页时才应用的样式表。下一节将介绍如何创建这样的样式表。

## 15.2 应用与介质相关的样式表

图 15.1 包含一个小型打印机图标，让用户能够浏览网页的打印友好版本。这种图标在许多新闻网站上很流行，这是一个重要特性，如果没有它，用户打印网页时将很麻烦，而且会在环绕文章的图像和广告上浪费纸墨。虽然打印机图标是一种直观而有效的方法，但还有一种更自动化的方法。

我说的是使用面向打印的样式表，当用户选择打印网页时，该样式表就自动应用于网页。CSS 支持“与介质相关的样式表”的概念，它指面向具体介质（如屏幕或打印机）的样式表。然而，CSS 不仅限于支持这两种介质。下面是关于 CSS 支持使用独特样式表的介质列表：

- 计算机屏幕；
- 打印机；
- 电视机；

- 投影仪屏幕;
- 手持设备;
- 盲文 (braille);
- 浮凸印刷 (embossed);
- 只有文本;
- 声音。

这些介质当中最有趣的可能是最后一个, 它使网页能够被朗读。显然, CSS 的架构师预见到的 Web 比我们现在想到的还要广泛, 因为现在我们设计的网页主要是为了在计算机屏幕上显示。虽然我并不是建议考虑太多声音网页设计, 但这可能成为将来发展方向。

关于应用于其他介质的样式表的好消息是, 你不需要学习新东西。可能在设计声音网页时需要学习一些新技巧, 但现在你可以使用学习过的样式属性来创建面向打印的样式表。真正的技巧在于知道如何将样式表应用于特定的介质。

前面提到, `<link />` 标签用于将样式表链接到网页。这个标签支持属性 `media`, 可使用它来指定样式表将应用的介质。默认情况下, 这个属性设置为 `all`, 即如果没有专门指定, 外部样式表将用于所有介质。下面是其他一些可用于该属性的值: `screen`、`print`、`tv`、`projection`、`handheld`、`braille`、`embossed` 和 `aural`。可以看出, 这些属性值对应于前面列出的介质。

要为网页指定面向打印的样式表, 需要使用两个 `<link />` 标签, 一个用于打印机, 一个用于其他所有介质。下面是执行这项任务的代码:

```
<link rel="stylesheet" type="text/css" href="player.css" media="all" />
<link rel="stylesheet" type="text/css" href="player_print.css" media="print" />
```

在这个例子中, 网页链接了两个样式表。第一个样式表将 `media` 属性设置为 `all`, 因此面向所有介质。如果没有其他设置, `player.css` 样式表将应用于所有介质。然而, 由于有了第二个样式表, `player_print.css` 样式表将用于打印网页。

**注意:** 也可以使用 `@import` 命令来链接面向介质的样式表。例如, 下列代码的功能与上面的 `<link />` 代码相同:

```
@import url(player.css) all;
@import url(player_print.css) print;
```

**警告:** 你可能想在前一段代码中第一个链接的样式表中指定 `media="screen"`。虽然在普通网页浏览器中浏览网页时这样也可行, 然而, 如果用户使用手持浏览器或其他类型的介质, 这可能引起问题。换句话说, 样式表只能应用于 `media` 属性中指定的介质类型, 不能应用于其他介质类型。

**By the way**

**Watch out**

可在一个 `<link>` 标签中指定多种介质类型, 彼此用逗号分开, 如下所示:

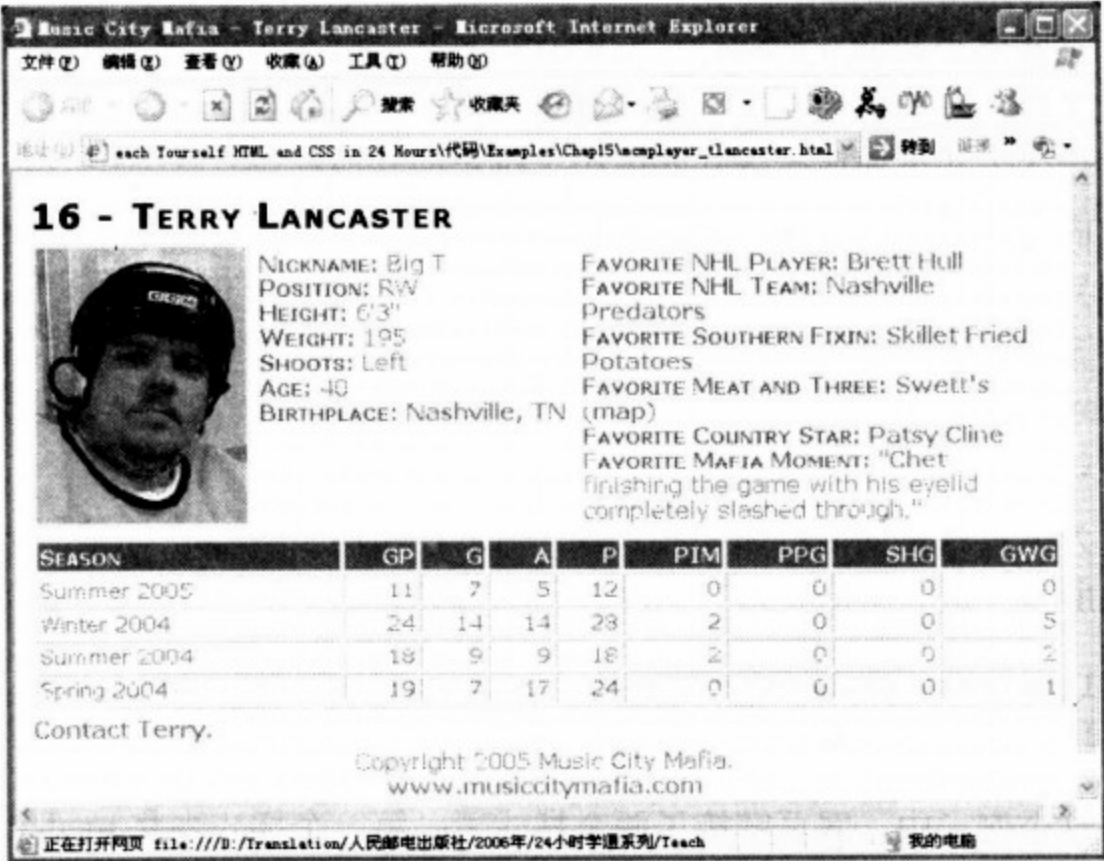
```
<link rel="stylesheet" type="text/css" href="player_pp.css" media="print, projector" />
```

这段代码使 player\_pp.css 样式表只应用于 print 和 projector 这两种介质类型。

### 15.3 设计用于打印网页的样式表

本章前面讲过创建打印友好的网页所需的修改，现在就来创建打印友好的样式表。先来看一下使用普通（屏幕）样式表的网页，见图 15.3。

图 15.3  
使用 CSS 样式的曲棍球队员网页在普通网页浏览器中的显示结果



该图显示了网页在普通网页浏览器中显示的结果。虽然在印刷出的网页中不能区分不同的颜色，但可以访问 <http://www.sampublishing.com/> 上的网页文件。实际上，这个网页还不是打印友好的，可再进行一些改进。

下面的修改可使该网页打印出来的效果更好：

- 将所有文本的颜色改为黑色；
- 删除链接的格式（粗体及其颜色）；
- 将队员的两部分信息垂直排列，因为这些信息在打印的网页上可能水平排列不下；
- 删除整个联系人链接。

前两个对常规样式表的修改很简单，主要是修改或删除已有的样式。然而，第三个修改就需要思考一下。因为打印的网页大小是固定的，因此在打印的网页上所有元素都可以使用绝对定位，这比较容易将内容区域放到希望的位置。最后，列表中最最后一条很容易实现，只要将 contact 元素的 display 样式属性设置为 none。

**警告：**虽然绝对定位在曲棍球队员网页中可行，但在指定面向打印的网页的样式时它并不总是可行。具体地说，如果网页的内容打印出来不只一页，最好使用相对定位，让内容分为多页。

**Watch  
Out!**

程序清单 15.1 包含了样式表 `player_print.css` 的 CSS 代码，它包含上述修改，形成一个特别适于打印曲棍球队员网页的样式表。

#### 程序清单 15.1 面向打印的曲棍球队员样式表的 CSS 代码

```
body {
    font-family: Verdana, Arial;
    font-size: 12pt;
    color: black;
}

div {
    padding: 3px;
}

div.title {
    font-size: 18pt;
    font-weight: bold;
    font-variant: small-caps;
    letter-spacing: 2px;
    position: absolute;
    left: 0in;
    top: 0in;
}

div.image {
    position: absolute;
    left: 0in;
    top: 0.5in;
}

div.info {
    position: absolute;
    left: 1.75in;
    top: 0.5in;
}

div.favorites {
    position: absolute;
    left: 1.75in;
    top: 2in;
}

div.footer {
```



```
    position:absolute;
    text-align:left;
    left:0in;
    top:9in;
}

table.stats {
    width:100%;
    text-align:right;
    font-size:11pt;
    position:absolute;
    left:0in;
    top:3.75in;
}

div.contact {
    display:none;
}

.label {
    font-weight:bold;
    font-variant:small-caps;
}

tr.heading {
    font-variant:small-caps;
    background-color:black;
    color:white;
}

tr.light {
    background-color:white;
}

tr.dark {
    background-color:#EEEEEE;
}

th.season, td.season {
    text-align:left;
}

a, a:link, a:visited {
    color:black;
    font-weight:normal;
    text-decoration:none;
}
```

这段代码中最绝妙的地方在于，所有绝对定位代码都以英寸（in）为度量单位。考虑到打印的网页大小以英寸为单位，而不是以像素为单位，因此这是合理的。仔细研究这段代码，

可以发现所有文本都是黑色的，链接的特殊格式都已经删除，内容区域现在是绝对定位的，这样可以使内容准确地显示在希望的位置。

## 15.4 预览网页的打印结果

图 15.4 是打印友好的曲棍球队员网页版本在 Internet Explorer 的“打印预览”窗口中显示的样子。

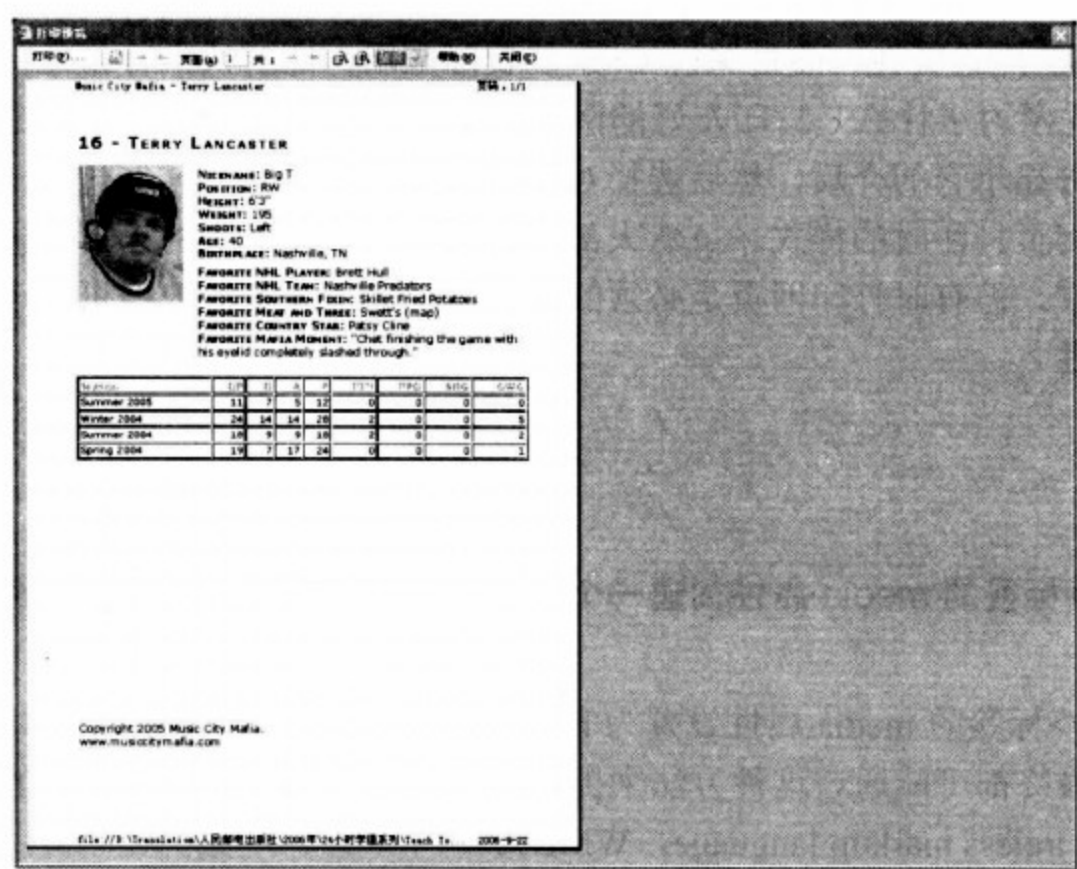


图 15.4  
可用网页浏览器的“打印预览”来查看打印友好的网页版本

该图揭示了该网页打印友好版本的一个微妙变化——脚注现在显示在网页的底部。

图 15.5 说明了在实际情况下如何使用打印友好的网页。该图在 Adobe Acrobat Reader 中显示同一曲棍球队员网页的 PDF 文档。

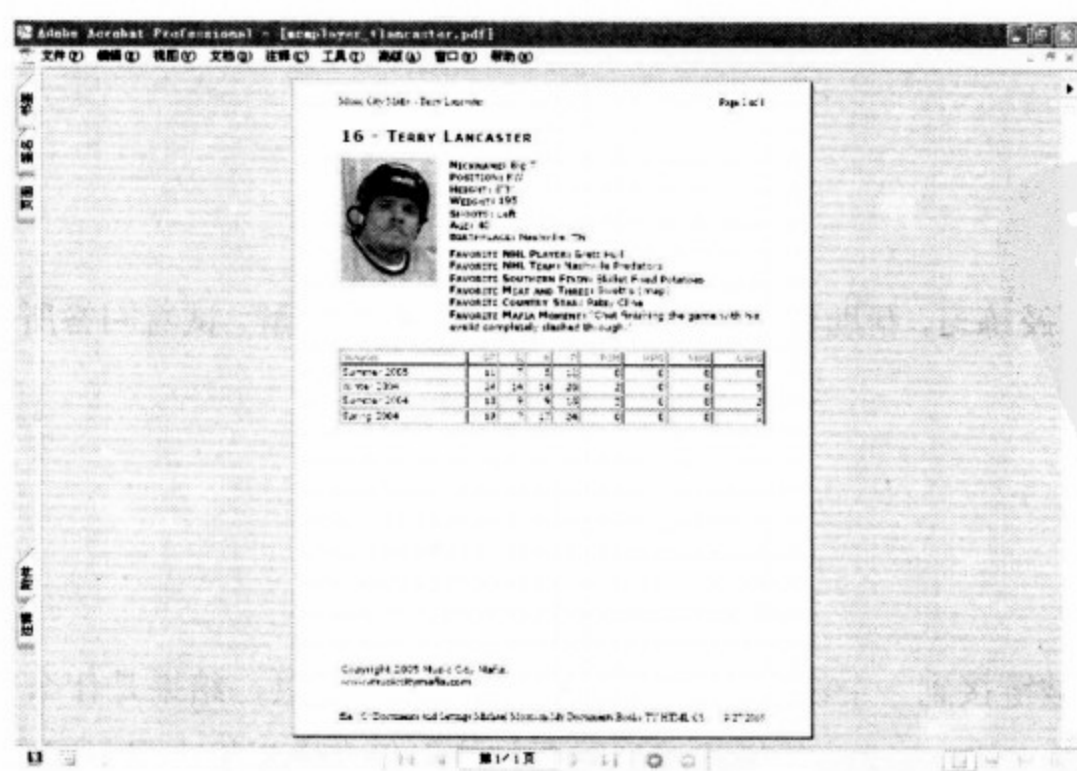


图 15.5  
通过将曲棍球队员网页打印到 Adobe 的 PDFWriter 虚拟打印机，将该网页转换为 PDF 文档

我使用 Adobe 的 PDFWriter 虚拟打印机来“打印”曲棍球队员网页。这以一种易于共享的电子格式创建了打印友好的网页版本。

**By the Way**

**注意：**Adobe 的 PDFWriter 虚拟打印机作为 Adobe Acrobat 产品线的一部分发布。关于 Acrobat 的详情，请参见 <http://www.adobe.com/products/acrobat/>。

## 15.5 总结

本章没有再介绍 CSS 样式属性，而是探讨 CSS 的实际应用，即解决一种常见问题：打印网页。读者在本章首先学习了什么是打印友好的网页。在这里，你了解了内置于 CSS 的一种机制，即允许页面区分显示它的介质，然后选择对应的样式表。最后，你创建了一个面向打印的样式表，它指定网页打印时的样式。虽然大部分用户更喜欢在较大的计算机屏幕上浏览网页，而不是在纸上看，但有时打印网页是必须的。通过提供易于打印的网页，可为网页访问者提供最大的灵活性。

## 15.6 问与答

**问：**可以使用<link />标签的 media 属性创建一个专门用于在手持设备中浏览网页的样式表吗？

**答：**可以。将<link />标签的 media 属性设置为 handheld，可以将样式表设为专门针对手持设备。所有手机网站最终都可能通过这种方法来提供手机网页，而不是使用专门的标记语言，如无线标记语言（wireless markup language, WML）。

**问：**我还需要在网页上提供一个打印机图标，让用户能够打印网页吗？

**答：**不需要。在本章学习的链接样式表方法让你能够不通过任何特别的网页链接来支持打印友好的网页。然而，如果让用户能够在浏览器中看到打印友好的网页版本，可以链接到另一个版本的网页，它使用打印专用的样式表作为主要（浏览器）样式表。也可以在网页中提供一些说明（fine print），提示用户使用浏览器的打印预览功能来查看打印友好的网页版本。

## 15.7 作业

作业包含一些测试题及练习，帮助你巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 15.7.1 测验

1. 什么是声音网页？
2. 如果外部样式表被链接到一个没有指定任何 media 属性的网页，结果是什么？
3. 如何链接一个名为 freestyle.css 的样式表到网页，使得仅在电视机上显示网页时才应

用该样式表？

### 15.7.2 答案

1. 声音网页是设计用于倾听的网页。现在，对于 Web 开发者来说，通过 CSS 设计真正的声音网页还是一种设想。然而，这提供了一种 Web 发展方向，它很大程度上依赖于样式表，不仅仅应用字体、颜色与定位等。

2. Media 属性的默认值是 all，这意味着该样式表将针对所有介质。

3.

```
<link rel="stylesheet" type="text/css" href="freestyle.css"
media="tv" />
```

## 15.8 练习

1. 选择一个有许多颜色和图像的网页，为它创建一个打印友好的样式表。务必在网页中添加一个<link />标签，它链接到面向打印的样式表。

2. 使用一个<link />标签，将其 media 属性设置为 handheld 值，为网页创建面向手持设备的版本。涉及的概念与创建面向打印的页面相同，但需要应对的是很小的屏幕，而不是打印纸。可以发布页面，然后使用手机或手持计算机浏览器打开它，对该页面进行测试。

## 第 16 章

# 使用框架创建多页面布局

读者可能见过这样的网站，网页浏览器似乎允许在几个不同的网页间切换。浏览器确实允许在几个网页之间切换。称为框架（frame）的 HTML 特性让用户能够将浏览器窗口分为几个包含不同网页的区域，每个区域称为一个框架。当然，从用户的角度看，所有 Web 内容形成一个窗口，但实际上有几个不同的网页在工作。

框架与表格（在第 11 章介绍过）类似，它们都让你能够将文本和图像排列成行和列；但与表格单元格不同的是，框架可包含链接来修改其他框架（或本身）的内容。例如，一框架显示目录，根据用户单击的链接改变在另一框架中显示的内容。

### 练习

框架是一种同时排列和显示多个网页的方法。如果在阅读本章之前准备好一些相关的网页，将可以更快地学习本章，也会更有收获。

- 如果针对网站的索引网页或目录，将它复制到另一个文件夹中，以便使用这个网页进行试验，而不会修改原始网页。另外，复制几个索引链接指向的网页。
- 修改本章的框架示例，将它们放到自己的网页中。

## 16.1 什么是框架

乍一看，图 16.1 与普通网页没什么不同，但实际上它是两个不同的 HTML 页面，显示在同一个网页浏览器窗口中。每个网页都显示在自己的框架中，两个框架上下相邻，由一条水平线分开。

框架是浏览器窗口中的一个矩形区域，每个框架显示一个网页，与其他网页的框架相邻。

如果单击图 16.1 中上面框架内的导航链接，使用框架的主要优点将显现出来。在这个例子中，上面的框架不变，而下面的框架将加载并显示一个新网页，如图 16.2 所示。



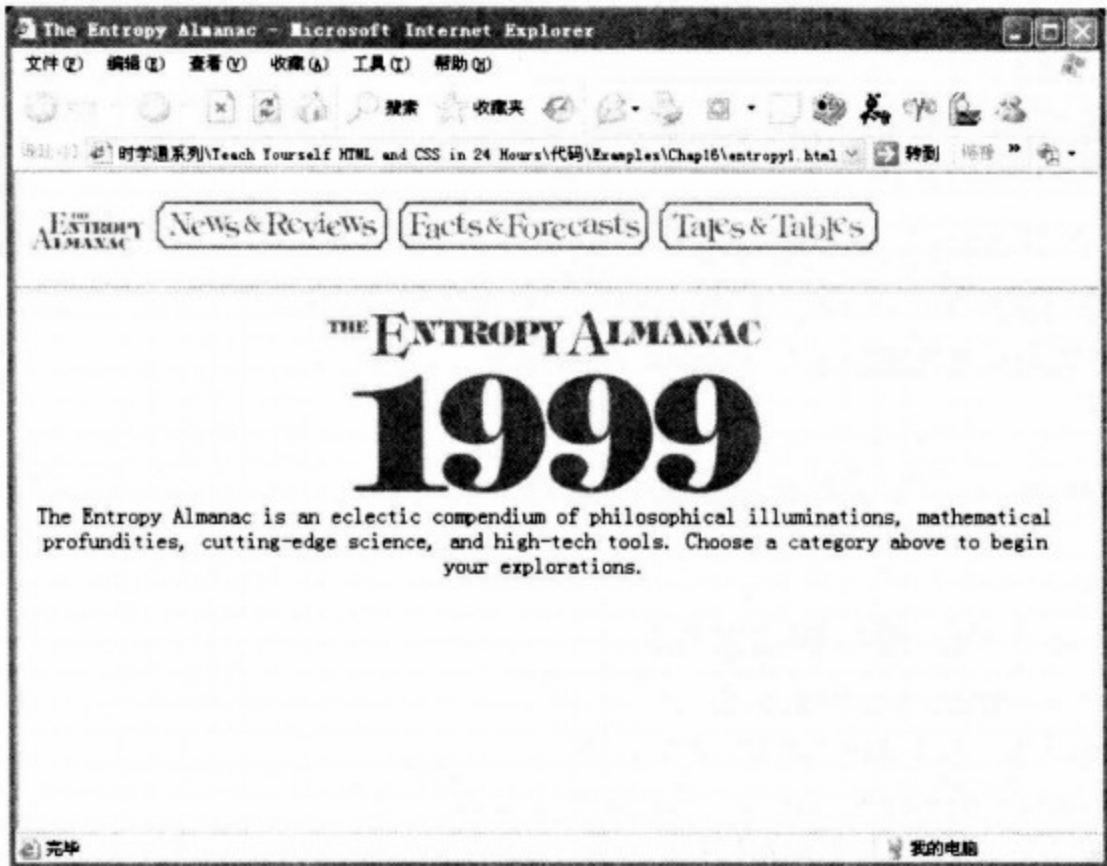


图 16.1  
框架让你能够同时显示多个网页

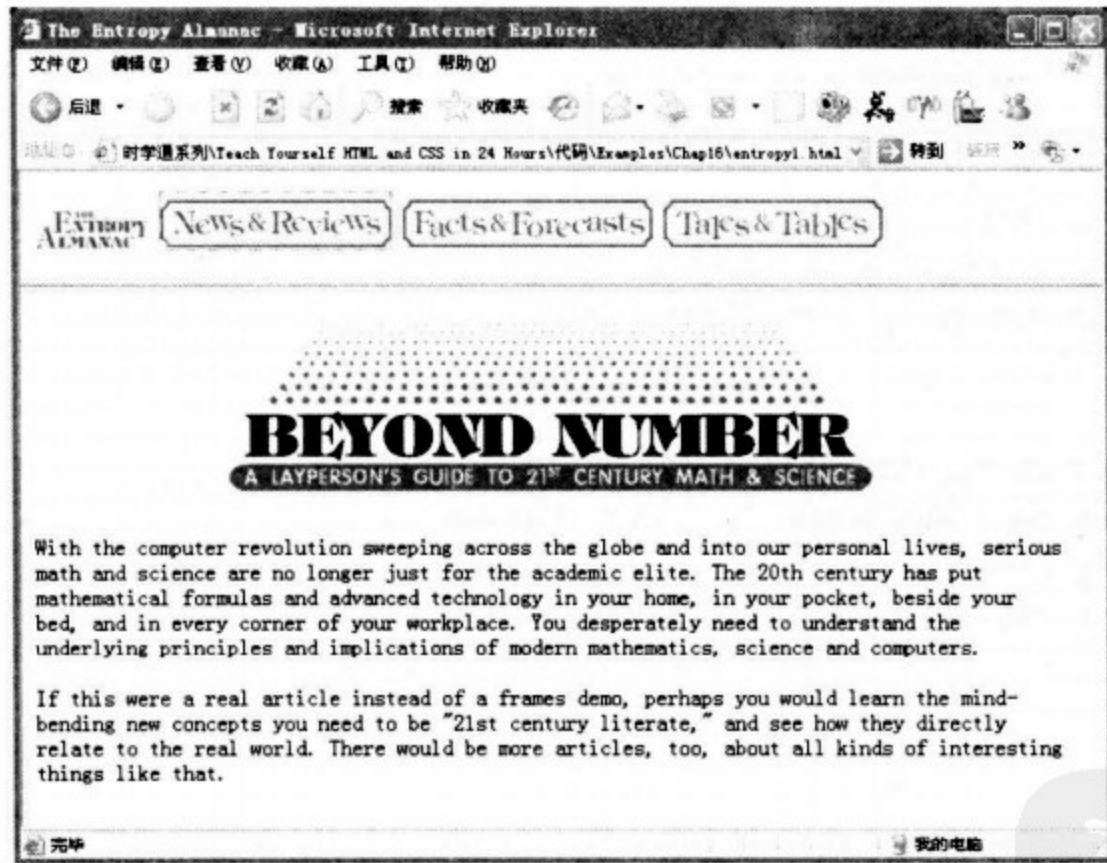


图 16.2  
单击图 16.1 中的 News & Reviews，窗口下半部将打开一个新网页，但上面的框架不变

虽然框架很有用，但坦率地说，我并不是很喜欢使用。原因是如果滥用框架，创建太多包含大量交叉链接的框架，有时候会使用户感到困惑。网页适当地包含两 3 个框架，效果可能很好，但再多的框架就可能给用户带来困惑和不便。

## 16.2 创建框架集

在 HTML 代码中使用框架并不困难。那么图 16.1 和图 16.2 所示的网页是如何创建的呢？首先，将每个框架的内容创建为普通 HTML 页面。这些页面使用的标签都是前几章介绍过的。

要将这些网页放在一起，需要使用一种称为框架集文档的特殊页面。

### 16.2.1 创建框架集文档

框架集文档实际上并不包含内容，它只是告诉浏览器要加载哪些网页以及如何在浏览器窗口中排列这些网页。程序清单 16.1 显示了图 16.1 和图 16.2 所示的 Entropy Almanac 网站的框架集文档。这个例子中的网页倒退到了 1990 年代，那时候的 Internet 与“熵”（entropy）是同义词。

框架集文档是 HTML 页面，指示网页浏览器将窗口分为几框架，并指定每一框架应显示哪个网页。

程序清单 16.1 图 16.1 所示网站的框架集文档

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Entropy Almanac</title>
  </head>

  <frameset rows="80,*">
    <frame src="banner.html" name="top" />
    <frame src="greeting.html" name="main" />
  <noframes>
    <body>
      <h1>The Entropy Almanac</h1>
      Your browser does not support frames. Please <a
href="noframes.html">click
      here</a> for the frameless version of this web site.
    </body>
  </noframes>
</frameset>
</html>
```

这个程序清单包含一个<frameset>标签而没有<body>标签。通常，可用于<body>标签内的标签都不能用于<frameset>标签内。在这个例子中，<frameset>标签包含一个 rows 属性，意思是框架应上下相邻，就像表格中的行。如果要使框架左右相连，使用 cols 属性代替 rows 属性。

By the Way

**注意：**这个网页中使用的 DTD 不是本书一直使用的 XHTML 1.1 DTD。这是因为标准 XHTML 1.1 DTD 不支持框架。因此，要让使用框架的网页有效，必须使用 XHTML 1.0 Frameset DTD，这是专门为使用框架的网页设计的 DTD。

必须指定 rows 或 cols 的大小,这可使用像素值或浏览器窗口的百分比。也可用星号(“\*”)来表明框架应尽可能占据窗口的可用空间。如果有多个框架有\*值,剩余的空间将在这些框架之间平均分配。

在程序清单 16.1 中, <frameset rows="80,\*"> 表示将浏览器窗口垂直分为两框架。上面的框架高 80 像素,下面的框架占据剩余的窗口空间。上面的框架包含文档 banner.html (见程序清单 16.2), 下面的框架包含 greeting.html (见程序清单 16.3)。

**提示:** 在程序清单 16.1 的框架集后,我在<body>标签和</body>标签之间包含了完整的网页。注意,这个网页在图 16.1 或图 16.2 中根本没有显示出来。支持框架的所有网页浏览器都会忽略<noframes>标签和</noframes>标签之间的所有内容。

当今所有主要浏览器都支持框架,因此现在框架的兼容问题不像几年前那么重要了。即使这样,包含<noframes>标签是很容易的,应考虑到那些仍使用旧浏览器的用户。

**Did you  
know?**

#### 程序清单 16.2 banner.html 文档作为 Entropy Almanac 网页的导航栏

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Entropy Almanac</title>
  </head>

  <body style="background-image:url(back.gif)">
    <div>
      <a href="greeting.html" target="main"></a>
      <a href="news.html" target="main"></a>
      <a href="facts.html" target="main"></a>
      <a href="tales.html" target="main"></a>
    </div>
  </body>
</html>
```

#### 程序清单 16.3 greeting.html 文档作为 Entropy Almanac 网页内惟一的内容框架

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Entropy Almanac</title>
  </head>

  <body style="background-image:url(back.gif)">
    <div style="text-align:center">
      <br />
      <br />
      The Entropy Almanac is an eclectic compendium of philosophical
      illuminations, mathematical profundities, cutting-edge science, and
      high-tech tools. Choose a category above to begin your explorations.
    </div>
  </body>
</html>

```

### By the Way

**注意：**程序清单 16.2 和程序清单 16.3 使用 XHTML 1.0 Transitional DTD，这实际上是最常用的 XHTML DTD。虽然 XHTML 1.1 DTD 更新也更严格，但为使文档有效，使用框架时必须遵守 XHTML 1.0，因此在框架内显示的网页也应使用 XHTML 1.0。

因为不能预测用户浏览网页时窗口的大小，因此通常使用百分比来指定行和列的大小比使用像素更方便。例如，要使左边的框架占据浏览器窗口宽度的 20%，右边的框架占据余下的 80%，可用下列的代码：

```
<frameset cols="20%,80%">
```

这条规则有一个例外，就是当框架包含特定大小的图像时。这时可用像素来指定大小，再加上几个像素用于边距和框架边框。程序清单 16.1 就是这样做的，在这个程序清单中，上面的框架中的所有图像都是 42 像素高，我将另外 38 像素用于边距和边框，这样整个框架高 80 像素。

用像素来指定框架大小时，框架集中应至少有一框架的宽度是可变的（使用\*），以确保文档可适应任意窗口大小。

## 16.2.2 添加框架

在<frameset>标签和</frameset>标签之间，应包含<frame />标签，指定每个框架要显示哪个 HTML 文档。如果<frame />标签的数量比<frameset>标签内定义的框架少，剩下的框架将为空。

在每个<frame>标签包含 src 属性，指明要在该框架中加载的网页地址。如果框架只包含一个图像，可使用图像文件的地址。

在框架内可以包含任何 HTML 或 XHTML 网页。然而，对于较小的框架，考虑到每框架显示区域的限制，最好能够创建专门用于框架的文档。例如，图 16.1 上面的框架（其代码见

程序清单 16.2) 比大多数网页都要简短得多, 因为它是专门为 80 像素高的框架设计的。

**注意:** 你可能注意到了, 在程序清单 16.2 所示的 banner.html 文档中, `<a>` 和 `<img/>` 标签的排列有点奇怪。因为不希望图像之间出现任何孔隙, 必须保证这些标签之间不出现任何空格或分行。因此, 将所有分行都放在标签内, 这使这段 HTML 代码较难阅读, 但使图像在网页上紧密排列在一起。

**By the Way**

## 16.3 在框架和窗口之间链接

在 `<frame />` 标签中, 使用 `name` 属性赋予框架惟一的名称后, 框架真正的威力便开始显示出来。然后, 可以使用 `<a>` 标签的 `target` 属性, 使网页中的链接改变框架的内容。例如, 程序清单 16.1 包含下列标签:

```
<frame src="greeting.html" name="main" />
```

**注意:** 从技术上说, `name` 属性已经过时了, 已被 `id` 属性取代。然而, 现在的网页浏览器在识别目标框架时仍然依赖于 `name` 而不是 `id`。因此, 现在需要使用 `name` 属性来标识框架。当然, 两个属性都用也可以。

**By the Way**

这段代码在框架中显示 `greeting.html` 页面, 并将该框架命名为 `main`。

在上面一个框架的代码中——如程序清单 16.2 所示, 可以看到下列链接:

```
<a href="news.html" target="main"></a>
```

用户单击这个链接时, `news.html` 将显示在名为 `main` 的框架 (下面的框架) 中。要在框架出现前实现这种交互, 必须使用复杂的编程语言或脚本语言。现在只要一个简单的链接就可以实现。

如果没有 `target="main"` 属性, `news.html` 页面将显示在当前 (上面的) 框架中。

为节省空间, 我没有提供 `news.html` 页面的程序清单; 它只是一个普通网页, 没有与框架相关的特性。从图 16.2 可以看到该网页, 在 <http://www.sampublishing.com/> 上可以看到整个框架集文件。

**注意:** 要在新窗口中打开网页而不使用框架, 应怎么做? 只要将下列专用名称之一用于 `<a>` 标签的 `target` 属性 (例如 `<a href="popup.html" target="_blank">Click here to open the popup.html document in a new window.</a>`):

- `_blank`: 将链接加载到一个新的未命名窗口。
- `_top`: 将链接加载到整个浏览器窗口。在需要删除所有框架或用一组新框架代替整个窗口时使用该名称。
- `_parent`: 如果当前框架嵌在其他框架中, 则将链接加载到父框架中。除非框架的嵌套深度超过一级, 否则该名称与 `_top` 的作用相同。

**By the Way**



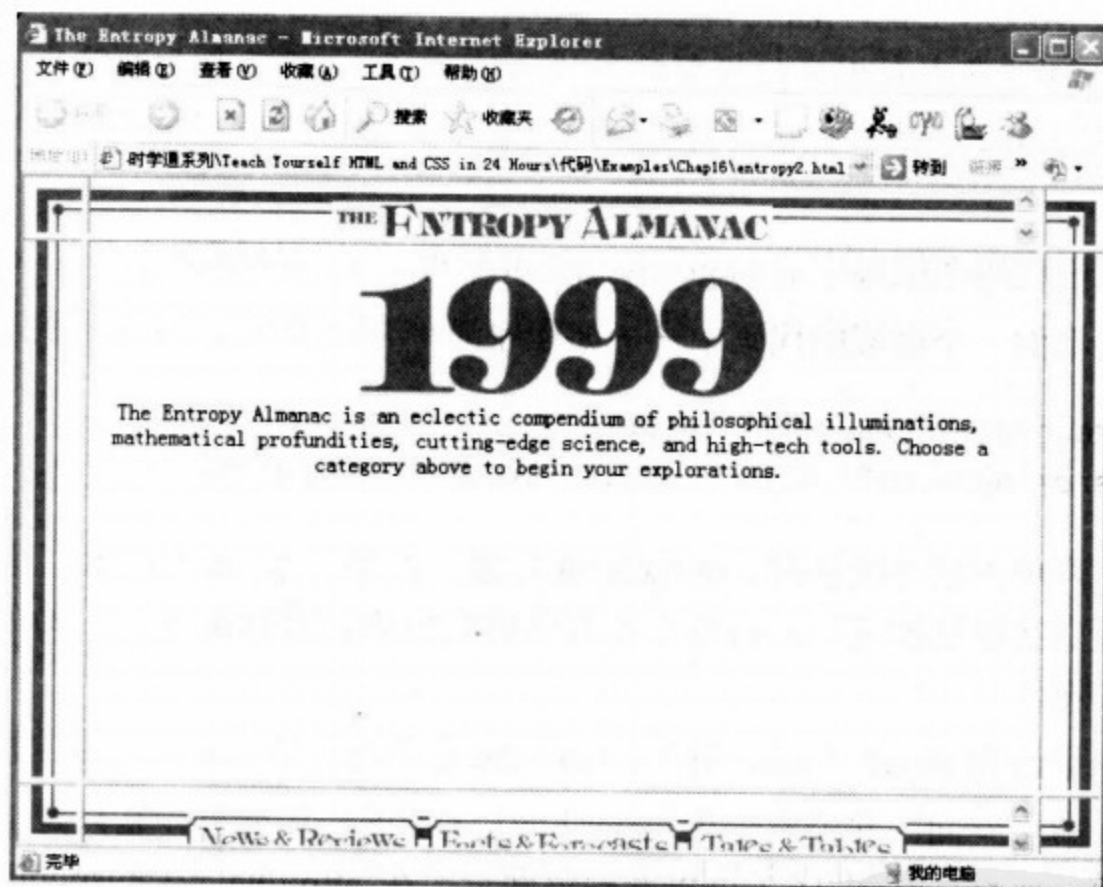
➤ `_self`: 将链接加载到当前框架中, 替换该框架当前显示的文档。你可能从不使用该名称, 因为它与完全不指定 `target` 属性的作用相同。在第 3 章, 读者知道 XHTML 1.1 放弃了使用 `target` 属性在新浏览器窗口中打开网页。在那一章, 读者还学习了如何使用 JavaScript 来避开这限制。在 `news.html` 中无需采用这种方法, 因为它使用的是 XHTML 1.0 Transitional DTD, 该 DTD 仍支持 `target`。

## 16.4 在框架中嵌套框架

通过将框架嵌套到另一个框架中, 可以创建复杂的框架布局。例如, 图 16.3 和程序清单 16.4 所示的文档共有 9 框架。使用 `cols` 的框架集将使用 `rows` 的框架集的每一行分为三部分。你可能认为我在框架的复杂性方面自相矛盾, 然而, 这个例子旨在说明如何嵌套框架, 而不是鼓励这样的用法。

图 16.3

这个窗口包含 9 框架, 其中一些框架是只有平铺背景图像的空白页面



程序清单 16.4 在每个垂直框架中使用三个水平框架创建图 16.3 所示的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Entropy Almanac</title>
  </head>

  <frameset rows="43,*,43">
```

```
<frameset cols="43*,43">
  <frame src="ctoplft.html" name="toplft" />
  <frame src="bordtop.html" name="top" />
  <frame src="ctoprgt.html" name="toprgt" />
</frameset>
<frameset cols="43*,43">
  <frame src="bordlft.html" name="left" />
  <frame src="main.html" name="main" />
  <frame src="bordrgt.html" name="right" />
</frameset>
<frameset cols="43*,43">
  <frame src="cbtmlft.html" name="btmlft" />
  <frame src="bordbtm.html" name="btm" />
  <frame src="cbtmrgt.html" name="btmrgt" />
</frameset>
</frameset>
</html>
```

图 16.3 包含 9 个不同的网页，分别位于 9 个框架中，其中许多框架只包含背景图像，起修饰作用。边上和角上的框架包含空白 HTML 文档，只显示了专门设计的背景图像。顶部的框架是永久性的标题图像，底部的框架是与前一个例子类似的导航栏。最后的效果是，用“相框”边框包围中间的框架。图 16.4 显示了所有背景图像及网页包含的其他图像的缩略图。

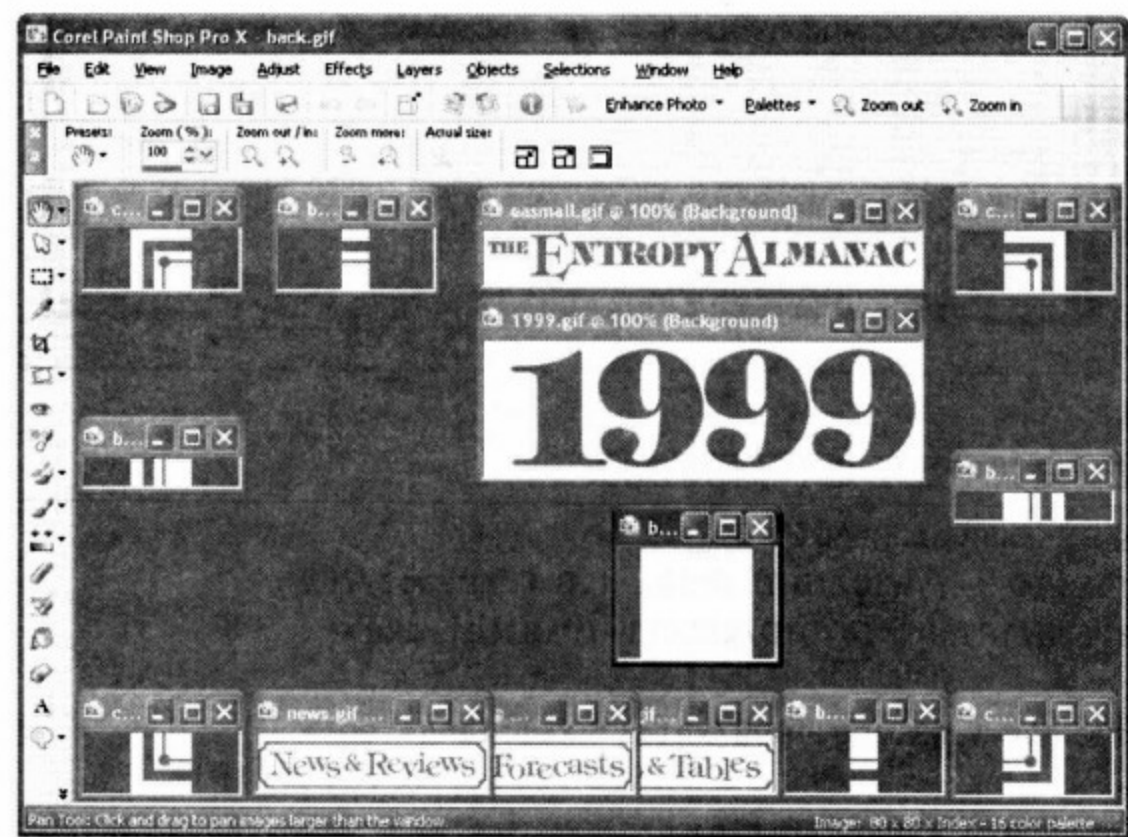


图 16.4  
为创建图 16.3 所示的边框效果，我设计了几个自定义的背景图像及与之匹配的标题图像

## 16.5 控制框架的边距、边框和滚动

图 16.3 中 9 个框架的排列有个问题，那就是网页很难看，且所有滚动条除惹人讨厌外没有任何实际用处。下面将解决这个问题。

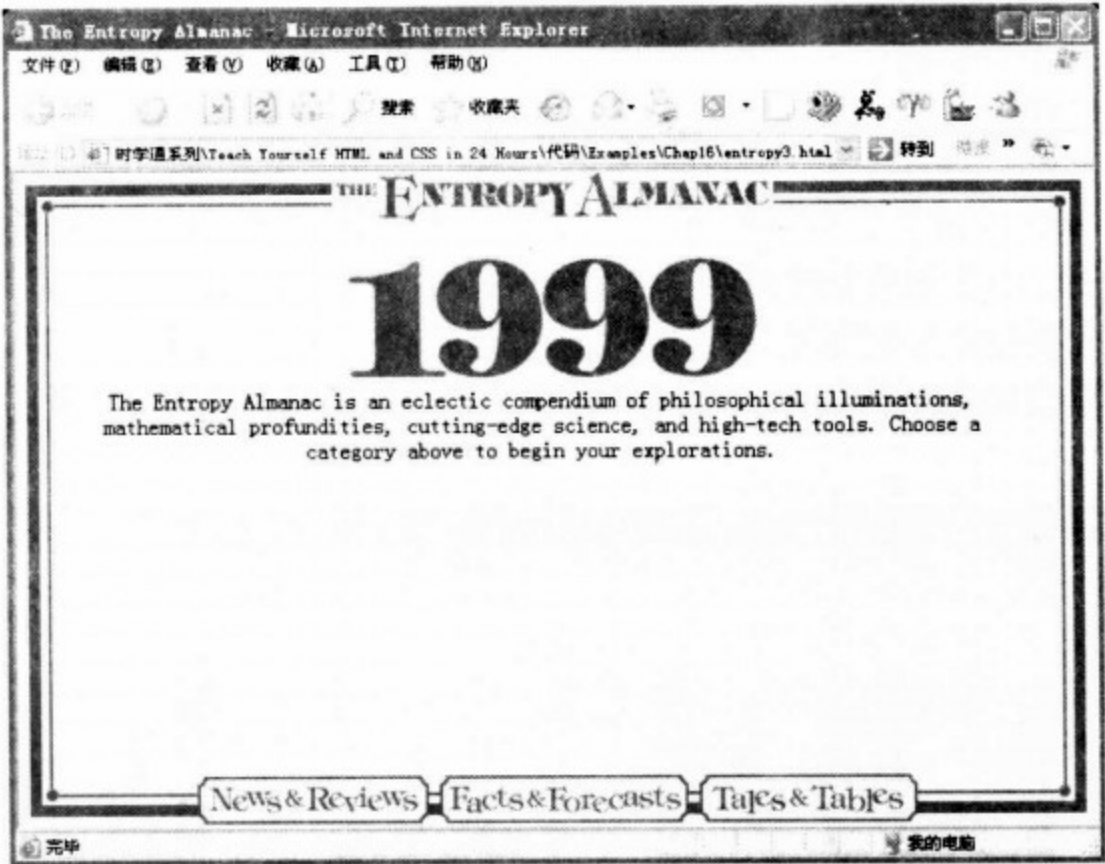
难看的地方在于框架之间的灰色分隔线，它们完全破坏了包围中间框架的优美图像的效果。另外，顶部和底部的框架如果没有滚动条，将没有足够的空间来显示图像。幸运的是，有 HTML 命令可以删除框架的分隔线；通过缩小边距可使小框架有更多的空间；还可以使框架不显示滚动条。

Coffee Break

休息一下：要看现实中利用框架的网页使用滚动条的情况，可访问 Clint Eastwood 网站 <http://www.clinteastwood.net/>。这个网站大量使用框架，让用户能够在网页的有限区域内导航和滚动网页内容。

学习这些 HTML 技巧前，先来看看这些技巧可以实现的动人效果。图 16.5 是图 16.3 的另一个版本，在包含 9 个框架的窗口中显示同样的网页。显然，图 16.5 看起来效果更好！在程序清单 16.5 中，可以看到对图 16.3 难看的地方所做的修改。

图 16.5  
这是程序清单 16.5 所示的网页。通过在 <frame /> 标签中添加几个属性，就使得这个网页看起来更美观



程序清单 16.5 在 9 框架中显示 9 个不同的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>The Entropy Almanac</title>
  </head>

  <frameset rows="43,*,43" border="0">
    <frameset cols="43,*,43">
      <frame src="ctoplft.html" name="toplft" scrolling="no" frameborder="0" />
      <frame src="bordtop.html" name="top" scrolling="no" frameborder="0"
```



```

    marginheight="1" />
    <frame src="ctoprgt.html" name="toprgt" scrolling="no" frameborder="0" />
</frameset>
<frameset cols="43,*,43">
    <frame src="bordlft.html" name="left" scrolling="no" frameborder="0" />
    <frame src="main.html" name="main" frameborder="0" />
    <frame src="bordrgt.html" name="right" scrolling="no" frameborder="0" />
</frameset>
<frameset cols="43,*,43">
    <frame src="cbtmlft.html" name="btmlft" scrolling="no" frameborder="0" />
    <frame src="bordbtm.html" name="btm" scrolling="no" frameborder="0"
    marginheight="1" />
    <frame src="cbtmrgt.html" name="btmrgt" scrolling="no" frameborder="0" />
</frameset>
</frameset>
</html>

```

**注意：**这段代码有个问题，那就是从技术上说，XHTML 不支持<frameset> 标签中的 border 属性。换句话说，由于使用了 border 属性，程序清单 16.5 中的文档即使用 XHTML 1.0 Transitional DTD 也是无效的。前面提到过，通常用 CSS 样式代替 border 属性来实现相同的效果，同时遵循 XHTML 标准。问题在于当前没有一个网页浏览器让你能够通过 CSS 设置 border="0"来实现相同的效果。因此，如果删除 border 属性，框架间将出现难看的灰色边框。因为我在实际使用中经常出错，因此我将美观排在有效性之前，浏览器最终将修改它们的方式。

**By the  
Way**

除 id 和 name 属性外，<frame /> 标签还可以使用下列与框架相关的属性。

- marginwidth: 框架的左、右边距，以像素为单位。
- marginheight: 框架的上、下边距，以像素为单位。
- scrolling: 框架是否显示滚动条（“yes”或“no”）。
- frameborder: 是否显示该框架与相邻框架之间的分隔线，1 表示是，0 表示否。
- noresize: 不允许用户调整框架的大小。

marginwidth 和 marginheight 的作用不言自明，但其他属性需要解释一下。

通常，大小不足以容纳其内容的框架将显示滚动条。在图 16.3 中的顶部和底部框架就是例子。如果希望框架不显示滚动条，可在<frame> 标签中加入 scrolling="no"；反过来，scrolling="yes"将在水平和垂直方向显示滚动条，而不管是否需要。

**提示：**当图像勉强显示在小框架中时，网页浏览器通常显示只能滚动几个像素的滚动条，没有实际用处。在这种情况下，通常可以设置 scrolling="no"不显示滚动条，而不是将框架设置得更大（让空边距占据宝贵的空间）。我能想到的使用 scrolling="yes"惟一情况是，除非保证滚动条总是显示，否则有些图像不能对齐。你很有可能从来不需要使用 scrolling="yes"。

**Did you  
know?**

浏览框架的用户可以像对待普通网页一样，使用鼠标拖曳到框架的边框，以调整框架的大小。如果不希望用户调整框架的大小，可在<frame />标签中加入 `nosize="nosize"`。

框架非常灵活，可以控制框架边框的大小或完全不显示边框。这使得框架文档看起来就像普通网页，不会有难看的边框分隔网页。为此，只需在每个<frame />标签中——而不是只在<frameset>中——加入属性 `frameborder="0"`。

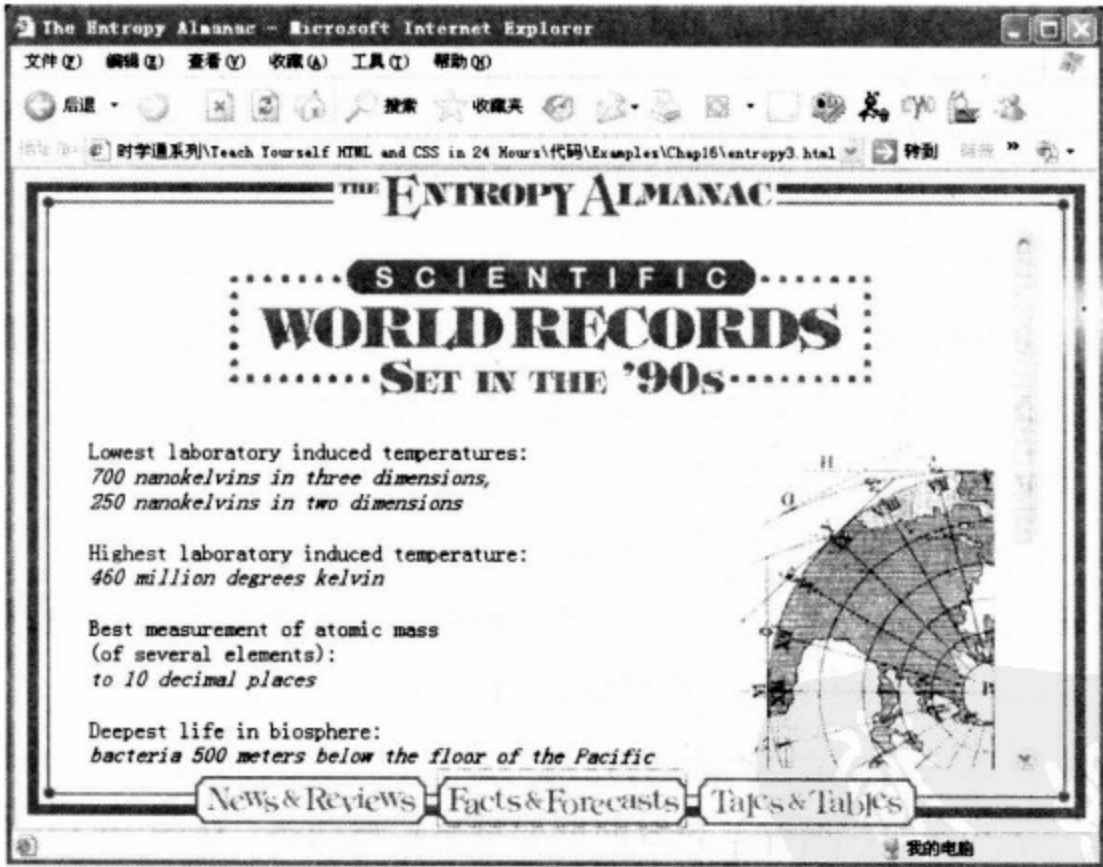
当与自定义图像一起使用时，无边框的框架让你能够创建更易于导航，用户更乐于访问的网站。例如，当有人访问图 16.5 所示的网站并单击底部框架中的导航按钮时，选择的页面将很快显示在中间的框架中，因为标题图像、导航按钮和边框的图像都保持不变。这些框架还能够自动适应浏览器窗口的大小，因此，“相框”效果在 1024×768 以及更高的分辨率下的显示与其在 800×600 分辨率下的显示一样。

Coffee Break

**休息一下：**很多网页设计者和用户不喜欢框架，这一点我没有讨论太多。争论框架是好是坏不在本章的范围。然而，要了解在网页中使用框架的优点和缺点，请参考 <http://www.yourhtmlsource.com/frames/goodorbad.html>。

图 16.6 显示了单击图 16.5 显示的 Facts & Forecasts 链接后的结果。注意，当内容太大不能放在中间的框架内时，中间的框架将显示滚动条。

图 16.6  
单击图 16.5 所示屏幕顶部的 Facts & Forecasts 链接后，中间的框架将显示一个新网页，其他框架没有重新绘制



By the Way

**注意：**要了解框架的高级特性，可以了解一下内联（inline）框架，这种框架可直接放到网页的文本区域中。内联框架用<iframe>标签来创建，如下列代码所示：

```
<html>
<head>
<title>bios</title>
```



```
</head>

<body>
  <h1>Short Bios</h1>
  <p>
    <iframe name="bioframe" src="mybio.html" width="200"
height="200">
    </iframe>
  </p>
  <p>
    <a href="yourbio.html" target="bioframe">Your Bio</a>
  </p>
  <p>
    <a href="mybio.html" target="bioframe">My Bio</a>
  </p>
</body>
</html>
```

这段代码将页面 mybio.html 显示在标题 Short Bios 下方一个 200×200 像素的区域中。如果文档 mybio.html 在这个小型区域内显示不下，将显示滚动条。单击链接 Your Bio，yourbio.html 将替换 200×200 像素区域中的内容。单击 My Bio，mybio.html 将重新显示在该区域中。

在 <http://www.sampublishing.com/> 上可以看到一个在线的<iframe>例子。

## 16.6 总结

本章读者学习了如何通过将网页浏览器窗口划分为框架，从而同时显示多个页面。读者学习了使用框架集文档来定义框架的大小与排列方式，以及每个框架要载入的网页或图像；学习了如何创建链接来改变选定框架的内容，同时保持其他框架不变；还了解到一些可选设置，它们可以控制框架的可调整边框和滚动条；最后，学习了如何嵌套框架集以创建复杂的框架布局。

表 16.1 总结了本章介绍的标签与属性。

表 16.1 第 16 章介绍的标签与属性	
标签/属性	功能
<frameset>..</frameset>	将主窗口划分成一组可以显示不同文档的框架。
属性	
row="行数"	将窗口或框架集纵向地分为数行，可用数字（如 7）、窗口宽度的百分比（如 25%）或星号（*）来指定。星号表示这个框架占据剩余的所有空间或平均分配这些空间（如果指定了多个*框架）。
cols="列数"	效果与 rows 相似，只是将窗口或框架集横向地分为数列。
frameborder="yes/no"	指定框架是否显示边框，选项有 1（是）与 0（否）。
<frame />	在<frameset>中定义框架。
属性	
src="url"	要在框架中显示的文档的 URL。

续表

标签/属性	功能
id="名称"	在<a href>链接的 target 属性中指向这个框架时使用的名称。符合 XHTML 标准，但在当前网络浏览器对框架的支持中这个属性并不太受支持。
name="名称"	在<a href>链接的 target 属性中指向这个框架时使用的名称。最终将被 id 取代，但现在仍然在当前网络浏览器中使用。
<marginwidth>	框架内文档左、右两边留出的间距，单位为像素。
<marginheight>	框架内文档上、下两边留出的间距，单位为像素。
scrolling="yes/no/auto"	决定一个框架是否有滚动条，可能的值有 yes、no 和 auto。
noresize="noresize"	防止用户使用鼠标调整框架（和邻近框架）的大小。
<noframes>...</noframes>	针对不支持框架的浏览器，在<frameset>文档中提供一个替代文档主体（通常包含<body> ...</body>）。
<iframe>...</iframe>	创建内联框架，<iframe>接受所有<frame />接受的属性。

## 16.7 问与答

问：我是否可以在框架中显示来自 Internet 的其他人的网页，同时在另一个框架中显示我自己的页面？如果那些网站也使用框架，将发生什么？

答：可以在框架中加载 Internet（或 Intranet）中的任何文档。如果这个文档是框架集，其框架将调整大小以适应加载它的框架。

例如，可以在框架中放入最喜爱的链接列表，然后让这些链接引用的网页显示在单独的框架中。这样可以很容易地提供到其他网站的链接，且不会使用户迷失而无法返回你的网站。但要记住，如果那个网站中的任何链接有 target="\_top"，它将替换你所有的框架。

另外，将别人的网页作为框架将导致它们看起来是你网站的一部分，这将给你带来法律方面的麻烦，很多诉讼正是因为这个问题而发生的。所以，如果将别人的网页放在你的框架中，要确保得到了网页的制作者的书面许可，就像将他们的网站中的图像或文字放在你的网页中一样。

问：需要在所有框架中放入<title>吗？如果这样做，哪个标题将显示在窗口顶部？

答：只有框架集文档的标题会显示。框架集文档不要求<head>和<title>这两个标签，但所有页面都有标题是个好主意，以防有人在框架外打开其中一个页面。

## 16.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 16.8.1 测验

1. 编写 HTML 代码，在一个占浏览器窗口左边 25%的框架中列出名字 Mickey、Minnie 和 Donald，并使得用户单击每个名字时，占浏览器窗口右边 75%的框架中将出现相应的网页。
2. 编写框架集文档来制作如图 16.7 所示的框架布局：



图 16.7

## 16.8.2 答案

1. 需要五个不同的 HTML 文档。第一个文档是框架集：

```
<html>
  <head>
    <title>Our Friends</title>
  </head>

  <frameset cols="25%,75%">
    <frame src="index.html" />
    <frame src="mickey.html" name="mainframe" />
  </frameset>
</html>
```

然后，要将 index.html 文档用于左边的框架：

```
<html>
  <head>
    <title>Our Friends Index</title>
  </head>

  <body>
    <p>
      Pick a friend:
    </p>
    <p>
      <a href="mickey.html" target="mainframe">Mickey</a><br />
    </p>
  </body>
</html>
```

```

        <a href="minnie.html" target="mainframe">Minnie</a><br />
        <a href="donald.html" target="mainframe">Donald</a>
    </p>
</body>
</html>

```

最后，需要三个 HTML 页面，其名称分别为 mickey.html、minnie.html 和 donald.html。它们包含各个朋友的信息。

2.

```

<html>
  <head>
    <title>Nested Frames</title>
  </head>

  <frameset rows="*,*">
    <frameset cols="*,*,*">
      <frame src="top1.html" />
      <frame src="top2.html" />
      <frame src="top3.html" />
    </frameset>
    <frameset cols="*,*">
      <frame src="bottom1.html" />
      <frame src="bottom2.html" />
    </frameset>
  </frameset>
</html>

```

## 16.9 练习

在第 23 章，读者将学习如何使一个页面在指定的时间后自动地载入另一个页面。通过将这个技巧与框架集结合使用，可以创建各种有趣的动画布局效果。考虑你将这种方式显示什么信息。



# 第四部分

## 动态网页

第 17 章 面向非程序员的网页脚本编程

第 18 章 使用 HTML 表单收集信息

第 19 章 在网页中嵌入多媒体

第 20 章 使用 HTML 和 CSS 使 eBay 拍卖更有趣

第 21 章 创建自己的博客







## 第 17 章

# 面向非程序员的网页脚本编程

脚本是计算机编程的一种文雅说法，这是一个很广泛的话题，不可能在一章内学到很多。然而，你仍然可以很快地学习关于脚本的一些知识，当然还有许多是无法做到的。因此，本章的目标是教你学习足够的网页编程，使你的网页从那些不同于没有使用脚本的网页。

具体地说，本章将学习如何使网页中的图像（或多媒体对象）根据鼠标的移动或单击而改变，以及如何将在不同广告图像之间循环的动画条幅广告添加到网页中，从而开展广告业务。还将学习如何显示随机格言（quote）、创建虚拟的鱼塘以及利用 Google Ad 来赚钱。是的，本章就像将一本书压缩到一章！通过使用 JavaScript 语言，可用一种与所有主要网页浏览器都兼容的方式来实现这些任务。

### 17.1 必须了解的脚本知识

网页脚本让你能够访问网页中样式和元素的内容。要开发自己的脚本，需要使用脚本语言，如 JavaScript 或 VBScript。这里将提供一些读者可在自己的网页中使用的脚本，而不是教你学习脚本语言——这显然超出了本章的范围。脚本编程学习起来并不是很难，但最好等熟悉 HTML 和 CSS 后再学。

Web 中使用的主要脚本语言是 JavaScript。本章开发的所有脚本都是 JavaScript 脚本，但读者不要太关心每个脚本的细节；我将指出重要的地方，让你知道它们如何发挥作用。你可能对有些脚本之简单感到惊讶。如果有 Java 或 C++ 编程语言的经验，将觉得 JavaScript 有些熟悉。即使没有任何编程经验，也没什么问题，只要复制并粘贴代码，将已有脚本加到自己的网页中。

你可能已经猜到了，要将脚本添加到网页中需要使用特殊的 HTML 标签。<script> 标签用于封装脚本代码。一些老式网页浏览器不支持脚本，因此将脚本代码放在 <script> 标签中时有一个技巧：将脚本代码放在 HTML 注释中，如下面的例子所示：

```
<script type="text/javascript">  
  <!-- Hide the script from old browsers  
  alert( "Hello!" );  
  // Stop hiding the script -->  
</script>
```

在这个简短的脚本中，在单行脚本代码前使用了标记注释的<!--。脚本代码后是表示注释结束的-->。这段脚本代码显示警告消息“Hello!”，如图 17.1 所示。

图 17.1

Hello 脚本在对话框中显示问候消息



这个 Hello 网页演示了如何创建显示警告消息的最小脚本。

脚本提供交互的主要方式是对用户操作进行响应。例如，如果用户单击鼠标或按下键盘上的键，脚本可能对此进行响应，改变网页中文本或图像的外观。用户交互（如鼠标单击或按键）称为事件，脚本根据事件采取措施的过程称为事件处理。使用特殊属性将事件处理脚本代码与网页中的元素关联起来。

下面是 JavaScript 中一些常用的事件属性，同时描述了网页元素何时产生这些事件。

- **onload**: 浏览器加载元素。
- **onkeydown**: 用户按下键。
- **onkeyup**: 用户松开键。
- **onclick**: 用户单击元素。
- **ondblclick**: 用户双击元素。
- **onmousedown**: 在鼠标指向元素时按下鼠标上的任何按钮。
- **onmouseup**: 在鼠标指向元素时按下鼠标上的任何按钮。
- **onmouseover**: 用户将鼠标移到元素的边界内。

- onmousemove: 鼠标在元素上移动。
- onmouseout: 用户将鼠标移出元素的边界。

可以看到，事件属性用于对常见的用户输入事件，如鼠标单击或按键进行响应。可将脚本代码指定给事件属性，从而将脚本代码同事件关联起来，如下所示：

```
<h1 onclick="this.style.color = 'red';">I turn red when clicked.</h1>
```

在这个例子中，脚本代码分配给<h1>标签的 onclick 事件属性，这意味着当用户在该文本上单击鼠标左键时这段代码将运行，对该事件进行响应。这段脚本代码的响应方式是将文本颜色改为红色。根据鼠标单击改变文本颜色，就为普通文本增加了交互式。这是许多交互脚本的工作原理。

## 17.2 使用交互式加亮使网页更有趣

如果你用过图形 CD-ROM 软件或 DVD 菜单，可能见过这样的按钮：当鼠标经过按钮时按钮将加亮或改变。这看起来很酷，且可在用户单击前提供视觉反馈，研究表明，这样可以减少用户的困惑或操作错误：它使用户界面更直观。

**注意：**将鼠标指向时加亮的图形链接有时也称为悬放按钮（hover button）。

**By the Way**

你也可以在自己的网页中为链接添加这种视觉反馈。要实现这种效果，第一步是为静态的（暗淡的）图标和悬放图标（加亮的）创建图像。图 17.2 显示了我为 Stalefish Labs 公司开发网站时创建的一些图标。每个图标都有两个版本：一个是暗淡的，一个是加亮的——仿佛四周有绿色阴影。

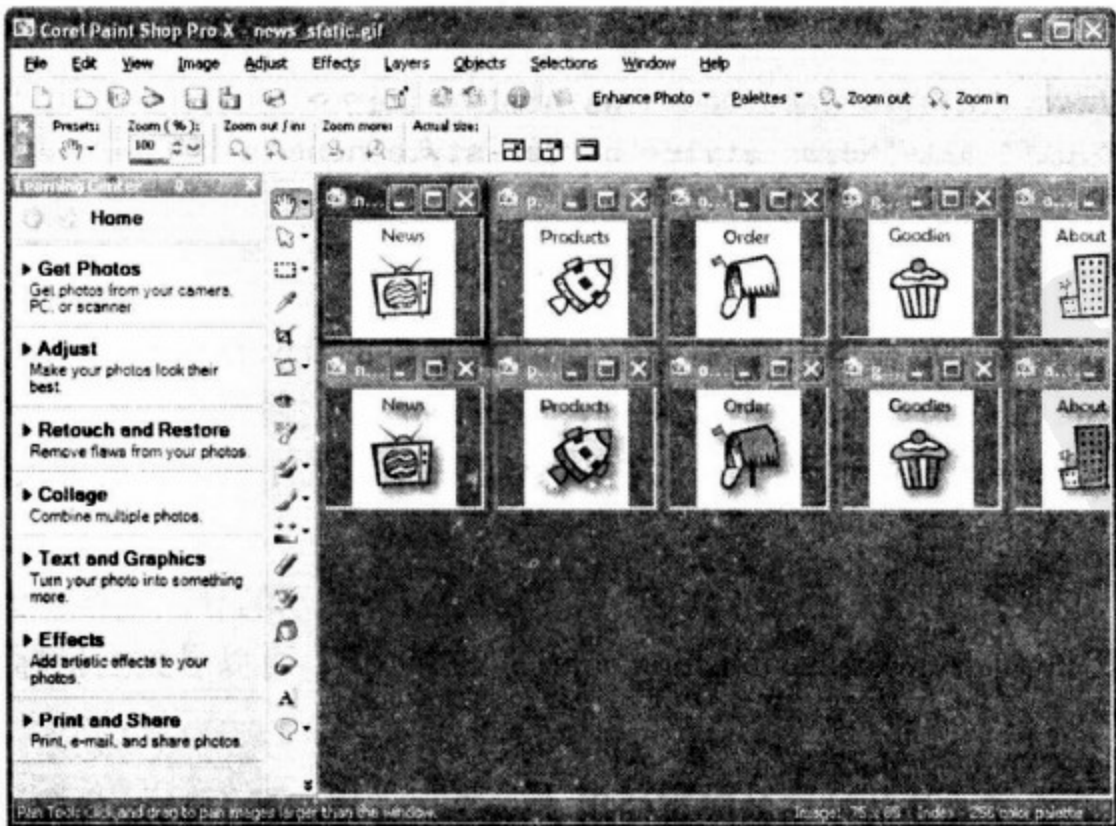


图 17.2  
五个图像，每个图像都有对应的加亮版本，当鼠标指向图像时加亮版本将取代原始图像

你是否要使一些网页看起来更酷，或者当鼠标经过导航图标或其他图像时它们发生改变以便易于理解？如果是，在阅读下面几段时可以创建图像的加亮版本，并修改自己的网页。下面是一些技巧。

- 使用 Paint Shop Pro 的文本工具来创建鼠标指向时改变颜色的图形标题。
- 使用 Paint Shop Pro 中的“Effect/3D Effect/Buttonize”命令，用各种背景颜色来创建在单击前加亮的按钮。
- 使用第 7 章所学的方法创建当鼠标在上面经过可以旋转、摇摆或闪烁的图标。当鼠标没有指向图标时，可以使用普通的非动画 GIF 图像。
- 如果有一个选项列表，在每个选项前放一幅空白（完全透明的）图像，然后当鼠标指向选项时在选项前显示箭头或项目符号。

下面是添加任何脚本前图形链接的 HTML 代码：

```
<a href="http://www.stalefishlabs.com/news.html">
</a>
```

你对这段代码应很熟悉；如果还不熟悉，复习一下第 8 章。

只要在网页的链接中包含两个特殊属性 `onmouseover` 和 `onmouseout`，就能创建交互式悬放功能。`onmouseover` 指示网页浏览器当鼠标经过链接内的文本或图像时应如何做；`onmouseout` 指示网页浏览器当鼠标离开链接区域时应如何做。

在这个例子中，你希望当鼠标经过链接时，将图像变为 `news_hover.gif`，然后当鼠标离开时又变回为 `news_static.gif`。下面是使用的 HTML 和 JavaScript：

```
<a href="http://www.stalefishlabs.com/news.html"></a>
```

注意，图像文件的文件名要放在单引号内，指定给事件属性的 JavaScript 代码要放在双引号内。这是因为 JavaScript 代码是作为属性提供的，而 HTML 中的属性必须包含在双引号中。在自己的网页中输入 JavaScript 代码时，一定要严格遵循这里的示例，只需替换图像名称和图像文件。

程序清单 17.1 是一个网页的完整 HTML 代码，它使用图 17.2 所示的导航图标作为链接。在图 17.3 和图 17.4（或 <http://www.samspublishing.com/> 的文件）中，可以看到当鼠标经过图标时图标是如何加亮的。

#### 程序清单 17.1 包含交互式悬放按钮的网页使用 JavaScript 增强的 HTML 代码

```
<?xml version="1.0" encoding="UTF-8"?>
```



```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Hover Buttons</title>
  </head>

  <body>
    <h1>Hover Buttons</h1>
    <p>
      Try out these hover "buttons" to see how JavaScript makes hovering just a
      bit more fun.
    </p>
    <hr />
    <p>
      <a href="http://www.stalefishlabs.com/news.html"></a>
      <a href="http://www.stalefishlabs.com/products.html"></a>
      <a href="http://www.stalefishlabs.com/order.html"></a>
      <a href="http://www.stalefishlabs.com/goodies.html"></a>
      <a href="http://www.stalefishlabs.com/about.html"></a>
    </p>
  </body>
</html>

```

**提示:** 属性 onmouseover 和 onmouseout 也可用于图像映射(第 10 章介绍过)。Sams Publishing 网站 <http://www.sampublishing.com/> 上的导航时钟就是一个大型交互式图像映射的例子, 请将鼠标在导航时钟上移动。看看源文件, 就知道如何将 JavaScript 命令包含到图像映射中。另外, 不要忘记 GIF 动画同脚本结合使用!

**Did you  
know?**

图 17.3  
鼠标经过 News 图标时，电视图标将加亮，同时显示替代文本

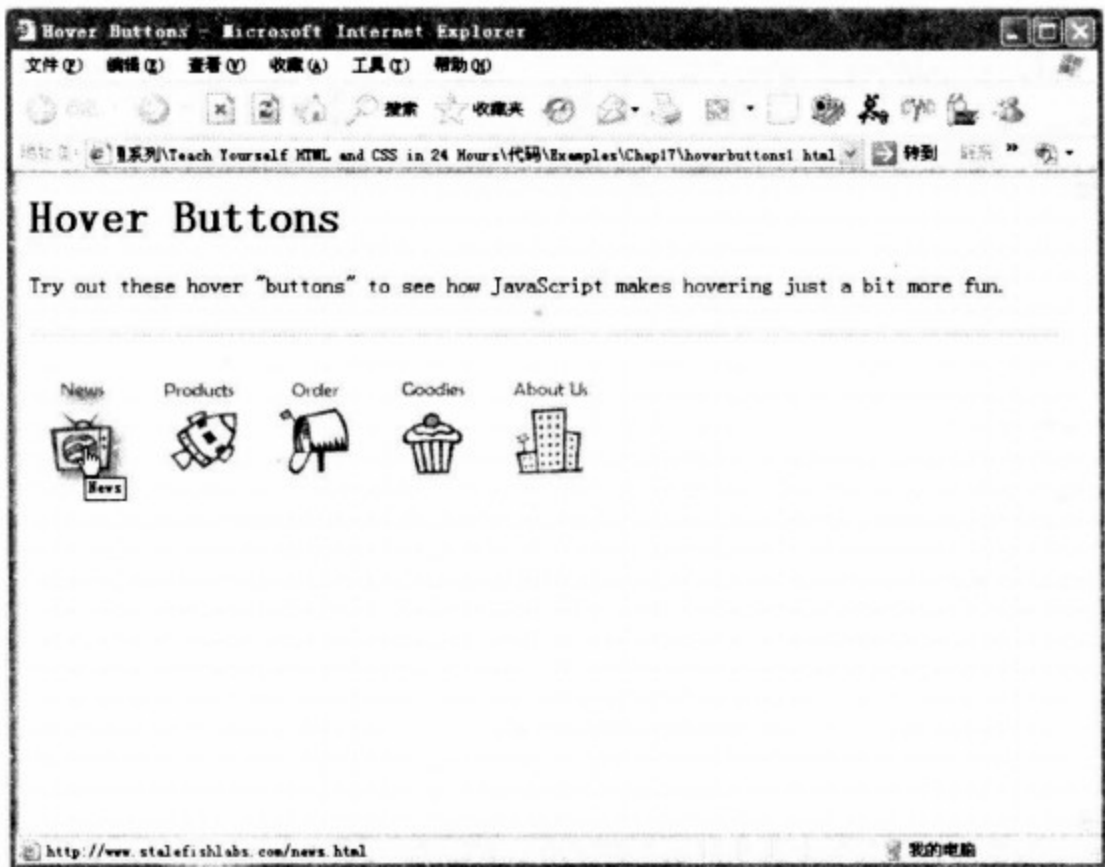


图 17.4  
鼠标指向 Goodies 图标时，加亮的是纸托蛋糕图标而不是电视图标



事件属性的 JavaScript 代码可执行多个操作。例如，可以修改多幅图像，图像之间分号隔开。

### 17.3 预加载图像以提高速度

程序清单 17.1 中的代码在所有主要的网页浏览器中都可用，但有一个小问题：仅当用户将鼠标移动到图像上时才从网站下载加亮的图像。这样有时在显示加亮的图像前有很大的延迟，特别是当浏览网页的用户的 Internet 连接速度很慢时。结果可能降低所有重要的 Gee Whiz

Factor (GWF)。

为避免这种情况，可以包含一些 JavaScript 代码，告诉浏览器在网页显示时尽快预加载图像。这样，等到反应较慢的用户将鼠标移到链接上时，图像已经加载好可以在屏幕上弹出。这使得动画的显示看起来没有下载延迟，使页面有更快的感觉，并提高 GWF。程序清单 17.2 列出了实现这种功能的脚本代码。

程序清单 17.2 预加载图 17.2 所示图像的脚本代码

---

```
<script type="text/javascript">
  !-- Hide the script from old browsers
  var image1 = new Image();
  image1.src = "news_static.gif";
  var image2 = new Image();
  image2.src = "news_hover.gif";
  var image3 = new Image();
  image3.src = "products_static.gif";
  var image4 = new Image();
  image4.src = "products_hover.gif";
  var image5 = new Image();
  image5.src = "order_static.gif";
  var image6 = new Image();
  image6.src = "order_hover.gif";
  var image7 = new Image();
  image7.src = "goodies_static.gif";
  var image8 = new Image();
  image8.src = "goodies_hover.gif";
  var image9 = new Image();
  image9.src = "about_static.gif";
  var image10 = new Image();
  image10.src = "about_hover.gif";
  // Stop hiding the script -->
</script>
```

---

这个程序清单有几个地方需要注意。最重要的是<script>标签，它可以放在文档的<head>或<body>部分中。然而，如果脚本要在页面首次加载时运行，就像在这个例子那样，大多数 Web 开发人员会将<script>放在页面的 head 部分。

前面指出过，标签<!--和-->实际上是<script>标签和</script>标签之间的注释标签，它们的作用是对老式浏览器隐藏脚本，否则老式浏览器将不能理解这些脚本，进而将代码作为文本显示在网页中。

这里不详细解释该程序清单中的 JavaScript 代码，因为这足以成为一门计算机编程课程。要将这些代码复制到自己的网页中，并使用自己的图像名称和图像文件，并不需要完全理解这些代码的工作原理。要注意的主要地方是两行预加载图像的代码。第一行创建一个空白图像，第二行将这个图像设置为一个图像文件以加载它。可根据需要用这种方法预加载任意多的图像。

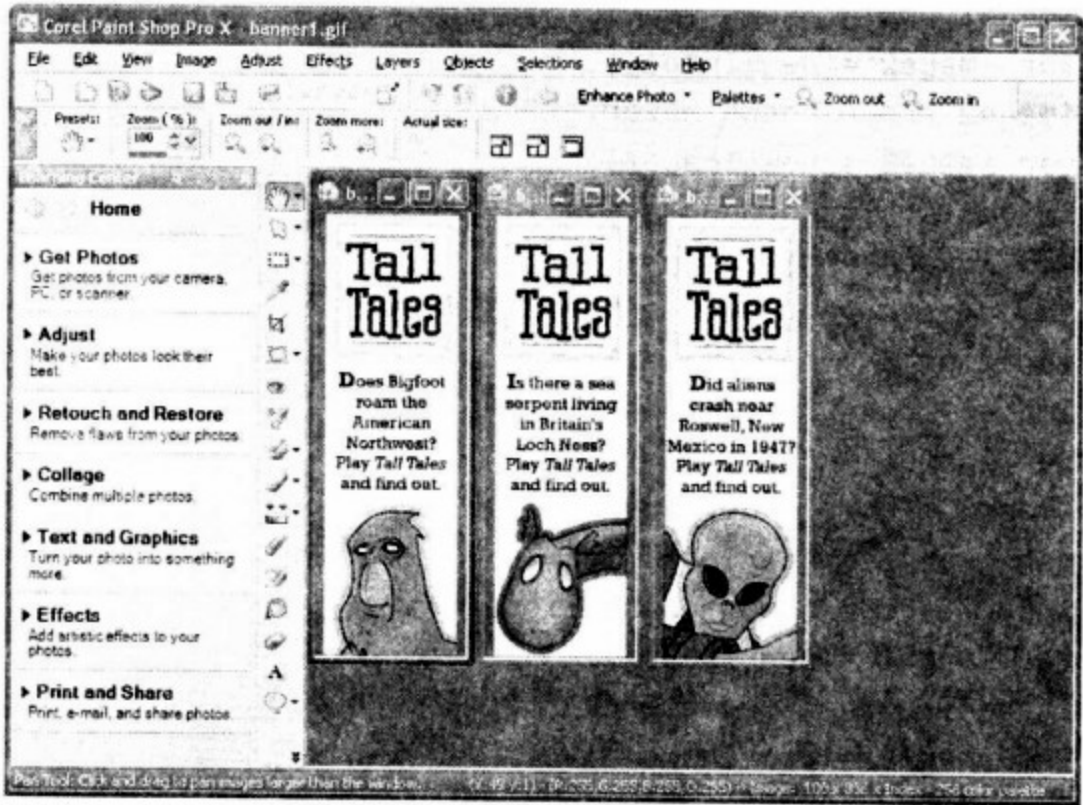
## 17.4 创建动画条幅广告

正如本章前面指出的，脚本最常见的用途之一是使图像变为动画。虽然悬放按钮和其他形式的用户界面改进很有用，但使用脚本将图像变为动画是一种更好的方法，我指的是使用动画条幅广告不断地显示广告。你肯定在 Web 上看到过这种广告，因为现在广告在所有大型网站上都非常普遍。

制作动画条幅广告的第一步是创建各个广告图像。条幅广告的大小和形状各异，但我个人更喜欢占据网页旁边空间的垂直广告。图 17.5 显示了三个垂直广告。

图 17.5

动画条幅广告由大小相同的一系列图像组成



上图中的三个图像依次显示，形成动画条幅广告。确保广告正常显示的关键在于创建定时机制，使每幅广告图像都显示一段时间，然后显示下一幅。这用 JavaScript 代码很容易实现，如程序清单 17.3 所示。

程序清单 17.3 显示一系列准确定时的条幅广告图像的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Tall Tales Banner Ad</title>

    <script type="text/javascript">
      <!--
      var bannerNum = 1;

      function rotateBanner() {
```



```

    if (++bannerNum > 3)
        bannerNum = 1;
    document.getElementById("ttbanner").src = "banner" + bannerNum +
        ".gif";
    window.setTimeout('rotateBanner();', 3000);
}
//-->
</script>
</head>
<body onload="window.setTimeout('rotateBanner();', 3000);">
    <h1 style="text-align:center">Game Lovers Anonymous</h1>
    <p>
        <a href="http://www.talltalesgame.com/">
            
        </a>
        Welcome to Game Lovers Anonymous. We love to play
        games, and invite you to support us by visiting our
        sponsors as you enjoy our site.
    </p>
</body>
</html>

```

要在自己的网页中有效地使用这段 JavaScript 代码，也不必完全理解它。有些地方需要解释一下，以便读者能够知道如何根据自己的需要修改代码，但除此之外读者不必太关心其工作原理。这里的目的是在网页中加入一些有用的代码，而不是一夜之间成为程序员。

图 17.6 显示了一个条幅广告网页。虽然印刷品不是最好的说明方式，但从中可以知道网页在不断显示不同的条幅广告图像。



图 17.6

程序清单 17.3 中的 JavaScript 代码生成在三幅不同广告图像间循环的条幅广告



**By the Way**

**注意：**第 7 章介绍了如何通过创建在三幅广告图像之间循环的 GIF 动画图像来创建类似的动画条幅广告效果。这种方法当然也可行，且由于淡出效果实际上能够提供更平滑的动画。然而，GIF 动画比 JavaScript 版本使用的三幅 GIF 图像大；另外，条幅广告的 JavaScript 版本更易于修改以容纳更多广告或修改已有的广告。

为创建自己的条幅广告，需要仔细研究一下程序清单 17.3 中的几行代码。首先看一下实际改变（轮换）广告图像的 JavaScript 函数：

```
function rotateBanner() {
    if (++bannerNum > 3)
        bannerNum = 1;
    document.getElementById("ttbanner").src = "banner" + bannerNum + ".gif";
    window.setTimeout('rotateBanner();',
    }
```

rotateBanner() 函数每隔 3 秒更换条幅广告图像。函数是一组 JavaScript 代码，组合在一起执行特定的功能，如执行计算或像在这个例子中那样更换网页上的图像。

你可能想知道代码如何知道在更换图像前等待多长时间。这由倒数第二行代码实现。看到数字 3000 了吗？它就是条幅广告的等待时间。等待时间的单位是毫秒，3000 毫秒等于 3 秒。要使图像的更换速度更慢或更快，只要修改这个数字即可。

rotateBanner() 函数的另一个重要信息是条幅广告数量，在这段代码中设置为 3（见第二行）。如果要提供更多的条幅广告图像，使页面在所有图像中循环，只要将这个数字改为图像的数量即可。另外，这段代码假设条幅广告图像的文件名为 banner1.gif、banner2.gif 和 banner3.gif 等。如果要使用 JPEG 图像，只要将第四行代码中的文件扩展名.gif 改为.jpg 即可。

在这个条幅广告示例中，另一行重要的代码不可省略，它就是：

```
<body onload="window.setTimeout('rotateBanner( );', 3000);">
```

该行代码使代码开始执行，它告诉 rotateBanner() 函数开始显示和更换图像。注意到这里使用了 onload 属性，前面提到过，这是一个脚本属性，使脚本代码在页面首次加载时运行。你可能还注意到了，这段代码也包含数字 3000。这是因为必须为条幅广告设置初始等待时间，确保第一幅图像显示一会儿后再换为第二幅图像。

## 17.5 显示随机格言

我自己是一个对格言着迷的人。如果你也是，可能觉得在网页中包含不断变化的格言很有趣。如果对格言不感兴趣，可以找其他一些感兴趣的文本放在网页中。关键在于，可以使用 JavaScript 在每次页面加载时显示不同的文本。

要创建包含每次页面加载时改变格言的网页，首先必须收集所有格言及其来源。然后将这些格言保存在 JavaScript 数组中，数组是一种便于保存列表的特殊脚本存储单元。将格言加载到数组中后，随机取出的脚本代码格言就相当简单。

程序清单 17.4 列出了每次加载时显示随机格言的网页的 HTML 和 JavaScript。

#### 程序清单 17.4 显示随机格言的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Quotable Quotes</title>

    <script type="text/javascript">
      <!-- Hide the script from old browsers
      function getQuote() {
        // Create the arrays
        quotes = new Array(4);
        sources = new Array(4);

        // Initialize the arrays with quotes
        quotes[0] = "When I was a boy of 14, my father was so ignorant..." +
          "but when I got to be 21, I was astonished at how much he had learned " +
          "in 7 years.";
        sources[0] = "Mark Twain";
        quotes[1] = "Everybody is ignorant. Only on different subjects.";
        sources[1] = "Will Rogers";
        quotes[2] = "They say such nice things about people at their funerals " +
          "that it makes me sad that I'm going to miss mine by just a few days.";
        sources[2] = "Garrison Keillor";
        quotes[3] = "What's another word for thesaurus?";
        sources[3] = "Steven Wright";

        // Get a random index into the arrays
        i = Math.floor(Math.random() * quotes.length);

        // Write out the quote as HTML
        document.write("<dl style='background-color:#EEEEEE' >\n");
        document.write("<dt>" + "<i>" + quotes[i] + "</i>\n");
        document.write("<dd>" + "-" + sources[i] + "\n");
        document.write("<dl>\n");
      }
      // Stop hiding the script -->
    </script>
  </head>

  <body>
    <h1>Quotable Quotes</h1>
    <p>
```

```

    Following is a random quotable quote. To see a new quote just reload this
    page.
  </p>
  <script type="text/javascript">
    <!-- Hide the script from old browsers
    getQuote();
    // Stop hiding the script -->
  </script>
</body>
</html>

```

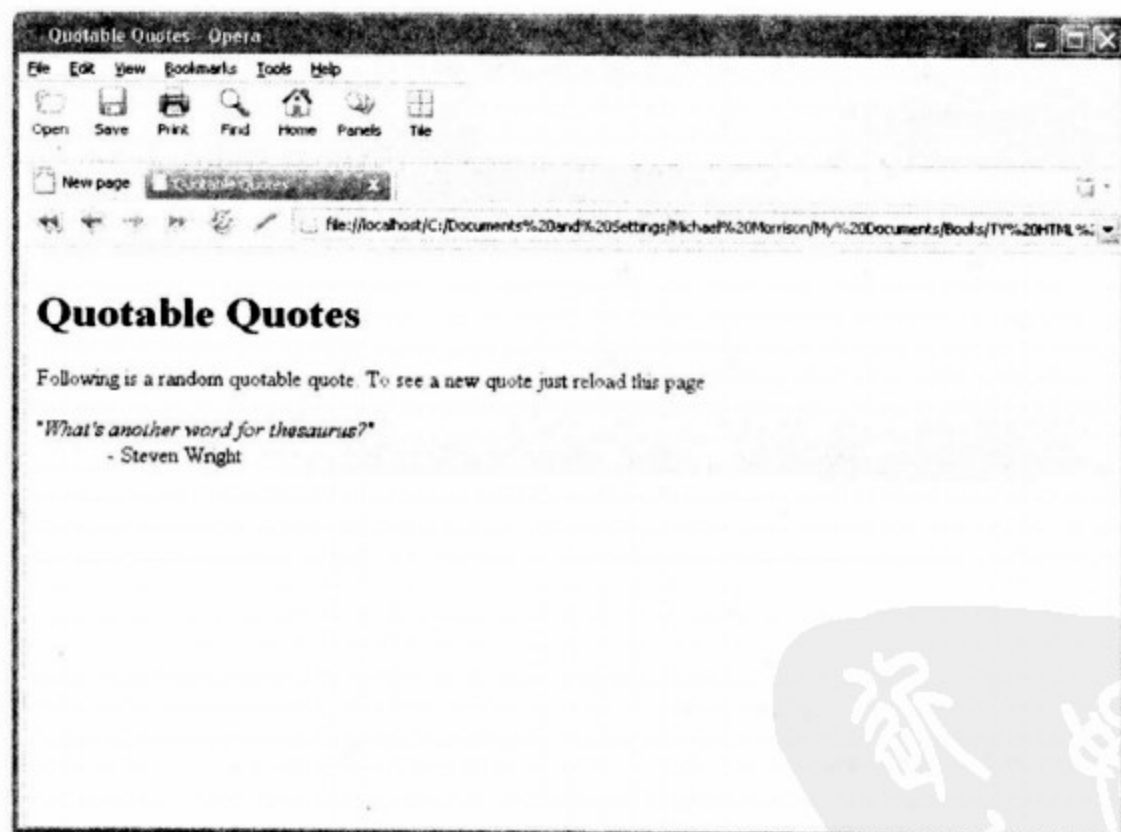
虽然这段代码看起来很长,但如果仔细研究,将发现其中很大的一部分由四条可在网页中显示的格言组成。这个网页的代码比较多,但并不很复杂。

创建包含格言及其来源的数组或列表后,脚本函数 `getQuote()` 获得一个随机数,并根据这个数字来选择要显示的格言。格言由 `getQuote()` 函数生成的 HTML 代码进行格式化。JavaScript 标准函数 `document.write()` 生成格式化格言的 HTML 代码。这个脚本函数的功能很强大,因为使用它可以在网页的任何位置动态地生成 HTML 代码。

图 17.7 显示了 Quotable Quotes 网页在 Opera 浏览器中的显示结果。要显示不同的格言,只要在网页浏览器中单击“刷新”按钮,这将重新加载该网页。

图 17.7

Quotable Quotes  
网页每次加载时随  
机显示一条格言



很容易修改这个网页,以包含自己的格言或其他要随机显示的文本。也可给代码中的 `quotes` 和 `sources` 数组增加元素,从而增加可显示的格言数量。这段脚本代码的有些地方可能比较难以理解。读者只要知道,学习如何使用 JavaScript 的最好方法就是利用一些可执行的代码,对其进行修改,然后试着在网页中使用它。

如果从 Quotable Quotes 网页开始,我保证不需多少功夫就能修改并创建自己感兴趣的东西。如果犯了错误,也没关系。在脚本代码中排除错误的关键是要有耐心,仔细分析输入的

代码。可以删除代码来简化脚本，直到脚本可以执行，然后每次增加一行代码，确保每行代码都可以执行。

## 17.6 创建虚拟鱼塘

为说明脚本的功能有多强大，下面介绍如何动态地在页面上移动图像。为演示移动图像很有用，一种有趣的方式是创建虚拟鱼塘网页，其中鱼看起来就像在鱼塘中游动。使用脚本代码和 CSS 样式属性来修改鱼图像的位置，使其在网页上移动。

可以使用 CSS 来定位图像和文本，使它们显示在网页中准确的位置。这种定位称为绝对定位，在第 14 章介绍过。通过使用绝对定位对元素进行定位，可以使用脚本代码轻松地修改它在网页中的位置。下面是一个使用绝对定位指定图像位置的例子：

```

```

绝对定位让你能够指定 HTML 内容在网页中的精确位置；动态定位使用脚本代码在页面加载完毕后修改网页内容的位置。

鱼图像使用绝对定位，将显示在其父元素下方 70 像素处；父元素是<img>标签所在的标签，如标签<p>或<div>。注意，图像的 id 属性被设置为“fish”。id 属性是必要的，这样脚本代码才能访问图像并修改其位置。

程序清单 17.5 是网页示例，它使用脚本代码使鱼图像变为动画，看起来像在网页中游动。

程序清单 17.5 用于在网页中移动鱼图像的 JavaScript 代码

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Virtual Fish Tank</title>

    <script type="text/javascript">
      <!-- Hide the script from old browsers
      var fishStyle;
      function StartSwimming() {
        fishStyle = document.getElementById("fish").style;
        fishStyle.left = document.body.offsetWidth + "px";
        window.setInterval("Swim()", 75);
      }

      function Swim() {
        var left = parseInt(fishStyle.left) - 5;
        fishStyle.left = left + "px";
        if (left < -118)
          fishStyle.left = document.body.offsetWidth + "px";
      }
    </script>
  </head>
  <body>
    <img alt="A fish swimming in a tank." id="fish" style="position:absolute; top:70px; left:0px; width:100px; height:50px; background-color:blue; border:1px solid black; border-radius:50%; border-style:dashed; border-width:2px;"/>
  </body>
</html>
```



```
    }
    // Stop hiding the script -->
</script>
</head>

<body style="background-image:url(water.jpg)" onload="StartSwimming()">
  <h1 style="text-align:center">Virtual Fish Tank</h1>
  <p style="text-align:center">
    Welcome to the virtual fish tank, where you never have to worry about
    remembering to feed the fish.
    
  </p>
</body>
</html>
```

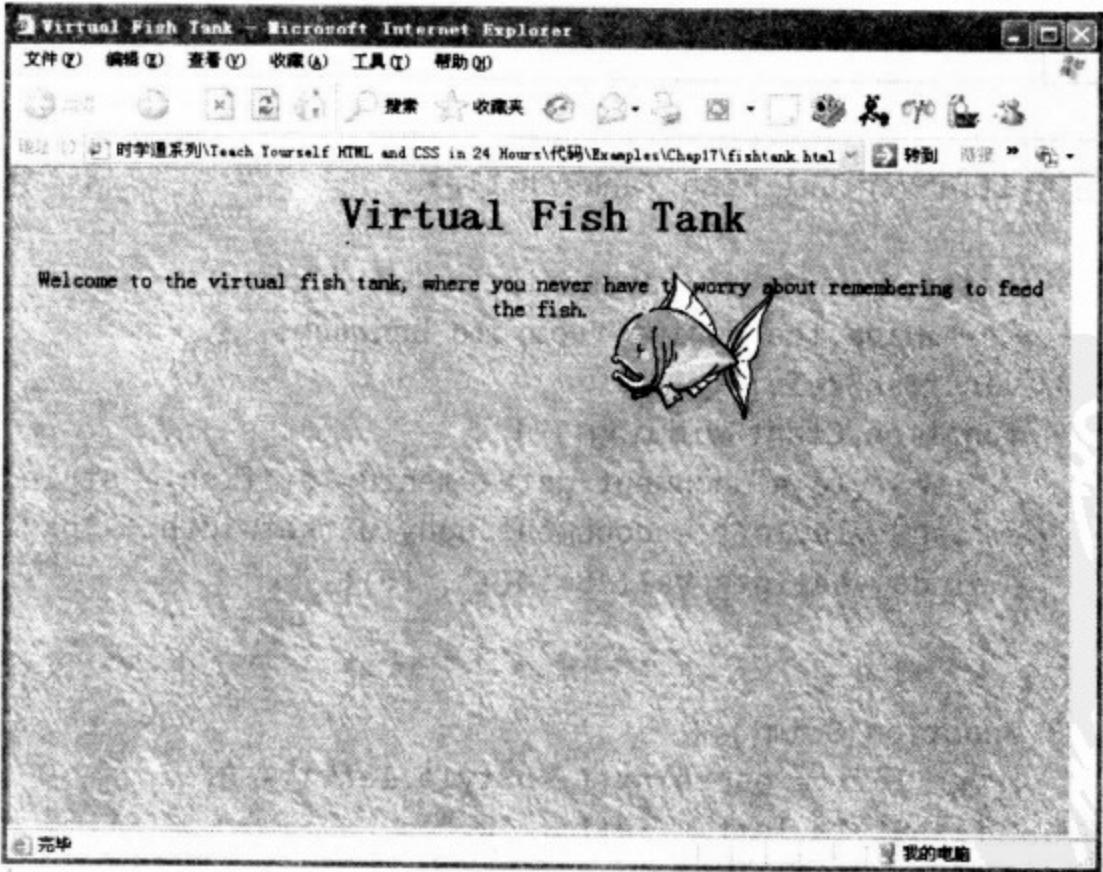
该网页中的脚本代码设置一个定时器，定时器每隔一段时间（这里为 75 毫秒）就执行一个脚本函数。脚本函数 Swim( )减小图像的 left 属性，使它从页面右边向左移动；它还检查 left 属性，看图像是否已移到屏幕左边缘外，如果是，将图像返回到右边缘。

By the Way

注意：根据浏览器和浏览器安全设置，有些包含 JavaScript 的网页可能引起安全警告。这种警告旨在防止恶意脚本对系统造成危害。网页开发人员在这一点上并没有什么解决办法，只能在网页中包含简单的注释，让访问者知道网页是安全的。

图 17.8 是鱼在网页中游动的屏幕截图。

图 17.8  
虚拟鱼塘网页演示了如何使用脚本代码动态地改变图像的位置



我还没有提到鱼图像本身是 GIF 动画，这大大增强了游动效果的真实性的。



我简单地介绍了虚拟鱼塘网页的一些细节，这是另一个不必成为 JavaScript 专家就可以在网页中包含脚本代码的例子。

## 17.7 在网站上使用 Google 广告赚钱

你可能没有认识到可以利用 JavaScript 来赚钱。我说的并非以 JavaScript 程序员为生，而是使用网页来吸引访问者，然后依靠他们对网页中的广告的反应来赚钱。不要担心，我并不是要教你如何创建烦人的弹出广告之类的东西。

流行的搜索引擎 Google 提供了名为 AdSense 的免费服务，让你能够将内容驱动的 Google 广告放在网页中。当访问者单击这些广告时，你就可以收取代理金。是的，就这么简单。利用广告赚钱的难处在于网站要有足够的流量，这样才有人单击广告。

尽管 Google 不能帮助提高网站流量，但 AdSense 根据网页内容智能地放置相关的广告，确实能够提高广告点击率。换句话说，可以让 Google 将广告放在你的网页中他们认为最能获益的地方。根据我的经验，Google 在网页内容的匹配方面做得非常好。

**休息一下：** Google 不仅是一个非常强大的搜索引擎，而且它能够读懂你的思想，将广告放在你的网页中，从而提供了一种利用网页赚钱的方式。如果不信，可以尝试 <http://www.google.com/intl/en/mentalplex/> 的 Google MentalPlex 特性。

Coffee  
Break

要看 Google AdSense，见图 17.9，它显示我个人网页上完全由网页主题驱动的广告（BlackBerry 设备）。

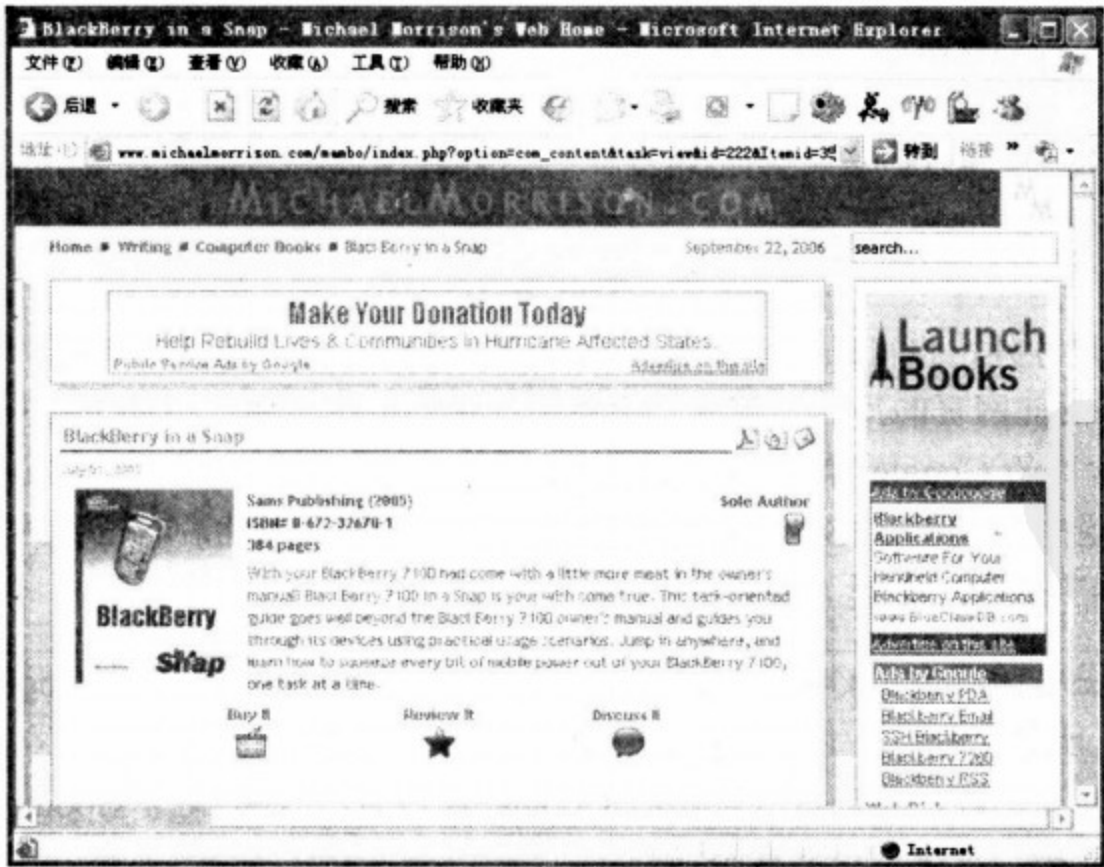


图 17.9  
我的个人网站使用 Google AdSense 来显示内容驱动的广告

该图的网页是关于我写的一本 BlackBerry 书，结果由 Google 显示的广告都与 BlackBerry 设备有关。顺便说一下，美国红十字会广告不是 AdSense 广告，这是我创建的，旨在鼓励人

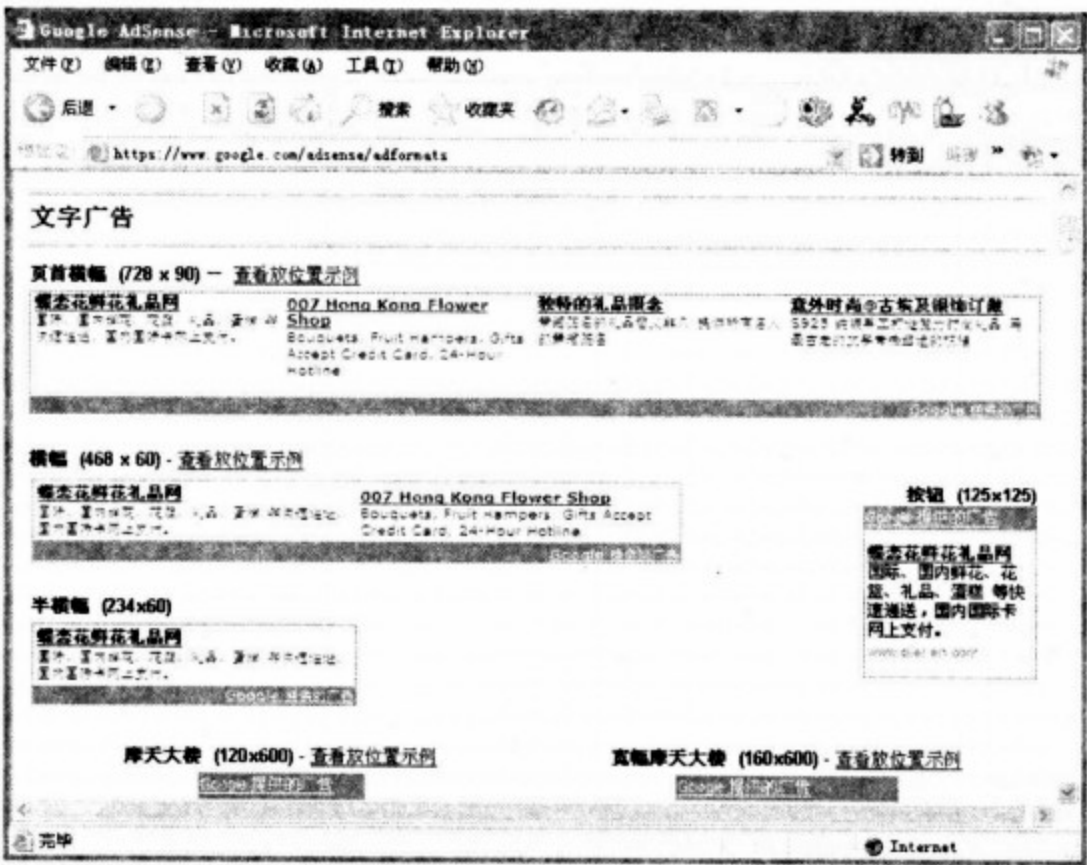
们为红十字会的灾难援助捐款。

要使用 Google AdSense，首先需要创建一个 Google 账号。这是完全免费的，主要要求提供一个电子邮件地址，并同意 Google 的使用条款。要注册 Google 账号并开始使用 AdSense，可访问 <https://www.google.com/adsense/>，然后单击“申请 AdSense”按钮。你将收到一个惟一的客户 ID，该 ID 用于将你的网页链接到 AdSense，这样你就能赚取代理金。

创建账号并登录后，就可以立即开始创建广告。Google 自动生成可直接放到网页中的 JavaScript 代码，所以 AdSense 很容易使用。AdSense 主要的代码生成工具称为 AdSense for Content，可从 <https://www.google.com/adsense/code> 获得。

AdSense 支持多种广告，包括文字广告、图像广告以及可使用文字也可使用图像的广告。还有各种尺寸的广告可供选择。图 17.10 显示了一部分可选择的广告尺寸和布局。在 Google AdSense for Content 网页中单击 View Samples link，就可看到这个网页。

图 17.10  
Google AdSense  
支持许多广告类型  
和尺寸



除选择广告类型和尺寸外，也可以从多个调色板中进行选择。指定所有细节后，将进入 Google AdSense for Content 网页的最后一部分，它将根据你的选择生成广告的 JavaScript 代码。这段代码将自动包含你的 Google 客户 ID，因此你不必自己输入。如果单击包含 JavaScript 代码的编辑框，将自动选中代码，让你能够复制代码（见图 17.11）。为此，右击代码然后选择“复制”，或从浏览器主菜单中选择“编辑/复制”。

**Watch Out!**

**警告：**图中所显示的 Google AdSense 示例依赖于<script>标签中 google\_ad\_client 属性的虚拟客户 id，与 googlead.html 网页中的代码一样。将广告放在自己的网页中时，要记得用自己的 Google 客户 ID。如果使用 Google 生成的脚本代码，应该不会有这个问题。

现在已经有一些 JavaScript 代码，可放到网页中。只要将这些代码粘贴到网页的正文中，如程序清单 17.6 所示。确保复制并粘贴这些代码，而不进行任何修改。

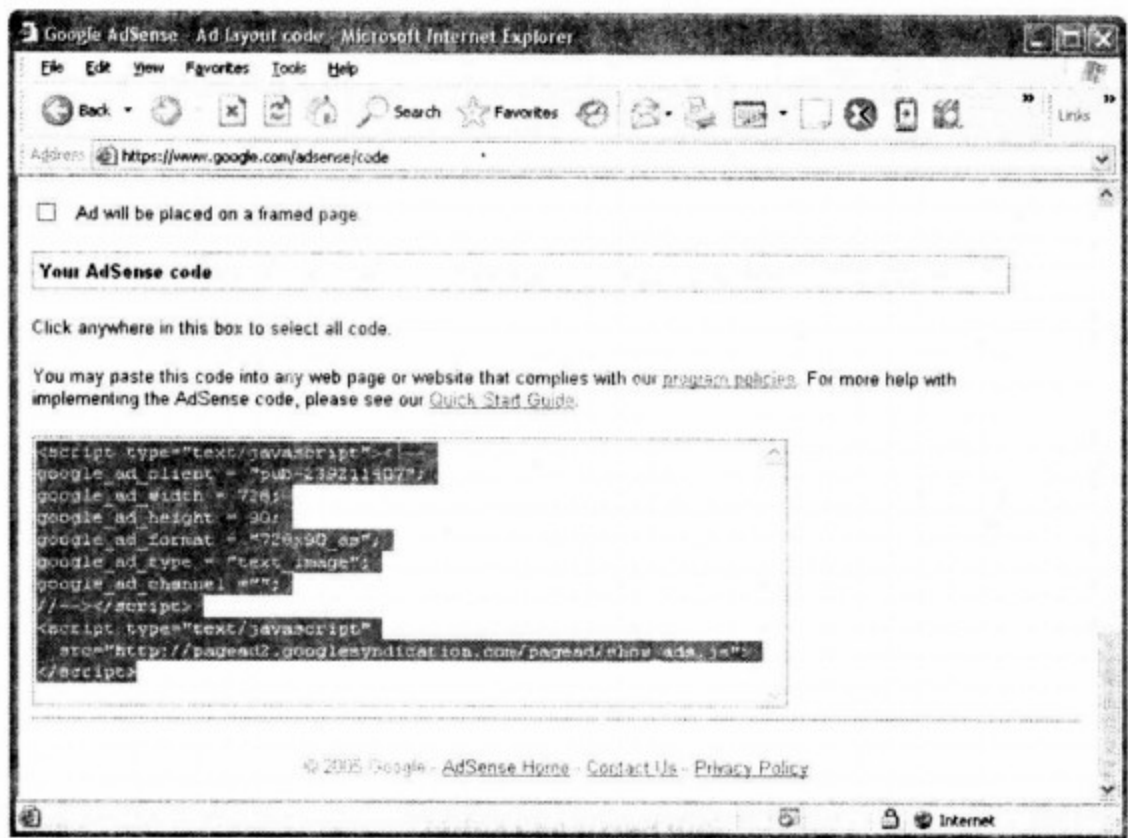


图 17.11  
Google AdSense  
自动为你创建的  
广告生成可用的  
JavaScript 代码

程序清单 17.6 将 Google AdSense 的 JavaScript 代码复制到网页正文中

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Google Ads</title>
  </head>

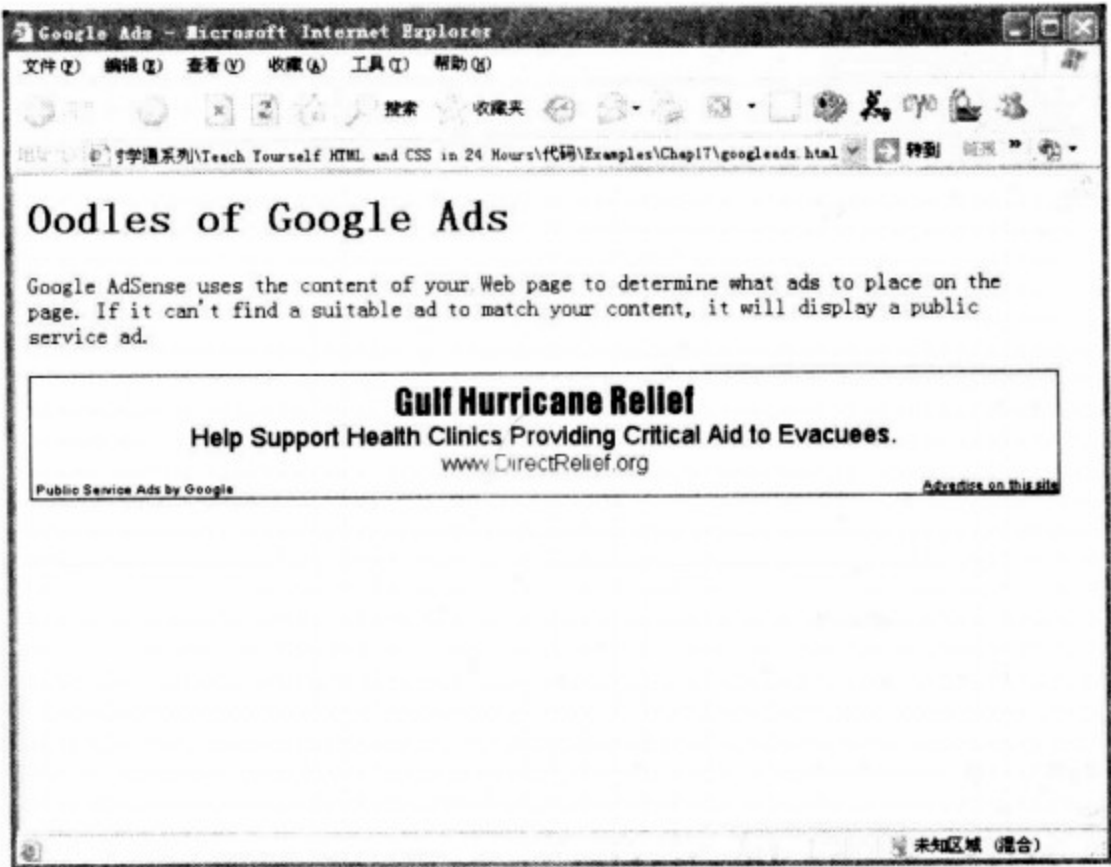
  <body>
    <h1>Oodles of Google Ads</h1>
    <p>
      Google AdSense uses the content of your Web page to determine what ads to
      place on the page. If it can't find a suitable ad to match your content,
      it will display a public service ad.
    </p>
    <script type="text/javascript"><!--
      google_ad_client = "pub-239211407";
      google_ad_width = 728;
      google_ad_height = 90;
      google_ad_format = "728x90_as";
      google_ad_type = "text_image";
      google_ad_channel = "";
      google_color_border = "000000";
      google_color_bg = "F0F0F0";
      google_color_link = "0000FF";
      google_color_url = "008000";
      google_color_text = "000000";
```



```
//--></script>
<script type="text/javascript"
  src="http://pagead2.googlesyndication.com/pagead/show_ads.js">
</script>
</body>
</html>
```

程序清单 17.6 所列的网页如图 17.12 所示，包含 Google AdSense 公共服务广告。

图 17.12  
Google AdSense  
自动网页上显示  
广告



在这个例子中，网页中没有足够的内容让 AdSense 能够匹配广告，因此它选择了一个公共服务广告。这种情况发生在网页的内容很少或很模糊且没有广告适用于这些内容时。

**Coffee Break**

休息一下：很难找到比数着坐享其成得来的钱更好的休息了。访问 <https://www.google.com/adsense/report/overview>，查看 AdSense 广告代理金报告。即使还没有产生许多可转换为现金的点击，至少也可以看看有多少人看了广告。

## 17.8 JavaScript 的广阔世界

你在本章学到了足够的知识，可以开始使用 JavaScript，为网页添加有吸引力的交互。你可能还没有完全认识到 JavaScript 还能实现许多功能，但它并不像你想的那么难。

还可以在网上找到不需要修改或只需少量修改就可以集成到网页中的脚本。在 <http://www.samspublishing.com/> 的文件中可以找到一些与 JavaScript 相关的链接，在这些链接中可以找到脚本。

找到要复用或进行试验的脚本后，注意 JavaScript 代码的放置位置；通常，函数放在 <head>

部分，前面加上<script type="text/javascript">，后面加上</script>。在页面加载时执行操作的脚本放在页面的<body>部分，但仍需要放在<script>标签中。响应具体事件（如鼠标移动）的脚本放在<a>或<input>标签中，并指定特定事件属性，如 onmouseover。

也可以将 JavaScript 代码放在一个独立的文件（通常扩展名为.js）中，然后将文件名放在<script />标签内的 src 属性中，如下所示：

```
<script type="text/javascript" src="bingo.js" />
```

使用其他人编写的脚本且不希望这些代码夹杂在 HTML 中时，将 JavaScript 放在单独的文件中很方便。

## 17.9 总结

本章读者知道了如何使用脚本使网页中的图像对鼠标移动作出响应；还了解了如何使用类似的 JavaScript 命令同时修改多个图像，以及创建实用的网页特性，如动画条幅广告。还学习了如何随机显示格言、动态地移动图像以及放置可赚钱的 Google 广告。这些任务都不需要太多的编程技巧，但它们可能激励你学习更多的 JavaScript 语言，使你的网页具有更复杂的交互特性。

## 17.10 问与答

问：微软是否推广另一种脚本编程语言，在 Internet Explorer 中使用？

答：是的。微软无疑希望你使用其基于 Visual Basic 的脚本编程语言 VBScript，但是 Internet Explorer 也完全支持 JavaScript。对于有些 JavaScript 命令，微软的实现与 Mozilla Firefox 中的实现有微小的差异，然而，因为本章介绍的 JavaScript 在所有主要浏览器中都完全一样，所以可以放心地使用。

问：我想显示一些条幅广告的图像，其中每个都链接到不同的网页。这实现起来有多难？

答：这不是很难，但需要一些本章没有介绍的 JavaScript 编程知识。如果你想利用本章的示例，对其进行修改来做更有趣的事情，强烈建议花些时间学习 JavaScript；否则，应使用这些示例而不做较大的修改。

## 17.11 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 17.11.1 测验

1. 假设有一个按钮图片名为 button.gif，还有一个简单的 GIF 按钮动画 flashing.gif，它在绿色与白色之间闪烁。编写 HTML 和 JavaScript 代码，使按钮在鼠标指向它时闪烁，在鼠



标单击它时链接到网页 gohere.html。

2. 修改问题 1 中编写的代码, 使按钮在鼠标指向它时开始闪烁, 并在鼠标移开后仍然闪烁。

3. 编写 HTML 代码实现一个 JavaScript 函数, 它可以显示 5 个条幅广告图像, 每个广告图像之间相隔 4 秒。广告<img>标签的 ID 是 mybanner, 广告图像的名称分别为 banthis1.jpg、banthis2.jpg 等。

## 17.11.2 答案

1.

```
<a href="gohere.html"
onmouseover="flasher.src='flashing.
onmouseout="flasher.src='button.
</a>
```

2.

```
<a href="gohere.html"
onmouseover="flasher.src='flashing
</a>
```

3.

```
function rotateBanner() {
    if (++bannerNum > 5)
        bannerNum = 1;
    document.getElementById("mybanner").src = "banthis" + bannerNum + ".jpg";
    window.setTimeout('rotateBanner();', 400);
}
```

## 17.12 练习

还等什么? 你现在已经是 HTML 专家, 可以进一步学习 JavaScript 命令, 准备在网页制作方面实现大的突破。

## 第 18 章

# 使用 HTML 表单收集信息

到现在为止，本书都集中于输出信息（第 6 章介绍的电子邮件链接除外）。但 HTML 是一条双行道；也可以使用网页向阅读网页的人收集信息。

Web 表单让你能够接收反馈、订单和来自网页浏览者的其他信息。如果用过 Google、Lycos 或 Yahoo! 等搜索引擎，应熟悉 HTML 表单。产品订单也是表单极其常见的一种用途。本章介绍如何创建表单以及如何处理表单提交的基础知识。

HTML 表单是网页的一部分，包含访问可以输入信息以发送给网页发布者的区域。

### 18.1 HTML 表单的工作原理

学习用于创建表单的 HTML 标签前，应理解访问者在表单中填写的信息如何发送给你。可能还需要让运行 Web 服务器的人对其进行设置以处理表单。

每个表单都必须包含一个按钮，让用户单击以提交表单。用户单击该按钮时，他填写的所有信息都将（以标准格式）发送到表单中指定的 Internet 地址。在该地址必须有特殊的表单处理程序，你才能收到这些信息；你也可以选择通过电子邮件接收这些格式化信息。

几乎所有提供网页托管服务的 ISP 公司都为客户提供预编程的脚本来处理表单。如果你只想使表单信息发送到电子邮件地址，则不需要使用脚本。脚本让你能够更进一步，它可以对表单信息进行处理，并可能将其保存到数据库中或路由给 Web 服务，如安全支付服务。表单处理脚本通常生成某种回复网页，然后将其发送给用户。

可以进行一些设置，以自动解释和处理表单中的大部分信息。例如，有些服务器软件能够自动批准 Internet 信用卡交易，将订单发送到客户的电子邮件地址进行确认，并将订单直接放到公司内部数据库中以便发货。显然，这种设置很复杂，而表单提交后的数据处理也超出了本书的范围。

**提示：**提供网页托管服务的大多数 ISP 都已经设置了“通用”的表单处理脚本，且会告诉你使用该脚本所需的 HTML 代码。如果 ISP 不能做到这一点

**Did you  
know?**

或要求额外收费，则你使用的很可能不是很好的 ISP！在这种情况下，有如下选择。

- 换一个更好的 Web 托管服务提供商。
- 学习高级的服务器编程。
- 使用表单创建服务（如 feedback.com）来创建和处理表单。虽然这种服务是免费的，用起来也不错，但它们会向表单用户显示其他公司的广告。

## 练习

将表单放到网上前，应完成下列任务。

- 询问 ISP 在表单处理脚本方面提供了哪些服务以及表单应将信息发送到哪个地址。本章稍后将介绍如何将该地址放到表单中以及放到哪里。
- 如果运行自己的 Web 服务器，服务器软件可能自带了一些基本的表单处理脚本。参考相关的文档正确地设置这些脚本，并找出每个表单处理脚本在服务器中的地址。
- 如果不确定有什么脚本可用，并希望使用最简单的表单处理方法，可对表单进行配置使其将原始表单数据发送到你的电子邮件地址。本章的示例使用这种方法。以后读者可尝试更有趣的脚本。

## 18.2 创建表单

每个表单都必须以 `<form>` 标签开始，该标签可放在 HTML 文档主体的任何位置。form 标签通常有两个属性，即 `method` 和 `action`：

```
<form method="post" action="mailto:me@mysite.com">
```

当前，`method` 属性几乎都设置为 `post`，这表示将表单输入结果作为文档发送。在有些特殊情况下，可能需要使用 `method="get"`，这将表单结果作为 URL 报头的一部分提交。例如，通过 Web 表单提交查询给搜索引擎时有时使用 `get`。由于读者不是表单专家，因此只要使用 `post` 即可，除非 Web 服务器管理员告诉你使用别的。

`action` 属性指定要将表单数据发送到的地址。有两种选择。

- 输入 Web 服务器中表单处理程序或脚本的地址，这样表单数据将发送给该程序。
- 输入 `mailto:` 和你的电子邮件地址，这样表单数据将直接发送给你。然而，这种方法完全依赖于用户计算机是否正确配置了电子邮件客户端程序。从公共计算机访问网站而没有电子邮件客户端程序的人将无法提交表单数据给你。

### By the Way

**注意：**许多 ISP 提供免费脚本用于处理表单。向 ISP 或 Web 服务器管理员核实是否有这样的脚本。如果有，只要在 `<form>` 标签中的 `action` 属性引用适当的脚本即可；否则，需要通过电子邮件来接收表单数据。

程序清单 18.1 和图 18.1 所示的表单包含当前可在 HTML 表单中使用的各种用户输入。图 18.2 是图 18.1 所示表单在用户填写数据后的样子。阅读下述对每种输入元素的解释时，可参阅这两个图。

程序清单 18.1 表单的所有部分都必须在标签<form>和</form>之间

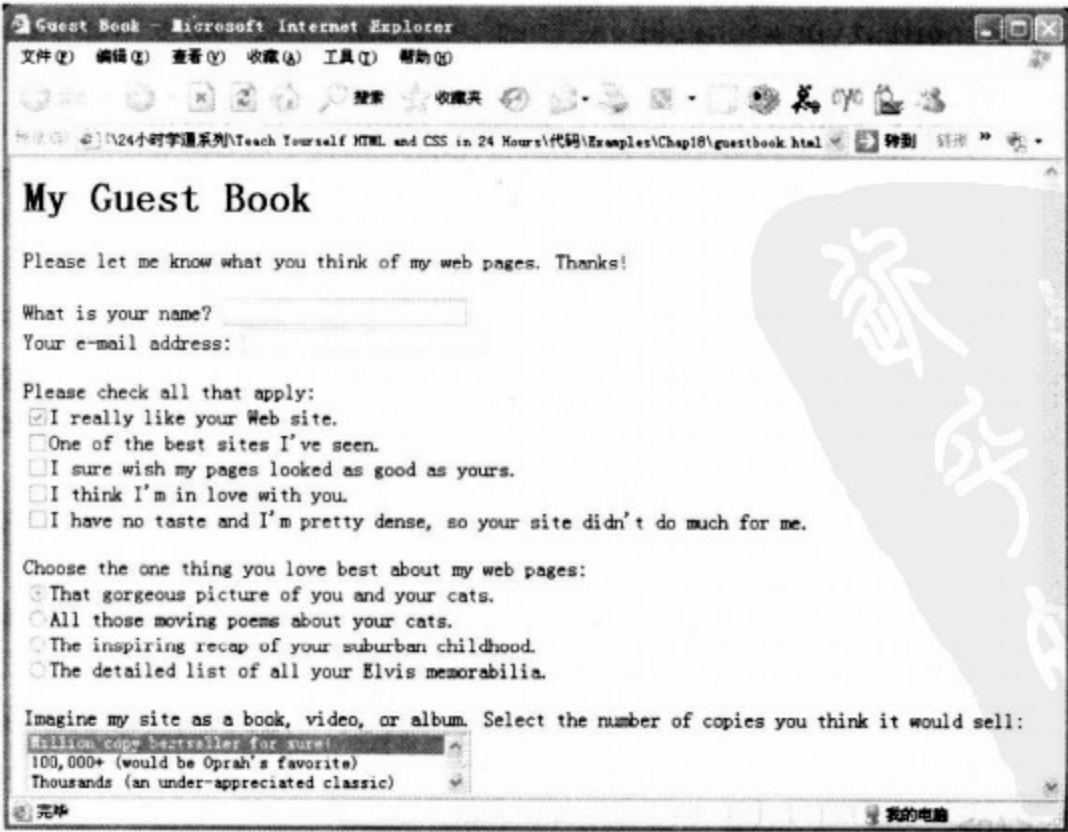
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Guest Book</title>
  </head>

  <body>
    <h1>My Guest Book</h1>
    <p>
      Please let me know what you think of my web pages. Thanks!
    </p>
    <form method="post" action="mailto:you@youremail.com"
    enctype="text/plain">
      <p>
        What is your name? <input type="text" name="fullname" size="25" /><br />
        Your e-mail address: <input type="text" name="e-address" size="25" />
      </p>
      <p>
        Please check all that apply:<br />
        <input type="checkbox" name="likeit" checked="checked" />I really like
        your Web site.<br />
        <input type="checkbox" name="best" />One of the best sites I' ve
        seen.<br />
        <input type="checkbox" name="envy" />I sure wish my pages looked as good
        as yours.<br />
        <input type="checkbox" name="love" />I think I' m in love with you.<br />
        <input type="checkbox" name="idiot" />I have no taste and I' m pretty
        dense, so your site didn' t do much for me.
      </p>
      <p>
        Choose the one thing you love best about my web pages:<br />
        <input type="radio" name="lovebest" value="me" checked="checked" />That
        gorgeous picture of you and your cats.<br />
        <input type="radio" name="lovebest" value="cats" />All those moving poems
        about your cats.<br />
        <input type="radio" name="lovebest" value="burbs" />The inspiring recap of
        your suburban childhood.<br />
        <input type="radio" name="lovebest" value="treasures" />The detailed list
        of all your Elvis memorabilia.
      </p>
    </form>
  </body>
</html>
```

```
<p>
  Imagine my site as a book, video, or album. Select the number of copies
  you think it would sell:<br />
  <select size="3" name="potential">
    <option selected="selected">Million copy bestseller for sure!</option>
    <option>100,000+ (would be Oprah' s favorite)</option>
    <option>Thousands (an under-appreciated classic)</option>
    <option>Very few: not banal enough for today' s public</option>
  </select>
</p>
<p>
  How do you think I could improve my site?
  <select name="suggestion">
    <option selected="selected">Couldn' t be better</option>
    <option>More about the cats</option>
    <option>More Elvis stuff</option>
    <option>More family pictures</option>
  </select>
</p>
<p>
  Feel free to type more praise, gift offers, etc. below:<br />
  <textarea name="comments" rows="4" cols="55">I just want to thank you so
  much for touching my life.</textarea><br />
  <input type="submit" value="Click Here to Submit" />
  <input type="reset" value="Erase and Start Over" />
</p>
</form>
</body>
</html>
```

图 18.1  
程序清单 18.1 所  
示的表单代码使用  
了各种 HTML 表单  
输入元素





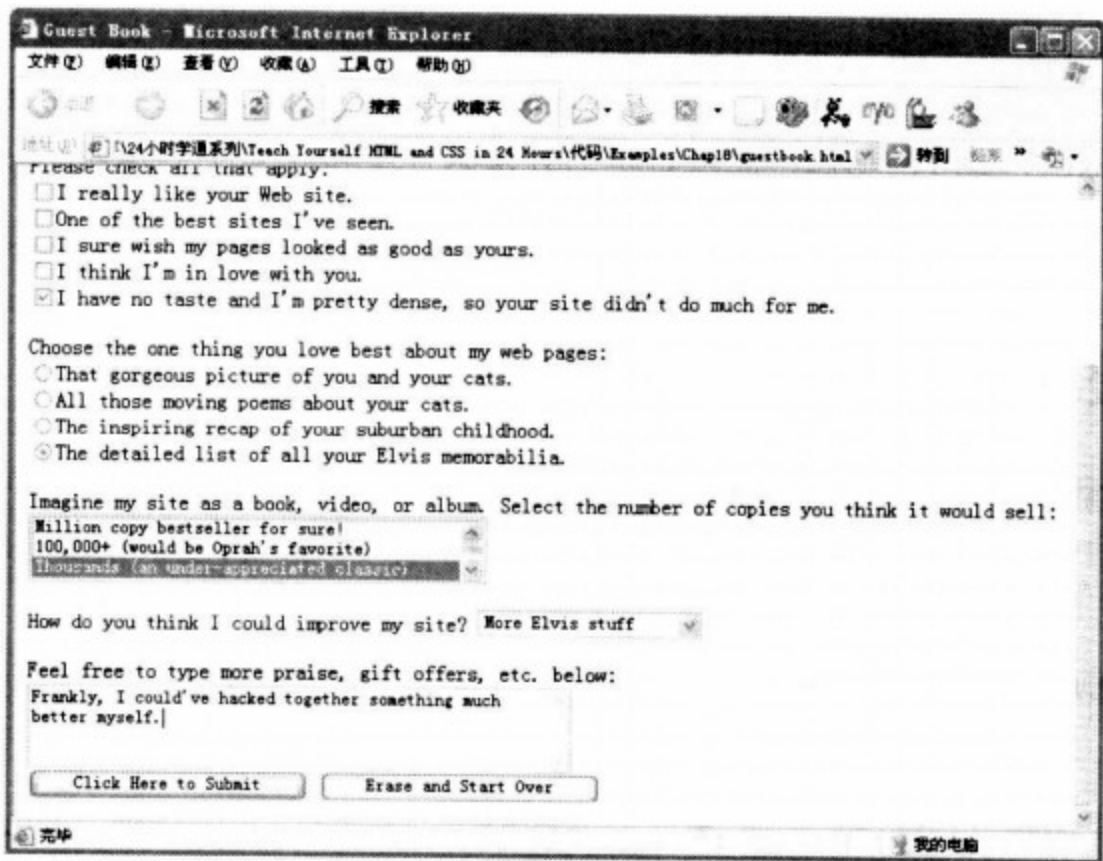


图 18.2  
网站访问者填写  
表单，然后单击按  
钮 Click here to  
Submit

该网页的代码（程序清单 18.1）使用了一个包含许多<input />标签的<form>标签。每个<input />标签对应于一个用户输入元素，如复选框或单选按钮。接下来的几小节将详细介绍<input />标签。

### 18.3 接受文本输入

要请用户在表单中输入信息，可使用<input />标签。该标签必须位于<form>和</form>标签之间，但与网页中任何文本、图像和其他 HTML 标签之间的相对关系没什么限制。例如，要询问用户的姓名，可以输入下列代码：

```
What's your first name? <input type="text" size="20" maxlength="30"
name="firstname" />
What's your last name? <input type="text" size="20" maxlength="30"
name="lastname" />
```

type 属性指明要显示的是哪种表单元素，这里是单行文本框（接下来的每小节将单独介绍一种元素）。

size 属性指明文本框的宽度大约为多少个字符。如果使用的不是等宽字体，则输入框的宽度将随用户输入的字符而异。如果输入太长，不能全部显示在文本框中，大多数网页浏览器将自动向左滚动文本。

maxlength 决定文本框允许用户输入的最大字符数。如果用户的输入超过指定的长度，多余的字符将不会显示。指定的长度可以比文本框大，也可以比它小，还可以相等。size 和 maxlength 只能用于 type="text"，因为其他输入类型（复选框、单选按钮等）的大小是固定的。

**Did you know?**

**提示：**如果希望用户输入文本而不显示在屏幕上，可用 `<input type="password" />` 来代替 `<input type="text" />`。这样用户输入的文本将显示为星号 (\*\*\*)。属性 `size`、`maxlength` 和 `name` 在 `type="password"` 中的用法与在 `type="text"` 中相同。记住，隐藏密码的方法只能提供视觉上的保护——密码传输时没有加密或其他保护措施。

## 18.4 标识每项表单数据

不管是哪种输入元素，都必须为其收集的数据赋予名称。可以为每个输入项使用任何名称，只要在表单中是惟一的。表单被发送给你（或表单处理脚本）时，每项数据都是根据名称来标识的。

例如，如果有人在前面定义的文本框中输入 Jane 和 Doe，表单被提交后，你将在电子邮件中看到下面这两行：

```
firstname=Jane
lastname=Doe
```

下列是程序清单 18.1 中的表单指定的表单处理脚本生成的电子邮件。注意到每个数据元素都由程序清单 18.1 指定的名称来标识：

```
fullname=Frank Rizzo
e-address=frank@rizzoratchets.com
idiot=on
lovebest=treasures
potential=Thousands (an under-appreciated classic)
suggestion=More Elvis stuff
comments=Frankly, I could've hacked together something much better myself.
```

**By the Way**

**注意：**Web 服务器处理电子邮件表单的方式不同，因此你可能看到与这里所示的格式不同的数据。

## 18.5 在表单中包含隐藏数据

要将数据项发送给处理表单的服务器脚本而不希望用户看到这些数据，可使用 `input type="hidden"` 属性。这个属性对显示没有任何影响，它只是在表单结果提交时添加指定的名称和值。

可用这个属性告诉脚本将表单结果发送到哪个电子邮件地址。例如，下面的代码指示将结果发送到 `me@mysite.com`：

```
<input type="hidden" name="mail_to" value="me@mysite.com" />
```

要使这个属性生效，必须有脚本或程序读取该行，并对其进行处理。我的 ISP 表单处理

程序使用这个隐藏值判断要将表单数据发送到哪个电子邮件地址。

大多数脚本都至少要求一两个隐藏的输入元素。具体细节请询问编写或提供脚本的人。

## 18.6 表单输入控件

有多种输入控制可用于从用户那里收集信息。前面介绍了用于表单的几种文本输入方法。接下来的几个小节将介绍设计表单时可使用的其他表单输入控件。

### 18.6.1 复选框

最简单的输入类型是复选框，它显示为一个小方框，用户可以单击选中它或取消选中。复选框就像开关。每个复选框必须在 `name` 属性中指定名称。要使表单出现时复选框默认被选中，可在标签中包含 `checked` 属性。例如，下列代码创建两个复选框：

```
<input type="checkbox" name="baby" checked="checked" />Baby Grand Piano<br />
<input type="checkbox" name="mini" />Mini Piano Stool
```

**提示：**如果发现输入元素的标签与元素靠得太近，只要在<input />标签和标签文本之间添加一个空格，如下所示：

```
<input type="checkbox" name="mini" /> Mini Piano Stool
```

**Did you Know?**

在这个例子中，复选框 `Grand Piano` 默认被选中。如果用户不想要钢琴，必须单击该复选框以取消选中它。复选框 `Mini Piano Stool` 默认没有被选中，因此用户必须单击才能选中它。表单提交时，选中的复选框将出现在表单结果中：

```
baby=on
```

没有选中的复选框不会包含在表单输出中。

**警告：**XHTML 要求所有属性都要有等号和值。这就是我为什么使用 `checked="checked"` 表示复选框被选中，而不是只使用 `checked` 的原因。该规则适用于所有的 HTML 布尔（`true/false`，`on/off` 和 `yes/no` 等）属性。

**Watch Out!**

**提示：**多个复选框可以有相同的名称和不同的值，如下列代码所示：

```
<input type="checkbox" name="pet" value="dog"> dog<br />
<input type="checkbox" name="pet" value="cat"> cat<br />
<input type="checkbox" name="pet" value="iguana"> iguana
```

如果用户选择复选框 `Cat` 和 `Iguana`，提交的结果将包括下列两项：

```
pet=cat
pet=iguana
```

**Did you Know?**

## 18.6.2 单选按钮

单选按钮每次只能选择一个，实现起来和复选框一样容易。只要使用 `type="radio"`，每个单选按钮放在一个 `<input />` 标签中，但同一组单选按钮的名称都相同：

```
<input type="radio" name="card" value="v" checked="checked" /> Visa<br />
<input type="radio" name="card" value="m" /> MasterCard
```

`value` 可以是任何名称或编码。如果包含了 `checked` 属性，单选按钮将默认被选中（同一组单选按钮最多只能有一个是 `checked` 的）。

如果用户选择了上述单选按钮组中的 MasterCard，表单提交给服务器脚本的结果中将包含下面的数据：

```
card=m
```

如果用户没有修改默认选择，表单结果包含的将是 `card=v`。

## 18.6.3 选择列表

滚动列表和下拉选择列表都由 `<select>` 标签创建。可将该标签与 `<option>` 标签一起使用，如下面的例子所示：

```
<select name="extras" size="3" multiple="multiple">
  <option selected="selected">Electric windows</option>
  <option>AM/FM Radio</option>
  <option>Turbocharger</option>
</select>
```

标签 `<select>` 和 `</select>` 之间除 `<option>` 标签和 `</option>` 标签外，不应包含其他标签。

与文本输入元素不同的是，这里的 `size` 属性决定选择列表可同时显示多少项。如果在上述代码中使用 `size="2"`，则只能同时看到两项，列表旁边将显示滚动条，用户可以向下滚动以查看第三项。

`multiple` 属性让用户能够同时选择多项，`selected` 属性指定默认被选中的项。表单提交时，返回的是被选中的选项的实际文本。例如，如果用户选择了 Electric windows 和 Turbocharger，表单结果将包含下列两行：

```
extras=Electric windows
extras=Turbocharger
```

和 `checked` 属性一样，XHTML 要求使用 `multiple="multiple"` 和 `checked="checked"`。

**Did you Know?**

**提示：**如果没有指定 `size` 属性或指定 `size="1"`，列表将为下拉选择列表。下拉选择列表不允许同时选中多项；从逻辑上说，它相当于一组单选按钮。例

如，选择信用卡类型可采用另一种方法：

```
<select name="card">
  <option>Visa</option>
  <option>MasterCard</option>
</select>
```

### 18.6.4 文本区域

前面介绍的<input type="text">只允许用户输入一行文本。如果希望用户能够输入多行文本，可使用标签<textarea>和</textarea>。包含在这两个标签之间的文本默认显示在文本区域中。下面是一个例子：

```
<textarea name="comments" rows="4" cols="20">Please send more information.
</textarea>
```

你可能猜到了，rows 和 cols 属性控制输入框文本的行数和列数。cols 属性不像 rows 属性那样精确，它是一行文本包含的大概字符数。然而，文本区域也有滚动条，因此用户可以输入比显示区域更多的文本。

## 18.7 提交表单数据

每个表单都必须包含一个用于将表单数据提交给服务器的按钮。可用 value 属性在这个按钮上添加任何标签：

```
<input type="submit" value="Place My Order Now!" />
```

按钮为灰色，大小由 value 属性包含的文本长度决定。用户单击该按钮时，表单中的所有数据都将发送到表单的 action 属性指定的电子邮件地址或程序脚本。

也可在表单中包含另一个按钮，让用户能够在改变主意或犯错时清除表单中的输入。为此，可使用下列代码：

```
<input type="reset" value="Clear This Form and Start Over" />
```

如果觉得标准的 Submit 和 Reset 按钮太乏味，有一种简单的方法可在这两个按钮中使用自己的图像：

```
<input type="image" src="button.gif" alt="Order Now!" />
```

图像 button.gif 将显示在网页中，用户单击该图像时，表单将提交给服务器。也可以包含可用于<img />标签的属性，如 alt 和 style（第 8 章介绍了<img />标签）。

**注意：**鼠标单击图像按钮时的像素坐标将与表单数据一起发送。例如，如果用户在图 18.3 所示的表单中输入 sol@shoesandglasses.net，表单数据将如下所示：

**By the Way**



```
anotherone=sol@shoesandglasses.net_&x=75_&y=36
```

通常应忽略坐标 x 和 y，但有些服务器脚本使用这两个坐标将按钮当作图像映射使用。

程序清单 18.2 和图 18.3 是一个非常简单的表单，使用一个自定义的 Submit 按钮（第 7 章介绍了如何创建与 signup.gif 类似的图像）。

#### 程序清单 18.2 这个网页使用标签创建一个自定义图形的 Submit 按钮

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>FREE!</title>
  </head>

  <body>
    <h1>Free Electronic Junk Mail!</h1>
    <form method="post" action="mailto:you@youremail.com" enctype="text/plain">
      <p>
        To start receiving junk e-mail from us daily*, enter your e-mail address
        below and click on the <em>SignUP!</em> button.
      </p>
      <p>
        <input type="text" name="anotherone" size="25" />
        <input type="image" src="signup.gif" alt="Sign Up Now!"
        style="vertical-align:middle" />
      </p>
      <p>
        By clicking the above button, you also agree to the terms of our Marketing
        Agreement, which is available upon request at our offices in Bangkok,
        Thailand. A fee may be charged for removal from our list if you elect at a
        later date not to receive additional sales literature.
      </p>
    </form>
  </body>
</html>
```

#### Did you Know?

**提示：**可以创建一个链接到另一个网页的按钮，单击该按钮将取消表单直接进入该网页（忽略用户输入的所有信息）。下面是一个例子：

```
<a href="nodice.html">Click here to cancel.</a>
```

没有专门用于创建图形重置按钮的标签，但可以使用指向当前页面的图形链接来实现这种效果，如下所示：

```
<a href="thispage.html"></a>
```

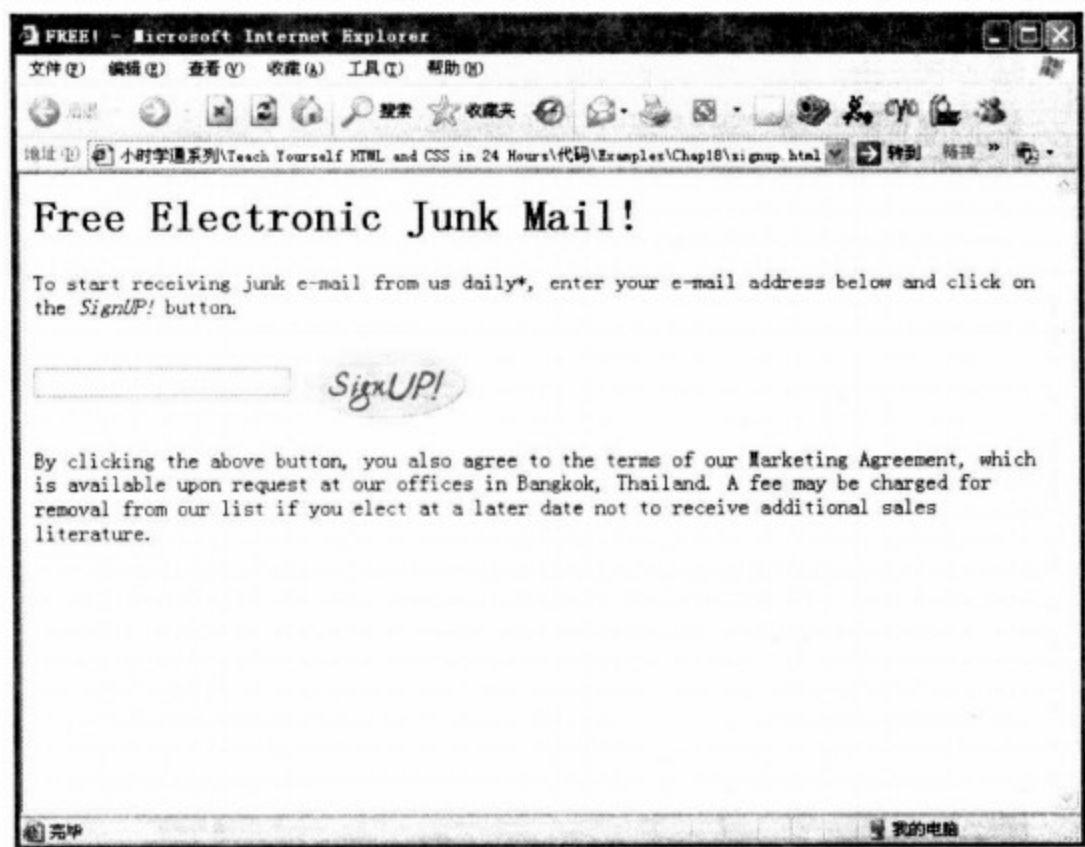


图 18.3  
有效的表单不一定很复杂（虽然它们可能稍为复杂些）

## 18.8 使用 PayPal Buy Now 按钮

表单最常见的用途之一是开展电子商务，电子商务通常都使用表单来实现基于 Web 的购物车。功能完善的购物车表单很复杂，但有一种简单的方法，可以避免自己开发购物车。

我指的是 PayPal Buy Now 按钮，它利用流行的 PayPal 电子支付服务提供可用于你自己的产品和服务的 buy now 链接。如果没有听说过 PayPal，请参阅 PayPal 的网站 (<http://www.paypal.com/>)。

**注意：**PayPal 在几年前被 eBay 收购了，现已成为 eBay 拍卖中最受青睐的支付方式。当然，PayPal 是一种在线支付方式。

**By the Way**

PayPal 以非常规方式使用表单——可以使用表单来创建作为图像显示在网页中的自定义按钮。用户单击该图像时，表单数据将提交给 PayPal，将要购买的商品细节提供给 PayPal。PayPal Buy Now 按钮通过表单来提交购买数据，不需要太多技术就能够提供了个性化购物体验。更重要的是，PayPal Buy Now 按钮让网页开发者很容易销售产品和服务，而不需要花大量的钱在昂贵的购物车应用程序和信用卡处理服务上。

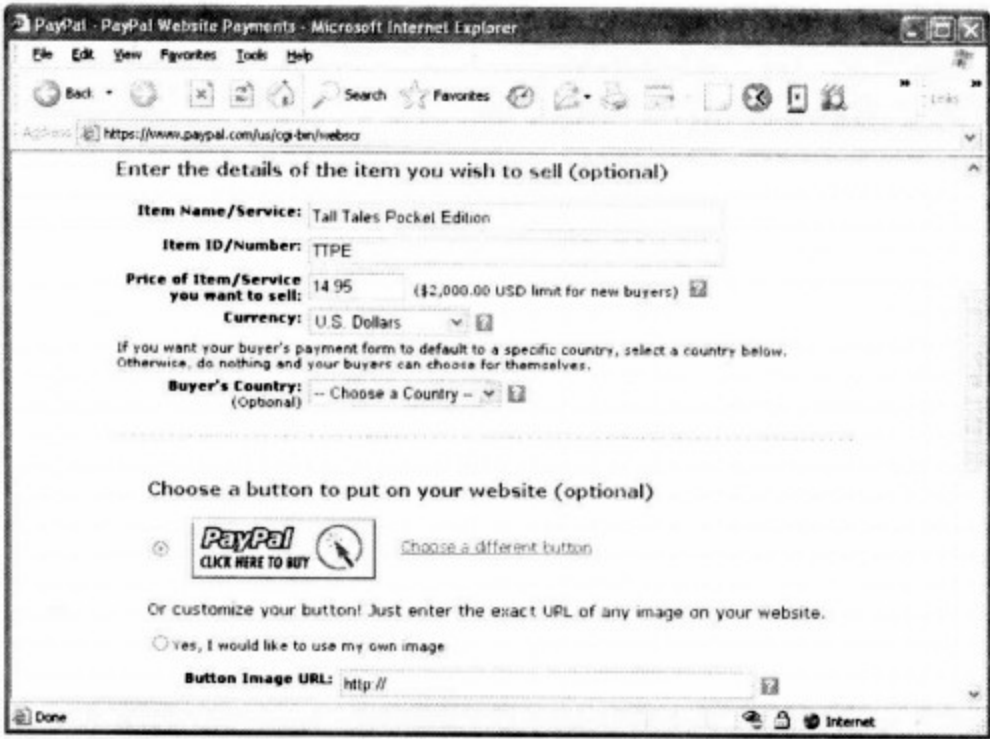
要使用 PayPal Buy Now 按钮，首先需要有一个 PayPal 账户。如果读者还没有 PayPal 账户，可访问 PayPal 的主页 (<http://www.paypal.com/>)，单击左上角的 Sign Up，就可以开始创建账户。

创建账户后，登录并访问 Buy Now Buttons 网页，其网址为 <https://www.paypal.com/us/cgi-bin/Webscr?cmd=p/xcl/rec/singleitem-intro>。单击 Get Started! 链接开始创建 PayPal Buy Now 按钮。图 18.4 是接下来将显示的网页，它提示输入有关要销售的产品或服务的信息。

在本例中，输入与我为公司 Stalefish Labs 创建的一个游戏相关的信息。在该图中可以看到，必须输入产品的名称、编号和价格；其中编号可以是任何内容。还必须指定交易货币，如美元、加拿大元等。也可以设置目标购买者的国家（地区），但我不建议这样做，除非对购买者所

在的地区有要求。

图 18.4  
PayPal Buy now Buttons 工具提示输入要销售的商品的细节



单击 Choose a Different Button，对 PayPal 按钮在网页中的外观进行修改。在本例中，我选择使用一个较大的按钮，使其在网页中更突出。也可以使用自己的按钮图像。

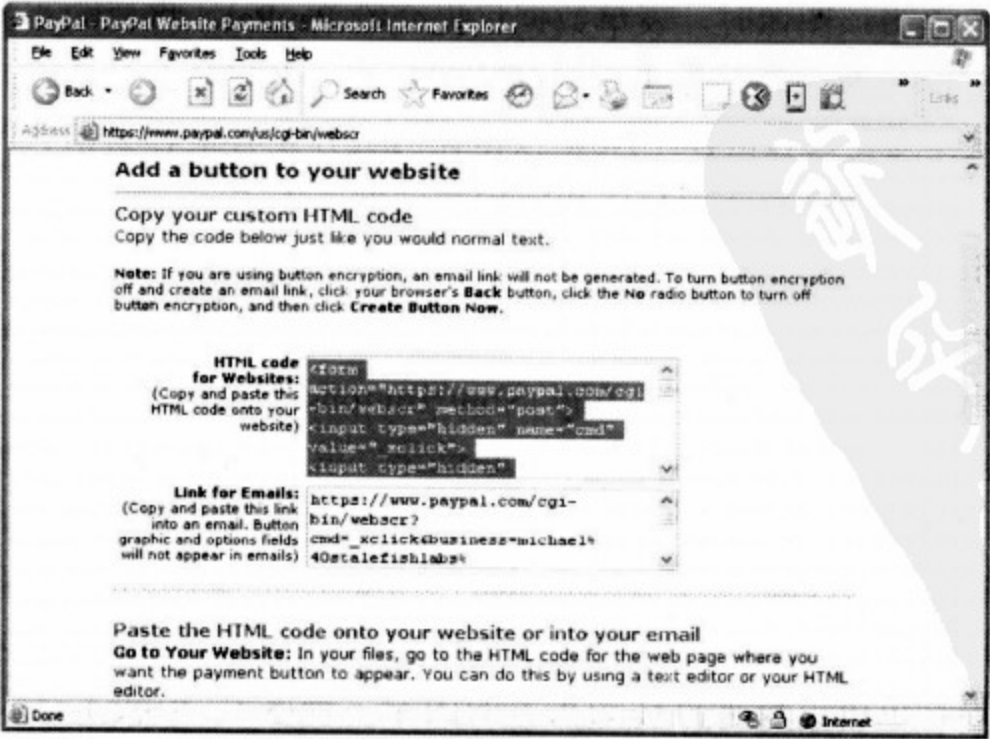
最后一个必须输入的选项在图 18.4 中没有显示出来，这个选项是，是否对 PayPal 按钮的 HTML 代码进行加密以保证安全。网站受到黑客攻击时，这可以使黑客更难以修改商品的价格或其他细节。是否使用该设置完全取决于你——在这里例子中，我选择不加密，使代码更简单。

**Did you know?**

**提示：**单击 Add More Options 按钮可设置 Buy Now 按钮的更多选项，如，自定义选项字段、购买多个商品以及可选的购买者说明等。如果使用加密的 Buy Now 按钮，将不能对该按钮使用某些高级选项。

图 18.5 显示了单击 Create Button Now 后显示的网页。

图 18.5  
完成对商品的描述后，PayPal 自动生成 HTML 代码，你可将代码复制并粘贴到网页中



该图显示了生成完整的 HTML 表单代码的 PayPal Buy Now Button 网页，你可以将这些代码直接复制并粘贴到网页中。为此，只要选中第一个编辑框中的代码，然后将其复制并粘贴到网页中。程序清单 18.3 所示的网页包含自动生成的 PayPal Buy Now 按钮表单代码。

**程序清单 18.3 Tall Tales 网页现在包含一个 PayPal Buy Now 按钮，单击它将通过 PayPal 购买游戏**

---

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Tall Tales - The Game of Legends and Creative One-Upmanship</title>
    <style type="text/css">
      ...
    </style>
  </head>

  <body>
    ...
    <p>
      To learn more about Tall Tales, visit the Tall Tales Web site at <a
      href="http://www.talltalesgame.com/">www.talltalesgame.com</a>. Or to buy
      the game just click the button below:
    </p>
    <form action="https://www.paypal.com/cgi-bin/webscr" method="post">
      <p style="text-align:center">
        <input type="hidden" name="cmd" value="_xclick" />
        <input type="hidden" name="business" value="michael@stalefishlabs.com"/>
        <input type="hidden" name="item_name" value="Tall Tales Pocket Edition"/>
        <input type="hidden" name="item_number" value="TTPE" />
        <input type="hidden" name="amount" value="14.95" />
        <input type="hidden" name="no_note" value="1" />
        <input type="hidden" name="currency_code" value="USD" />
        <input type="hidden" name="bn" value="PP-BuyNowBF" />
        <input type="image"
          src="https://www.paypal.com/en_US/i/btn/x-click-but5.gif"
          style="border-style:none" name="submit" alt="Make payments with
          PayPal - it's fast, free and secure!" />
      </p>
    </form>
  </body>
</html>
```

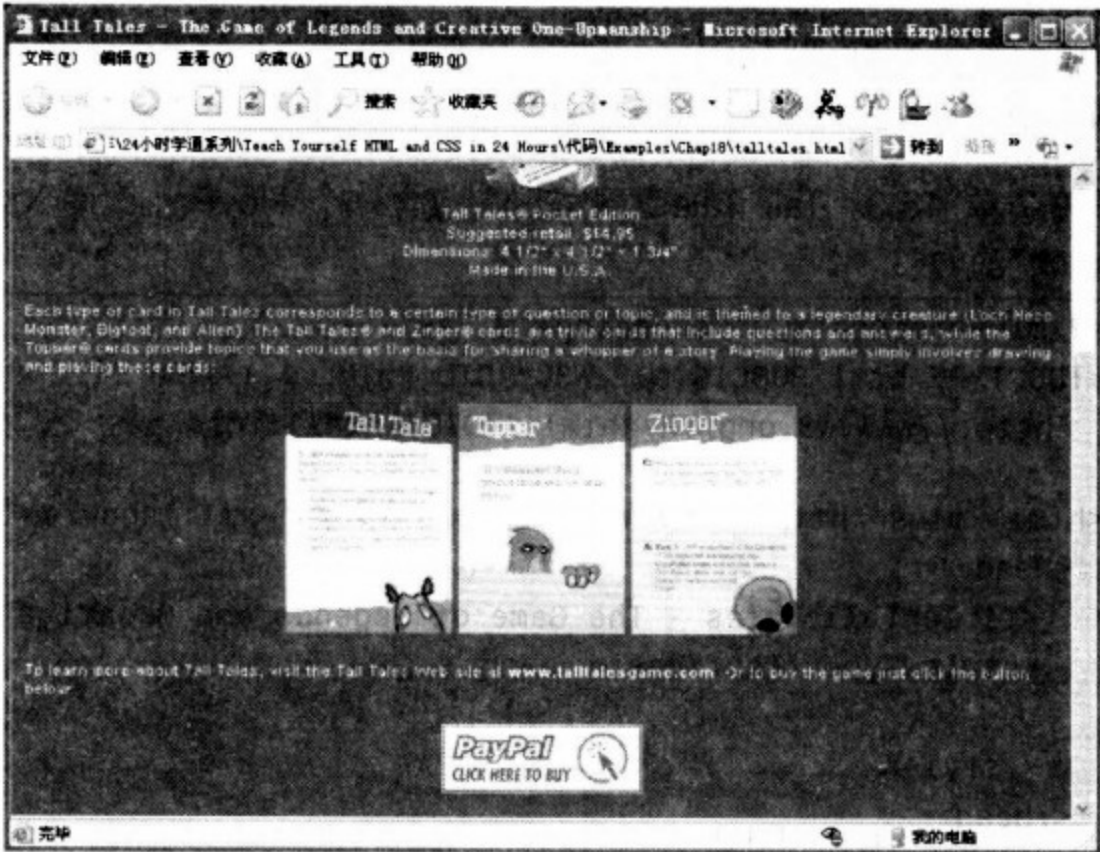
---

这个网页在本书前面也出现过。在这个例子中，一个有趣的促销网页通过简化的购物车提高了价值。换句话说，现在 Tall Tales 网页的访问者不仅可以了解这个游戏，还可以单击 Buy Now 按钮，通过 PayPal 付款以购买游戏（见图 18.6）。



图 18.6

程序清单 18.3 所示的 PayPal 表单代码生成的表单，它只是一个链接到 PayPal 购买网页的按钮



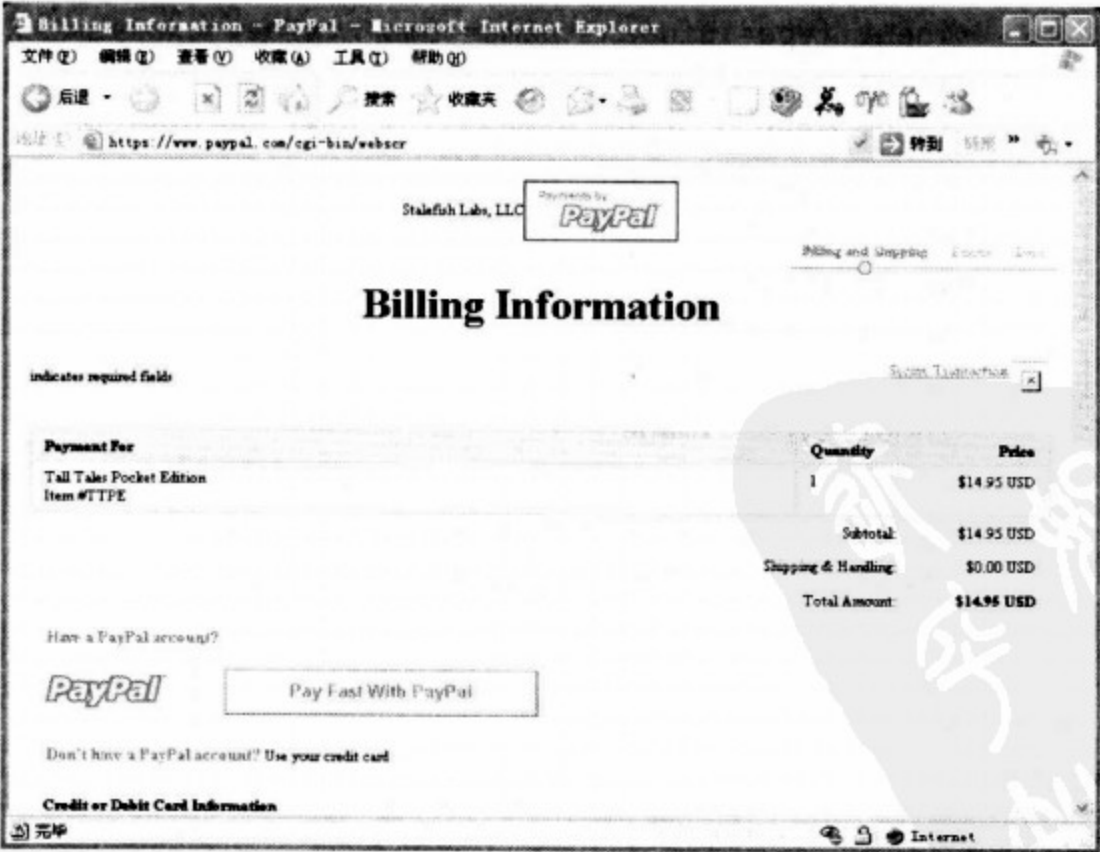
By the Way

**注意：**这里要指出的是，我对 PayPal 生成的代码做了一些修改。PayPal 生成的代码不是 XHTML 兼容的，我不得不将

单击图 18.6 所示屏幕底部的 Buy Now 按钮，将链接到 PayPal 网站的 Checkout 网页，该网页显示了商品细节，让用户确认是否要购买（见图 18.7）。

图 18.7

用户单击 Buy Now 按钮时，表单将链接到 PayPal 网站的购买网页，在这个网页进行购买



从这里开始，交易就交给了 PayPal。通过 PayPal Buy Now Button 购买的金额将自动存到你的 PayPal 账户中，与 PayPal 账户相关联的电子邮件地址将收到购买通知。别忘了将商品



或服务发送给客户还是要由你来完成——PayPal 只是处理支付。

## 18.9 总结

本章演示了如何创建 HTML 表单，表单让网页访问者能够输入特定的信息，并通过电子邮件将其发送给你。读者还了解到，可以在 Web 服务器上建立脚本或程序来处理表单数据，ISP、Web 服务器管理员或服务器软件厂商可以帮助你完成这项工作。读者还学习了如何使用 PayPal Buy Now 按钮，它让人们能够直接通过网页购买商品并支付货款，这对于需要接受在线产品订购的网页非常有用。只要使用 PayPal 生成的 HTML 表单代码，就可以在几分钟内开展在线商业活动并赚钱。

表 18.1 总结了本章介绍的 HTML 标签和属性。

表 18.1 第 18 章介绍的 HTML 标签与属性	
标签/属性	功能
<form>...</form>	指定输入表单。
属性	
action="脚本 URL"	处理表单输入的脚本的地址。
method="post/get"	表单输入如何传送到服务器，通常设置为 post，而不是 get。
<input />	表单的输入元素
属性	
type="控制类型"	输入控件的类型，可能的值包含 checkbox、hidden、radio、reset、submit、text 和 image。
name="名称"	传送给脚本时，该项使用的唯一名称。
value="值"	文本或隐含项的默认值；对于复选框或单选按钮，是与表单一起提交的值；对于重置或提交按钮，是按钮本身的标签。
src="图像 URL"	图像的源文件。
checked="checked"	对于复选框或单选按钮，指定该项是被选中。
size="宽度"	文本区域的宽度，单位为字符。
maxlength="最大长度"	文本区域中可以输入的最大字符数。
<textarea>...</textarea>	表单输入元素多行文本，可包含默认文本。
属性	
name="名称"	要传送给脚本的名称。
rows="行数"	文本区域显示的行数。
cols="字符数"	文本区域显示的列（字符）数。
<select>...</select>	创建一个包含可能选项的下拉列表或滚动列表。
属性	
name="名称"	要传送给脚本的名称。
size="元素数"	显示的项数。如果指定了 size，则为滚动列表；如果没有指定 size，则为下拉列表。
multiple="multiple"	允许在列表中选择多项。
<option>...</option>	指定<select>元素中的选项。
属性	
selected="selected"	包含该属性的选项默认被选中。
value="值"	提交表单时，如果该选项被选中，将提交这个值。

## 18.10 问与答

**问：**我听说通过 Internet 传送信用卡号很危险。窃贼会在数据传送给我的过程中截取表单数据吗？

**答：**表单数据（和任何网页或电子邮件信息）在 Internet 的传送过程中可能被截取。如果在表单中请求信用卡号或其他敏感信息，应向运行 Web 服务器的公司询问安全表单处理。有一些可靠的技术可以消除高科技窃听的风险，但通常都必须付费——通常需要从安全服务提供商，如 Thawte (<http://www.thawte.com/>) 或 VeriSign (<http://www.verisign.com/>) 那里获得安全证书。

要截取 Internet 中传送的信息，比在餐厅或零售店中从别人背后偷看难得多。虽然如此，还是应该总是使用安全网页来处理敏感的金融信息，如信用卡号，尤其是在别人非常信任你的处理时。

**问：**我可以将表单放入 CD/DVD-ROM 吗？它们必须在 Internet 上吗？

**答：**可以放置网页的地方就可以放置表单。如果表单在硬盘或 CD-ROM，而不是 Web 服务器上，则不管填写者是否连接到 Internet 都可以填写它们。当然，单击提交按钮时，他们必须连接到 Internet（或本地 Intranet），否则无法将信息传送给你。

**问：**我还是不太理解何时使用脚本来处理表单，何时使用 Email。根据什么来选择？

**答：**是否使用服务器脚本来处理表单数据主要取决于是否有这样的脚本可用。与通过 Email 接收原始表单数据相比，表单处理脚本是一种更好的方法，且在用户端也更容易处理。然而，你可能没有 Web 服务器的访问权限，即使有，也可能找不到正好能够完成所需处理的脚本。在这种情况下，应该询问 ISP 或 Web 服务器管理员，确定可使用的表单处理方式。

## 18.11 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 18.11.1 测验

1. 编写 HTML 代码来创建一个访客本表单，它询问姓名、性别、年龄和电子邮件地址。假定你在/cgi/generic 有一个表单处理脚本，需要包含如下隐藏输入元素来告诉脚本将这个表单结果传送到哪里：

```
<input type="hidden" name="mailto" value="you@yoursite.com" />
```

2. 如果创建了一幅名为 sig-in.gif 的图像，如何将其用作问题 1 中访客本的提交按钮？

## 18.11.2 答案

1.

```
<html>
  <head>
    <title>My Guestbook</title>
  </head>

  <body>
    <h1>My Guestbook: Please Sign In</h1>
    <form method="post" action="/cgi/generic">
      <p>
        <input type="hidden" name="mailto" value="you@yoursite.com" />
        Your name: <input type="text" name="name" size="20" /><br />
        Your sex: <input type="radio" name="sex" value="male" /> male
        <input type="radio" name="sex" value="female" /> female<br />
        Your age: <input type="text" name="age" size="4" /><br />
        Your e-mail address: <input type="text" name="email" size="30" />
        <br />
        <input type="submit" value="sign in" />
        <input type="reset" value="erase" />
      </p>
    </form>
  </body>
</html>
```

2. 将<input type="submit" value="Sign In" />替换为<input type="image" src="sign-in.gif" alt="Sign In" />

## 18.12 练习

1. 创建一个使用各种输入元素和选择列表的表单，以确保理解它们的工作原理。
2. 创建一个 PayPal Buy Now 按钮以便通过网站上销售商品。你可能拥有世界技术含量最高的柠檬水摊位 (lemonade stand)。

## 第 19 章

# 在网页中嵌入多媒体

多媒体是一个流行术语，表示声音、视频和交互式动画。本章将介绍如何在网页中包含多媒体。

首先要知道，Internet 多媒体技术仍在发展中。计算机多媒体总的来说相对较新，但 Internet 多媒体更新，因此还处于不断变化的阶段。Internet 多媒体的快速增长给要在网页中包含音频和视频材料的人带来了三个障碍：

- 有许多种不兼容的多媒体文件格式可供选择，但还没有一个成为业界标准；
- 有些人的 Internet 连接速度不够快，接收高质量音频和视频数据需要等待很长时间；
- 用于在网页中包含多媒体的 HTML 标签很不一致，XHTML 提供了一种标准方式使其更简单。

这里要说明的是，不管你现在如何实现多媒体网站，要准备随着多媒体文件格式和技术的发展对网站进行修改。

好消息是，现在可以在一定程度上绕过这些障碍，且这些障碍在不久的将来可能变得更容易克服。本章首先介绍如何将多媒体放在网页上，最大程度地与所有 Web 浏览器兼容，不管是新浏览器还是老浏览器。另外还将介绍将多媒体放在网页上的标准 XHTML 方法，这种方法清晰地描绘了 Web 多媒体的未来。

要了解如何将多媒体放在网页中，必须有一些多媒体内容。

创建任何一种多媒体内容都可能是复杂而充满挑战的任务。如果计划从头开始创建多媒体内容，需要远比本书多的知识，才能成为优秀的多媒体开发人员。然而，如果已经有一些内容，本章将教你如何将其放到网页中。

对于不是很有艺术才能的人，有几种方法可获得有用的多媒体材料。除很显然的方式（如雇用艺术家）外，可参考下列建议。

1. Web 本身充斥各种有用的多媒体内容，现在网上有各种形状和大小的媒体库。有关

一些最好的媒体资源网站，请参阅 Sams Publishing 网站的热点列表。

2. 不想花钱？Internet 上的许多材料都是免费的。当然，向媒体内容的作者或当前拥有人核实是个不错的主意，你肯定不希望因为版权问题而被起诉。另外，美国政府的各个机构生成的内容，从法律上说属于所有美国人（例如，NASA 的在线电影胶片可以免费使用）。

3. 许多搜索引擎（google.com, yahoo.com 和 lycos.com 等）都能够搜索多媒体文件。只要谨慎对待版权问题，通过这种方式很容易找到与特定主题相关的多媒体。

4. 参考面向电视录像制作人的在线论坛、Usenet 新闻组和 RSS 新闻。尽可能清楚地描述你的网站以及网站的目的和功能，很可能找到一些未来的艺术家，他们很乐于让别人预先使用他们的作品。



## 19.1 将多媒体内容放到网页中

下面几小节介绍如何以两种方式在网页中添加音频和视频：

- 最大程度地与所有网页浏览器（包括很老式浏览器）兼容的“老式方法”；
- 只适用于最新版本的网页浏览器的“新方法”，但却是当前和将来的标准方法。

**注意：**在本章的示例网页中，我使用了 QuickTime 视频（.mov）、Windows Media 视频（.wmv）、RealAudio 流（.ra、.ram）和 MIDI（.mid）声音文件。这主要是为了与非 Windows 计算机更好地兼容。然而，也可以使用 Windows AVI 或 MPEG 视频文件、MP3、Windows Media Audio 或 AAC（iTunes）。好消息是，无论使用哪种多媒体格式，将多媒体嵌入到网页中的步骤都与这里介绍的相同。

**By the Way**

## 19.2 链接到多媒体文件

要将视频或音频文件包含到网站中，最简单、最可靠的方法是使用<a href>链接它，就像链接到另一个 HTML 文件一样（有关<a>标签请参见第 3 章）。

例如，下面一行代码可用于提供曲棍球游戏的 AVI 视频：

```
<a href="hockey.mov">View the hockey video clip.</a>
```

当用户单击“View the hockey video clip.”时，QuickTime 视频文件 hockey.mov 将传输到其计算机上。他安装的助手应用程序或插件将在文件下载完毕时自动启动。如果没有找到与 AVI 兼容的助手程序或插件，网页浏览器将为用户提供一个下载适当插件或将视频保存到硬盘中供以后浏览的机会。

**注意：**助手应用程序是指网页浏览器调用来显示自己不能处理的任何类型的文件的外部程序。一般来说，当网页浏览器自己不能显示某种文件类型时，

**By the Way**



将调用与这种文件类型相关联的助手应用程序。插件是直接安装在网页浏览器中的特殊助手应用程序，让用户能够在浏览器窗口中直接浏览多媒体内容。

程序清单 19.1 列出了一个网页的代码，它使用简单图像链接来播放 QuickTime 视频。这是第 9 章池塘示例的修改版本。

#### 程序清单 19.1 使用<a>标签 GIF 动画图像链接到 QuickTime 视频

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael's Pond</title>
  </head>

  <body style="background-image:url(water.jpg)">
    <p style="text-align:center">
      
    </p>
    <p>
      My backyard pond is not only a fun hobby but also an ongoing home
      improvement project that is both creative and relaxing. I have numerous
      fish in the pond, all Koi from various places as far as Japan, Israel,
      and Australia. Although they don't bark, purr, or fetch anything other
      than food, these fish are my pets, and good ones at that. The pond was
      built in a matter of weeks but has evolved over several years through
      a few different improvements and redesigns. I still consider it a work in
      progress, as there are always things to improve upon as I continue to
      learn more about how the pond ecosystem works, how to minimize
      maintenance, and how to get a more aesthetically pleasing look.
    </p>
    <p style="text-align:center">
      <a href="pond1.jpg"></a>
      <a href="pond2.jpg"></a>
      <a href="pond3.jpg"></a>
      <a href="pond4.jpg"></a>
    </p>
```

```
<p style="text-align:center">
  To view a video of my fish playing in the pond, click the video below:
  <br />
  <a href="pond.mov"></a>
</p>
</body>
</html>
```

这段代码中的大部分都来自原来的池塘示例。然而，下列代码是新的：

```
<a href="pond.mov"></a>
```

这些代码将 GIF 动画图 projector.gif 用作链接，指向 QuickTime 视频剪辑 pond.mov。图 19.1 显示了包含放映机图像的池塘网页。

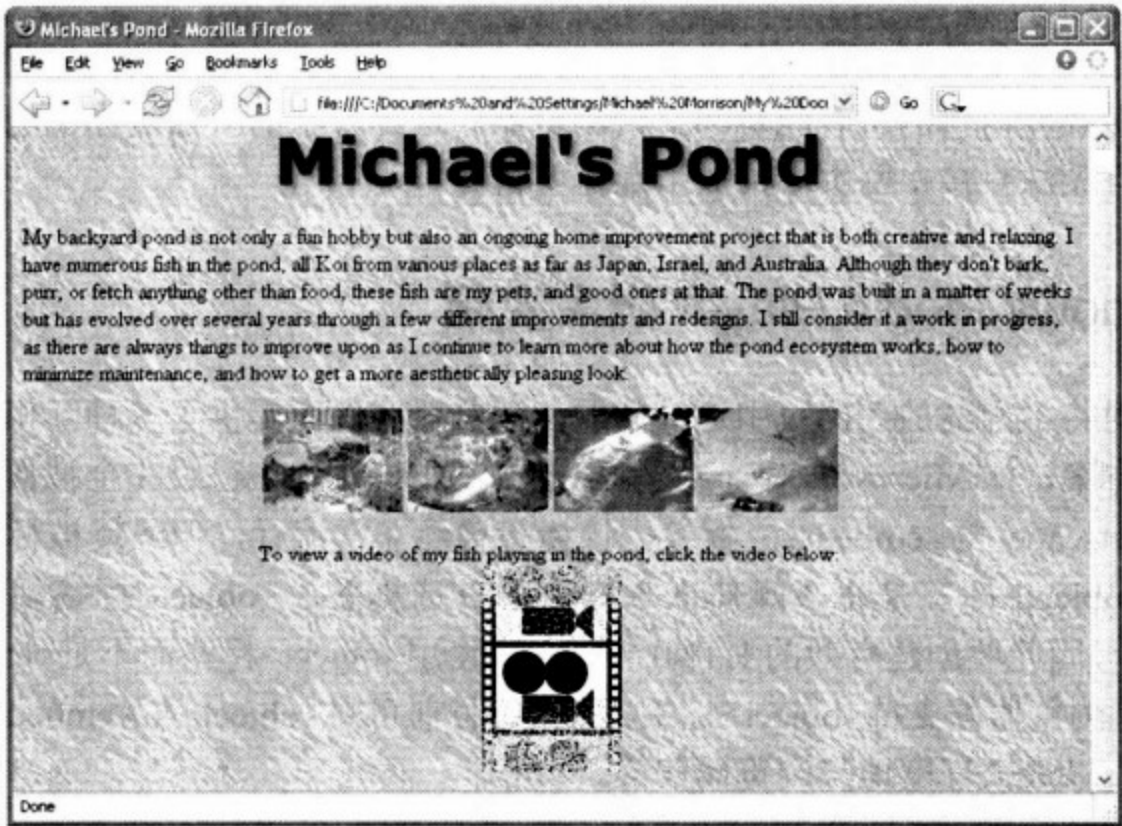


图 19.1  
将 GIF 动画图像 projector.gif 作为链接，指向 QuickTime 视频

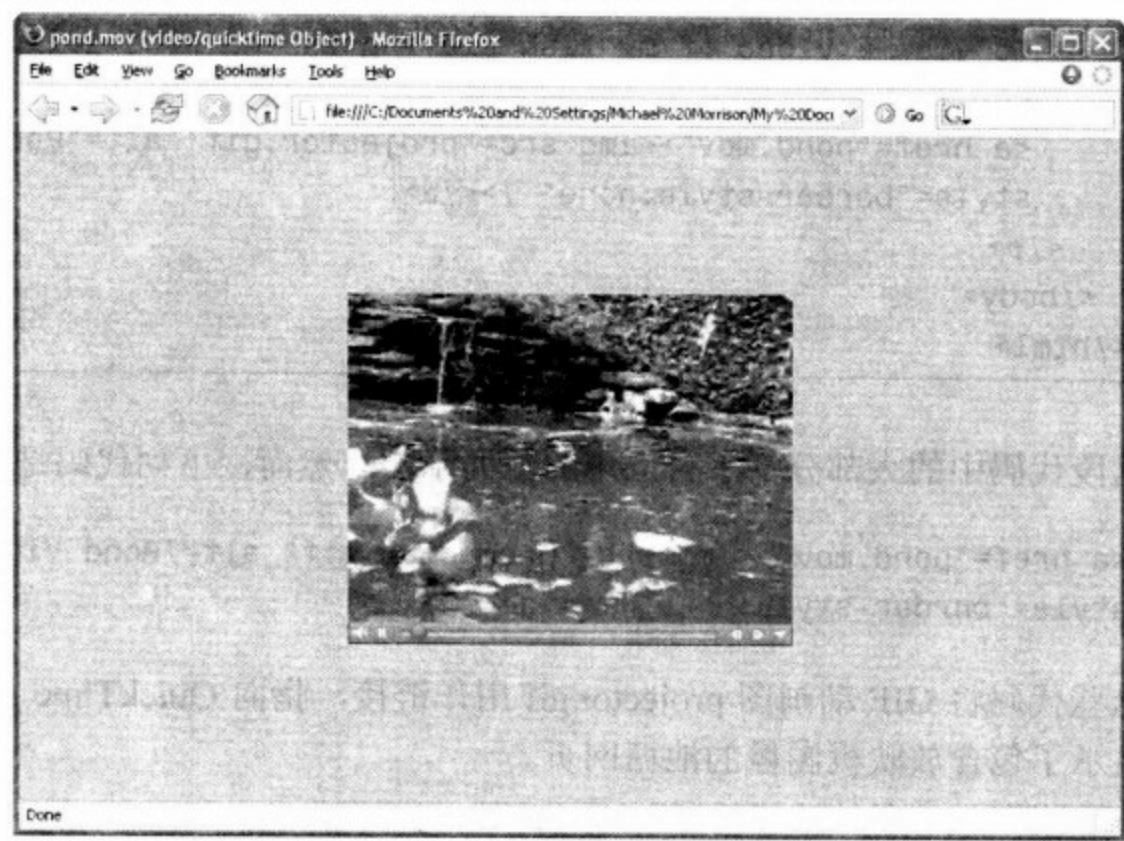
要浏览这段视频，只要单击放映机动画图像。如果有可以播放视频插件，浏览器将在其中播放视频，否则将其交给合适的助手程序。图 19.2 显示了 Mozilla Firefox 使用插件播放该 QuickTime 视频剪辑。

**注意：**如果你的浏览器不支持 QuickTime，可从 Apple 网站免费下载 QuickTime 播放器，其网址为 <http://www.apple.com/quicktime/>。即使安装了 QuickTime，有些浏览器对 QuickTime 视频的播放也将随是否安装了插件而异。例如，在我的 Windows 计算机上，Internet Explorer 和 Firefox 都直接在浏览器窗口中通过插件播放 QuickTime，而 Opera 将 QuickTime 作为助手程序启动。

By the Way

图 19.2

用户单击放映机图像链接时，QuickTime 浏览器插件将播放 QuickTime 视频 pond.mov



你可能猜到了，这种使用简单链接来播放多媒体文件的方法提供了最好的向后兼容，因为它将确定如何播放多媒体剪辑的所有职责都交给浏览器。这种方法的缺点是你不能控制多媒体剪辑如何播放，且无法直接在网页中播放多媒体剪辑。

### 19.3 使用 Windows Media Player 嵌入视频

在过去几年，针对将多媒体放到网页中这一问题，浏览器开发商提供了多种互相冲突的解决方案。其中有些方案，如 Microsoft 对<img>标签的专用扩展，现在已经完全被摒弃。另一个非标准标签是 Netscape 的<embed /> 标签，该标签曾经非常流行，但最终被标准标签<object>取代，现在<object>标签是将多媒体嵌入网页的首选标签。<object>标签现在也是 XHTML 标准的一部分，因此要创建有效的 HTML 网页，应使用<object>标签而不是<embed /> 标签。然而，不是所有浏览器都支持<object>标签，因此，目前可将<object>与<embed />这两个标签结合起来使用，以最大程度地与浏览器兼容。

以前，<object>标签的问题之一是，它很久后才被网页浏览器接受，在不同的浏览器中得到一致的支持。这是因为 Internet Explorer 在<object>标签标准化很久前就支持它，而且包括几个专用属性，这些属性当时在 Netscape Navigator 或其他浏览器中并没有得到支持。虽然这些不一致性已经解决，但现在仍不可能只使用<object>标签就在不同的浏览器中取得一致，稍后将更详细地介绍如何解决这个问题。

下面的代码只使用<object>标签嵌入前面的池塘视频：

```
<object classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6" width="320"
height="305">
  <param name="type" value="video/x-ms-wmv" />
  <param name="URL" value="pond.wmv" />
  <param name="uiMode" value="full" />
```



```
<param name="autoStart" value="true" />
</object>
```

如果考虑到这段代码将一段视频嵌入到网页中（见图 19.3），它并不很复杂。最复杂的部分是<object>标签的 classid 属性，它设置为一个很长的字母数字编号。这个编号是 Microsoft Windows Media Player 的全局 ID（global ID），它告诉<object>标签将 Windows Media Player 嵌入网页中以播放视频剪辑。你可以将这段代码复制并粘贴到自己的网页中。

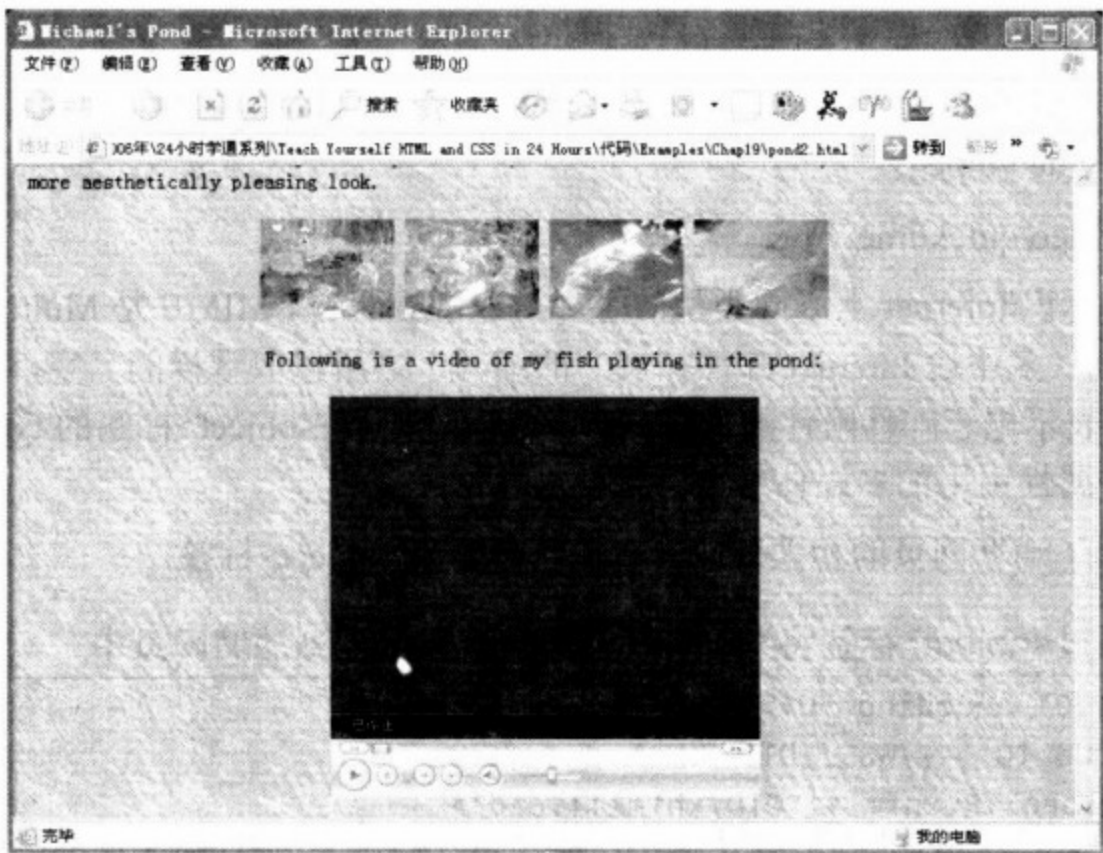


图 19.3

<object> 标签让你能够在网页中嵌入视频剪辑，并指定用于播放它的媒体播放器

**注意：**需要指出的是，Media Player 是一个复杂的媒体播放器，能够自动将多媒体流化，即下载开始的一两秒钟内容后就开始播放。其他内容在播放前面的内容的同时在后台下载。这样访问者在浏览或收听多媒体内容时不必长时间地等待下载完毕。

By the Way

<object>标签的 width 和 height 属性决定嵌入的 Windows Media Player 窗口的大小。如果没有设置这两个属性，有些浏览器将自动调整窗口的大小以适应内容，而有些将不显示内容——因此要安全地播放，应将这两个属性设置为多媒体内容播放时的大小。

<object>标签内有四个<param>标签，负责指定关于如何播放剪辑的更多细节。每个<param>标签都有两个属性：name 和 value，负责将数据（value）同特定设置（name）关联起来。在这个例子中，媒体剪辑的 URL 设置为 pond.mov。第三个参数 uiMode 决定 Windows Media Player 可用哪些按钮和用户界面选项：full 表示所有用户界面元素都可用，如控制按钮和音量滑块。最后，autoStart 参数设置为 ture，这样，当页面在浏览器中打开时，视频剪辑便自动开始播放。

我将最复杂的参数留在最后：type 参数指定要播放的媒体类型，这里为 Windows Media Video（WMV）文件。媒体类型必须指定为标准的 Internet MIME 类型之一，你可能听说过 MIME 类型。下面是几种流行的可在网页中使用的 MIME 声音和视频格式。

- WAV 音频: audio/x-wav。
- AU 音频: audio/basic。
- MP3 音频: audio/mpeg。
- MID 音频: audio/midi。
- WMA 音频: audio/x-ms-wma。
- RealAudio: audio/x-pn-realaudio-plugin。
- AVI: video/x-msvideo。
- WMV: video/x-ms-wmv。
- MPEG 视频: video/mpeg。
- QuickTime: video/quicktime。

MIME 类型是惟一标识 Internet 上不同类型的媒体对象的标识符。MIME 是 Multipurpose Internet Mail Extensions (多用途 Internet 邮件扩展) 的缩写, 之所以取这样的名字, 是因为 MIME 类型原来是用于表示电子邮件附件的。这些 MIME 类型应在<object>标签的 type 属性中使用, 用于标识 data 属性引用的多媒体对象类型。

程序清单 19.2 显示了池塘网页的相关代码, 从中可以看到<object>标签。

#### 程序清单 19.2 通过<object>标签将 WMV 视频剪辑直接嵌入到池塘网页中

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Michael's Pond</title>
  </head>

  <body style="background-image:url(water.jpg)">
    ...
    <p style="text-align:center">
      Following is a video of my fish playing in the pond:
    </p>
    <p style="text-align:center">
      <object classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6" width="320"
        height="305">
        <param name="URL" value="pond.wmv" />
        <param name="uiMode" value="full" />
        <param name="autoStart" value="true" />
        <embed width="320" height="305" type="video/x-ms-wmv"
          src="pond.wmv" controls="All" loop="false" autostart="true"
          pluginspage="http://www.microsoft.com/windows/windowsmedia/" />
        </object>
      </p>
```



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

**注意：**由于 XHTML 不支持 `<embed />` 标签，因此使用该标签的网页是无效的。不幸的是，这个问题实在没有其他解决方法——只能等待浏览器完全支持 `<object>` 标签。

**By the Way**

稍等一下，还有一些代码没有出现在前面的 `<object>` 标签示例中。不幸的是，正如本章前面提到的，并非所有网页浏览器对 `<object>` 标签的支持都完全一致。实际上，如果使用 `<object>` 标签来嵌入媒体播放器，Opera 不会显示任何内容。因此，必须在 `<object>` 标签内包含 `<embed />` 标签，以处理浏览器的不一致性。这并不是理想的解决方法，但只能这样做，因为浏览器开发商迟迟不能在标准上达成一致。如果仔细研究，将发现 `<object>` 标签包含的信息在 `<embed />` 标签也都包含。

**注意：**本课的剩余部分将继续使用 `<embed />` 标签，以确保在所有主要浏览器中获得最大的兼容性。在将来希望能够只使用 `<object>` 标签，而不再需要 `<embed />` 标签。

**By the Way**

这里介绍的 `<object>` 标签的用法比较简单，这是因为仅播放多媒体内容并不需要知道如何使用 `<object>` 标签的更复杂方面。换句话说，要在网页中共享多媒体剪辑，并不需要成为多媒体专家。

## 19.4 使用 RealPlayer 嵌入音频

上一节介绍了如何使用 `<object>` 标签将 Windows Media Player 嵌入到网页中，以便能够播放视频剪辑。然而，如果访问网站的用户使用其他媒体播放器，如 RealPlayer 呢？将 `<object>` 标签中的 `classid` 值修改为适当的播放器，就可以使用另一个播放器。

**注意：**RealAudio 是 RealPlayer 媒体播放器推广的一种流音频格式。要下载 RealPlayer 的免费版本，可访问 <http://www.real.com/player/>。

**By the Way**

下列代码演示了如何使用 RealPlayer 播放 RealAudio 音频剪辑：

```
<object classid="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBCCFA" width="290"
height="65">
  <param name="type" value="audio/x-pn-realaudio-plugin" />
  <param name="src" value="http://wiredforbooks.org/alice/chapter1.ram" />
  <param name="controls" value="All" />
  <param name="loop" value="false" />
  <param name="autostart" value="true" />
  <embed width="290" height="65" type="audio/x-pn-realaudio-plugin"
src="http://wiredforbooks.org/alice/chapter1.ram" controls="All"
loop="false" autostart="true" pluginspage="http://www.real.com/player/" />
</object>
```

在这个例子中，RealAudio 剪辑 chapter1.ram 使用嵌入的 RealAudio 播放器播放。这里用到了几个<param>标签，旨在正确地初始化播放器。例如，type 参数设置了 RealAudio 的标准 MIME 类型，这让播放器知道要播放的是哪种媒体。

你可能注意到了，其他参数并不与之前嵌入 Windows Media Player 使用的参数完全相同。这是因为每种媒体播放器都可以定义自己的参数。幸运的是，不同媒体播放器的参数基本类似。

RealPlayer 的 src 参数标识多媒体内容的 URL，而 controls 参数类似于 Windows Media Player 控件中的 uiMode。参数 loop 和 autostart 控制剪辑是否重复循环播放以及是否在页面下载时就立即开始播放。

相信读者还注意到了，我在<object>标签中包含了<embed />标签，以兼容较不一致的老式网页浏览器。

程序清单 19.3 列出了出现在真实网页中的 RealPlayer 控件。

#### 程序清单 19.3 使用<object>标签将 RealAudio 音频剪辑直接嵌入到网页中

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Alice's Adventures in Wonderland</title>
  </head>

  <body>
    <div>
      <h1>Alice's Adventures in Wonderland</h1>
      <h2>By Lewis Carroll</h2>
      <p>
        This audio production of Lewis Carroll's "Alice's Adventures in
        Wonderland" was made possible by <a href="http://wiredforbooks.org/">
        Wired for Books</a>. To listen to the complete recording, visit the
        <a href="http://wiredforbooks.org/alice/">Alice page</a> at the Wired
        for Books Web site.
      </p>
      <p>
        <object classid="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA"
        width="290" height="65">
          <param name="type" value="audio/x-pn-realaudio-plugin" />
          <param name="src"
value="http://wiredforbooks.org/alice/chapter1.ram" />
          <param name="controls" value="All" />
          <param name="loop" value="false" />
          <param name="autostart" value="true" />
          <embed width="290" height="65" type="audio/x-pn-realaudio-plugin">

```

```
src="http://wiredforbooks.org/alice/chapter1.ram" controls="All"
loop="false" autostart="true"
pluginspage="http://www.real.com/player/" />
</object>
</p>
</div>
</body>
</html>
```

图 19.4 显示了 RealAudio 剪辑“艾丽斯漫游仙境”通过嵌入的 RealPlayer 控件播放的情况。这个剪辑可从俄亥俄大学的 Wired for Books 网站获得，其网址为 <http://www.wiredforbooks.org/>。

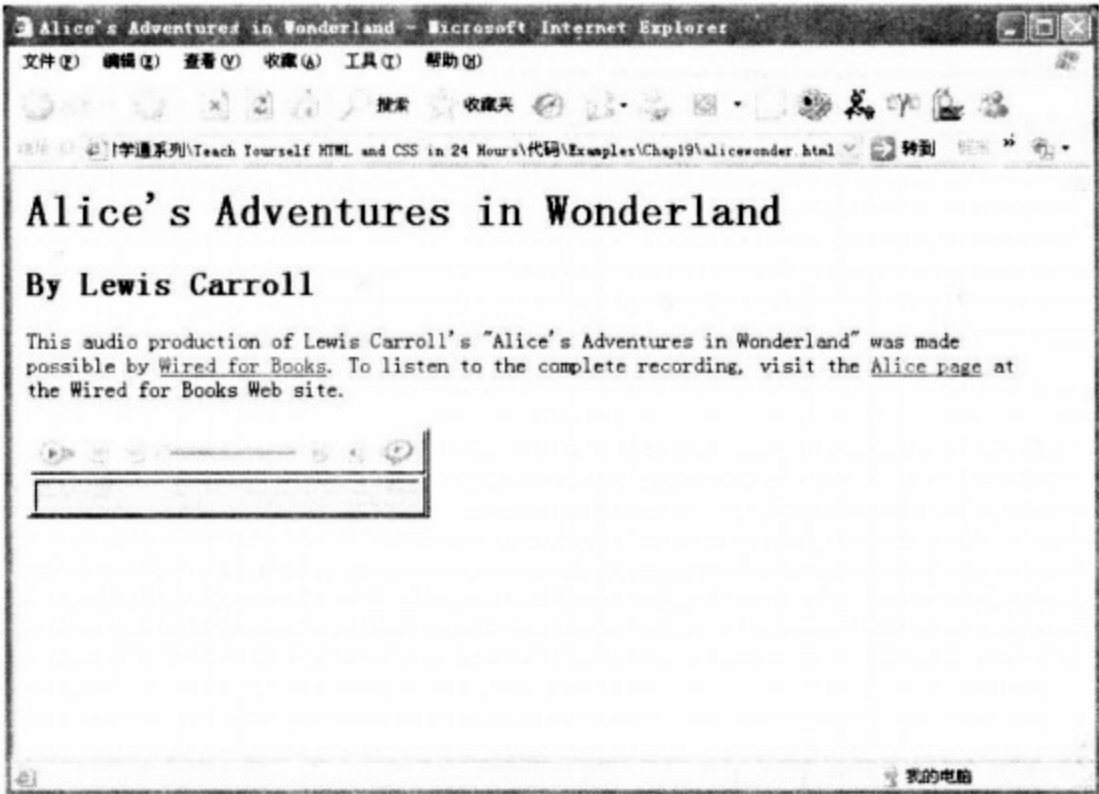


图 19.4  
结合使用标签 `<object>` 和 `<embed/>` 可播放嵌入音频，包括可朗读的畅销书音频

这个例子说明了如何使用网页来听有声音的书，这可能是你从来没有想到的。甚至还可以录制自己的音频并将它放到网上，然后以这样的方式播放。许多公司和组织正在使用这样的方式在其网站上发布定期的音频计划消息。

**提示：**设置了网页中其自动播放音频时，应让用户能够关闭声音（再比没有比听着最喜欢的 CD 在 Web 冲浪时，突然打开一个不能关掉音频的网页更烦人的了）。在“艾丽斯漫游仙境”示例中，所有 RealPlayer 媒体控件都是可用的，让用户能够控制如何播放音频。

**Did you Know?**

## 19.5 自动嵌入合适的播放器

至此，你已经知道如何使用 `<object>` 标签（和 `<embed />` 标签）在网页中嵌入媒体播放器。问题是，必须对典型用户可以使用哪些播放器做出假设。大部分 Windows 用户都安装了 Windows Media Player，而 Apple 用户通常安装了 QuickTime、RealPlayer 或两者兼有。

一种使多媒体网页能够适合于每个用户的方法是，根据用户的操作系统使用合适的媒体播放器。一种合理的假设是，Windows 使用 Windows Media Player，其他操作系统（Mac、Linux 等）使用 RealPlayer。幸运的是，可以用一行简单的 JavaScript 代码来检查操作系统。第 17 章介绍了 JavaScript 以及如何在网页中使用 JavaScript。

下面这行 JavaScript 代码判断用户的操作系统是否是 Windows：

```
navigator.appVersion.indexOf("Win")
```

### Did you Know?

**提示：**要检查使用的是否是 Macintosh 操作系统，可使用 `navigator.appVersion.indexOf("Mac")`。

这行代码的结果为 true 或 false，因此可放在 if 语句的条件中，如下所示：

```
if (navigator.appVersion.indexOf("Win") != -1)
    // Play a media file using Windows Media Player
else
    // Play a media file using RealPlayer
```

### By the Way

**注意：**我在我开发的在线音乐游戏 Guess That Groove (<http://www.guessthatgroove.com/>) 中使用了类似的 JavaScript 代码，这个网站也大量依赖于嵌入媒体。

现在有了在不同媒体播放器间切换所需要的代码。程序清单 19.4 在一个播放 Amazon.com 试听音乐的网页中使用了这段代码，使用的媒体播放器取决于用户的操作系统。

**程序清单 19.4** JavaScript 代码智能地选择 Windows Media Player 或 RealPlayer 来播放 Amazon.com 试听音乐

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Tom Waits on Amazon.com</title>
  </head>

  <body>
    <div style="text-align:center">
      <h1>Tom Waits - The Heart of Saturday Night</h1>
      <p>
        This song is titled "New Coat of Paint," and appears on the Tom Waits
        CD, "The Heart of Saturday Night."<br />
        This CD is available on Amazon.com via the following link:
      </p>
      <p>
```

```

<iframe src="http://rcm.amazon.com/e/cm?t=michaelmorris-20&o=1&p=8&
l=as1&asins=B000002GXS&fc1=000000&=1&lc1=0000ff&bc1=ffffff&&#108;&
#116;1=_top&IS2=1&bg1=ffffff&f=ifr" style="width:120px;height:240px;"
scrolling="no" marginwidth="0" marginheight="0" frameborder="0"
style="text-align:center"></iframe>
</p>
<p>
<script type="text/javascript">
<!-- Hide the script from old browsers
if (navigator.appVersion.indexOf("Win") != -1)
    document.write("<object
        classid='CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6' width='290'
        height='65'><param name='type' value='audio/x-ms-wma' />
        <param name='URL' value='http://www.amazon.com/exec/obidos/
        clipserve/B000002GXS001001/0/104-1401177-1194303' /><param
        name='uiMode' value='full' /><param name='autoStart'
        value='true' /><embed width='290' height='65'
        type='audio/x-ms-wma' src='http://www.amazon.com/exec/obidos/
        clipserve/B000002GXS001001/0/104-1401177-1194303' controls='All'
        loop='false' autostart='true'
        pluginspage='http://www.microsoft.com/windows/windowsmedia/' />
        </object>");
else
    document.write("<object
        classid='clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA' width='290'
        height='65'><param name='type'
        value='audio/x-pn-realaudio-plugin' /><param name='src'
        value='http://www.amazon.com/exec/obidos/clipserve/
        B000002GXS001001/1/104-1401177-1194303' /><param name='controls'
        value='All' /><param name='loop' value='false' /><param
        name='autostart' value='true' /><param name='prefetch'
        value='false' /><embed width='290' height='65'
        type='audio/x-pn-realaudio-plugin' src='http://www.amazon.com/
        exec/obidos/clipserve/B000002GXS001001/1/104-1401177-1194303'
        controls='All' loop='false' autostart='true'
        pluginspage='http://www.real.com/player/' /></object>");
    // Stop hiding the script -->
</script>
</p>
</div>
</body>
</html>

```

这段代码看上去很复杂，实际上并非如此。第一个比较复杂的部分是由 Amazon.com 自动生成的一大块代码，我指的就是<iframe>标签内的代码，这是由 Amazon.com 的 Build-A-Link 工具生成的代码，其网址为 <http://associates.amazon.com/gp/associates/network/build-links/main.htm>。这个工具很方便，让你能够快速、轻松地链接到 Amazon.com 的产品。在这个例



子中，我使用了这个工具，因为这个网页播放的音频是 Amazon.com 上一张 CD 中的曲子。

By the Way

**注意：**<iframe> 标签是可包含来自其他网页的 HTML 代码的“浮动”的窗口。框架在第 16 章中有更详细的介绍。在这里读者只需知道<iframe> 标签用于插入一个“网页”，该网页包含指向 Amazon.com 中 CD 的链接。

Amazon.com 音乐播放器示例中另一个比较复杂的地方是，根据操作系统动态选择不同播放器的代码。JavaScript 代码必须为合适的媒体播放器生成 HTML 代码，这看上去比较混乱，因为所有代码都必须在连续的文本中。最终结果是一个嵌入的媒体播放器对象，如图 19.5 所示。

图 19.5  
根据操作系统是 Windows 还是其他操作系统来选择并使用媒体播放器



Coffee Break

**休息一下：**现在可以探索一下 Amazon.com，看看它是如何在网页中使用流音频的。访问 <http://www.amazon.com/>，搜索喜欢的音乐。仔细观察网站是如何让你能够收听 CD 上的单曲的。这是一个很好的例子，说明多媒体内容不仅使网页更有趣，还可以帮助人们在 Amazon.com 购物。

## 19.6 总结

本章读者学习了如何在网页中嵌入视频与声音。知道了如何使用一个指向多媒体剪辑的简单链接，这种播放多媒体内容的方法得到了最广泛的支持，但最不灵活。然后学习了如何使用<object> 标签直接在网页中嵌入媒体播放器。不仅如此，还了解到，为最大程度地与所有浏览器兼容，可结合使用标签<embed />和<object>。标签<object>和<embed />可用于包含各种媒体类型，除前面使用过的 AVI、WMV 和 QuickTime 外，还有 WAV、MP3、RealAudio 和 MIDI 文件，而这只是其中的一小部分。

表 19.1 总结了本章介绍的标签。

表 19.1第 19 章介绍的 HTML 标签与属性

标签/属性	功能
<embed />	嵌入由插件程序读取或显示的多媒体文件；这个标签从技术上说已被摒弃，但由于浏览器并不都支持<object>标签，所以仍然很有用。
属性	
width="宽度"	嵌入对象的宽度，单位为像素。
height="高度"	嵌入对象的高度，单位为像素。
type="mimetype"	多媒体内容的 MIME 类型。
src="媒体 URL"	嵌入文件的 URL。
control="控件"	媒体播放器的用户输入控件的配置；若使用 All，则所有控件都可用。
loop="循环"	媒体剪辑只播放一次还是多次播放；设置为 true 或 false。
autostart="自动开始"	是否在打开网页时就开始播放媒体剪辑；设置为 true 或 false。
pluginspage="插件 URL"	播放媒体所需插件的 URL。
<object>...</object>	在文档中插入图像、视频、Java 程序、ActiveX 控件或其他对象。关于<object>标签的属性，请参见第 18 章。
<param>...</param>	对象的运行阶段设置，如在网页中占据的宽度与高度。
属性	
name="名称"	名称参数属性
value="值"	名称参数属性的值。

## 19.7 问与答

问：我听说很多关于流式视频与音频的内容，它们指的是什么？

答：过去，通过大部分调制解调器下载视频与音频文件都需要花很多分钟甚至几章，这样严重地限制了视频与音频在网页中的应用。所有人的目标都是朝流式视频与音频发展，它可以一边接收数据一边播放。也就是说，没有完全下载剪辑时，就可以开始看视频或听音频。

现在流式播放在大部分媒体播放器中得到了广泛支持，有独立版本也有插件版本。使用<object>标签嵌入媒体对象时，支持它的媒体播放器将自动地以流方式播放媒体剪辑——如果这个播放器支持流。

问：应如何在视听文件格式 QuickTime、Windows AVI/WAV、RealVideo/RealAudio 和 MPEG 之间做出选择？它们之间有很大的区别吗？

答：QuickTime 是 Macintosh 用户最常用的视频格式，但也有用于 Windows 的 QuickTime 播放器。同样，AVI/WMV 和 WAV/WMA 是 Windows 用户首选的视频和音频格式，但也有支持这些格式的 Macintosh 播放器。MPEG 是另外一个流行的视频与音频标准。MPEG-3 简称为 MP3，是极其流行的高保真度音频标准。另一种基于 MPEG 的音频格式是 Apple 的 AAC 格式，读者可能知道它是 iTunes 使用的音乐格式。

如何选择呢？如果大部分用户都使用 Windows，就选择视频格式 AVI/WMV 和音频格式 WAV/WMA。如果用户中很大一部分是 Macintosh 用户，就选择 QuickTime 或 RealVideo/RealAudio，或至少将其作为一种可选择的方式；对于 Mac 音频来说，MP3 也是可

行的选择。如果跨平台兼容性非常重要，考虑使用 MP3 作为音频格式，RealVideo/RealAudio 作为视频格式——但要观看 RealVideo/RealAudio 剪辑，必须从 <http://www.real.com/player/> 下载专门的软件。

## 19.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 19.8.1 测验

1. 要让尽可能多的用户能够观看你的网站中的视频，最简单的方法是什么？
2. 编写 HTML，在网页中嵌入名为 myvideo.avi 的视频，使所有主要网络浏览器的用户都能够播放它。视频需要在网页中占据 320×305 像素的区域。
3. 如何在<object>标签中加入<param>标签来重复播放多媒体剪辑？

### 19.8.2 答案

1. 只需这样链接它：

```
<a href="myvideo.avi">my video</a>
```

2. 由于视频剪辑是 Microsoft 格式 (AVI)，所以应嵌入一个 Windows Media Player 对象。使用下列 HTML 代码：

```
<object classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6"
width="320" height="305">
  <param name="type" value="video/x-ms-avi" />
  <param name="URL" value="myvideo.avi" />
  <param name="uiMode" value="full" />
  <param name="autoStart" value="true" />
  <embed width="320" height="305" type="video/x-ms-avi"
src="myvideo.avi" controls="All" loop="false" autostart="true"
pluginspage="http://www.microsoft.com/windows/windowsmedia/"></embed>
</object>
```

- 3.

```
<param name="loop" value="true" />
```

## 19.9 练习

1. 创作自己的视频剪辑，然后将它嵌入到网页中，如果读者是 Windows 用户，可尝试

使用免费的 Windows Movie Maker 软件，它可以从下述网址下载：<http://www.microsoft.com/windowsxp/moviemaker/>。如果使用的是 Mac 计算机，可尝试使用 iMovie，它是 iLife 应用程序套件 (<http://www.apple.com/ilife/>) 的一部分。

2. 本章介绍的嵌入媒体的方法和标签也可用于 Macromedia Flash 文件。要解如何使用 Flash 在网页中放置交互式的动画，请参见 Flash 主页，其网址为：<http://www.macromedia.com/software/flash/>。



## 第 20 章

# 使用 HTML 和 CSS 使 eBay 拍卖更有趣

eBay 自 1995 年创立以来，已成为最著名、流量最大的网站之一。作为最基本的在线市场，eBay 允许用户销售和购买任何东西，包括烟酒和人类器官。曾有整个城镇在 eBay 上拍卖，更不用说一架 Grumman Gulfstream II 喷气机只卖了 490 万美元。虽然 eBay 上肯定有一些不寻常的货品，但大多数货品还是大众市场产品，如运动商品和电子商品。

本章将介绍如何在 eBay 的拍卖清单中包含 HTML 和 CSS，有效地使平淡无奇的销售清单变得有趣。你还将学习如何创建拍卖列表的幻灯片播放，这样就可以显示多个货品的照片而不会占据太多屏幕空间。

### 练习

在将 HTML 和 CSS 应用于拍卖列表前，首先要有东西卖。翻一翻阁楼或衣柜，应该不难找出可卖的东西。一般来说，如果这个东西在 eBay 上卖不到 5 或 10 美元，就不需要找了。可以在 eBay 上搜索与你要卖的东西类似的产品，进行“市场比较分析”。这有助于估计起拍价格以及希望的最后拍卖价格，还可以帮助估计发货费用。

找到要卖的东西后，执行下列步骤将该商品放在 eBay 上。

1. 为商品拍几张照片，最好是从各个角度拍。如果是包装的产品且工厂没有将盒子封起来，可以打开盒子，拍拍里面的东西。顾客不会嫌信息太多。
2. 看看产品的制造商是否有网站，访问它的网站看看是否能够找到关于产品的更多信息。将这些信息放在 HTML 和 CSS 中，可以更好地描述产品。
3. 在 eBay 上搜索类似的商品，注意看这些商品是如何展示的以及它们的起拍价格和条件。如果你的商品是可收藏的，比现在已有的商品更好，要进行说明。

### Did you Know?

**提示：**确定商品是否值得在 eBay 上拍卖的一个好方法是搜索，单击屏幕左边的 Completed Listings，然后单击它下面的 Show Items 按钮。搜索结果将只显示那些已经结束的拍卖。绿色价格是实际卖出的价格，红色价格是没有卖出的价格——只关注绿色价格，了解一下商品卖出了什么样的价格。



4. 如果有天平，可以称称商品的重量，并将它包装好。然后估计一下发货需要多少钱。

▲

## 20.1 为什么要在 eBay 上使用 HTML 和 CSS

将商品列在 eBay 上以便拍卖时，你可能会想为什么要使用 HTML 和 CSS。eBay 只是一个拍卖网站，它当然不能为你创建网页。你习惯地认为网页是完全由你创建的；实际上，在别人的网页上创建一部分页面是可能的，这正是在 eBay 上的情况。

**注意：**如果对 eBay 一无所知，建议你访问 eBay 网站 <http://www.eBay.com>，花些时间探索一下。关于 eBay 的运作方式及如何使用它，请参阅 eBay 的学习中心 <http://pages.eBay.com/education/>。

By the  
Way

在 eBay 上列出要拍卖的商品时，可以控制商品的描述，这是拍卖列表中描述商品的部分。每个 eBay 拍卖列表都有标准的模板信息，它显示在描述前，如图 20.1 所示。



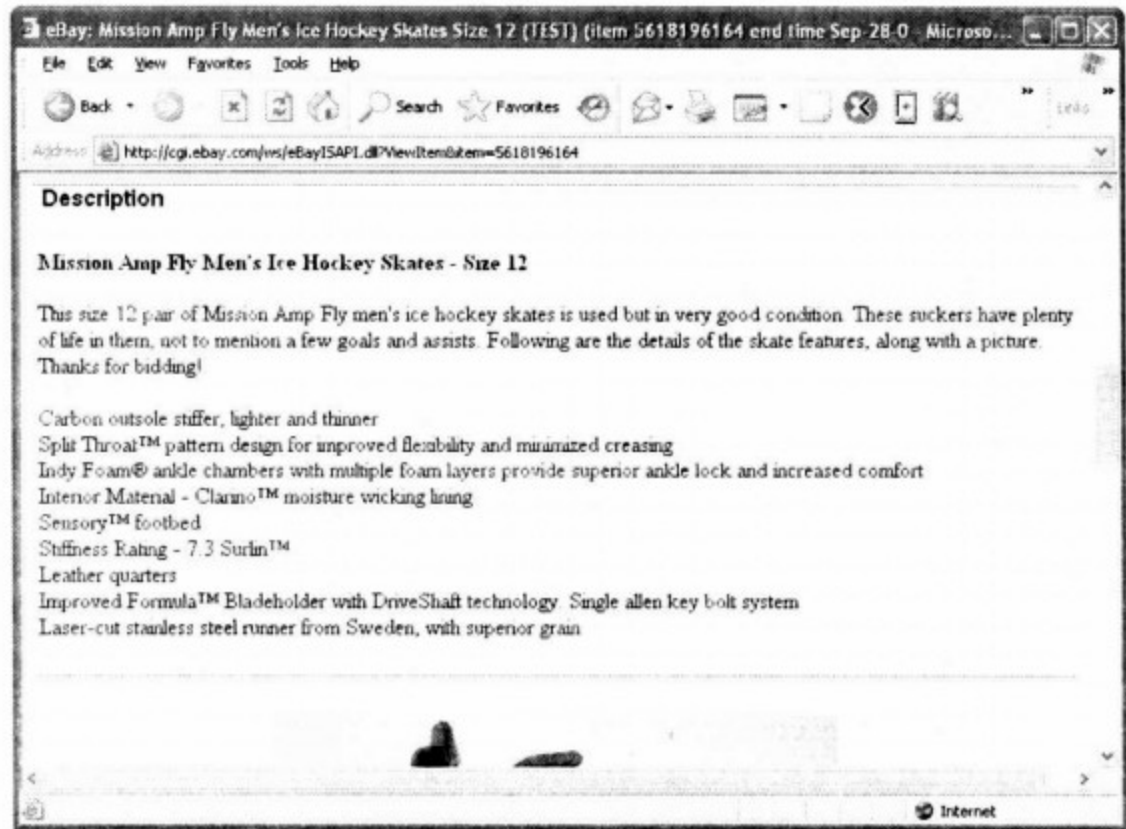
图 20.1  
每个 eBay 商品网页都以标准的模板拍卖信息开始，不能修改

eBay 商品清单网页开头的信息是在你首次展示商品时收集的，不能通过 HTML 和 CSS 编辑或修改。HTML 和 CSS 作用的地方是商品描述，它位于标准的拍卖信息下方（见图 20.2）。

在这个例子中，你看到的可能是最单调的商品描述，只包含了粗体标题和没有样式的文本。描述下方确实有幅图像，但使用没有标题和边框进行修饰。

现在翻到图 20.6，看看 HTML 和 CSS 如何使商品清单变得更有活力、对顾客更有吸引力。这是同一个拍卖列表，使用了 HTML 和 CSS 使其变得更有趣。在本章的学习过程中，你将创建图 20.6 所示的商品清单。

图 20.2  
商品描述是可用 HTML 和 CS 使其变得有趣的地方



## 20.2 理解如何将 HTML 应用于 eBay

使用 HTML 和 CSS 在 eBay 上创建新商品清单时，有很重要的一点，那就是要知道什么可以做，什么不能做。好消息是有许多 HTML 和 CSS 标签、属性和样式可用于创建突出的商品网页。然而，编写不放在你自己网站上的 HTML 代码时，有些东西必须了解。

首先，要知道编写的任何代码都只能放在 eBay 商品网页的主体中，不能放在其他地方。这意味着不能使用<meta>标签或其他只能放在网页头部的标签。这还意味着你只能使用内联样式，而不能使用内部样式表，因为<style>标签只能放在网页头部。

**By the Way** **注意：**有些网页浏览器允许将内部样式表直接放在网页主体中，但我不推荐这样做。

因为商品网页将放在 eBay 的 Web 服务器上，而不是你的 Web 服务器上，因此在引用自己的图像或网页时必须使用完整的 URL。例如，对于存储在自己的 Web 服务器上的商品图像，<img>标签中的 src 属性必须使用完整的 URL（以 http://开始）。要链接其他网页来提供更多的产品信息，务必在<a>标签的 href 属性中包含完整的 URL。

因为 eBay 网页被设计为不包含框架的网页，因此不能使用框架。因为在使用框架的网页中，<frameset>标签取代了<body>标签。在 eBay 的商品清单中，只能在<body>和</body>之间修改，因此框架显然是无法使用的。框架在第 16 章介绍过。

说到与<body>标签相关的限制，因为不能访问<body>标签，因此要修改商品网页整个页面范围的样式有点困难。例如，页面的背景通常是用<body>标签的 background-color 或 background-image 样式属性来设置的。因为这种方法无法使用了，因此必须考虑其他的方法来实现同样的效果。在 20.5 节中，将学习如何使用 JavaScript 来实现这一点。关于 JavaScript 如何应用于网页中，可参见第 17 章。

## 20.3 用 HTML 创建 eBay 商品清单

使用 HTML 和 CSS 创建 eBay 商品清单与创建其他网页很类似。实际上，首先将清单当作普通的网页来创建，然后在要列出商品以便拍卖时再将它包含到 eBay 的商品描述中。

**提示：**如果你只想测试商品网页的布局，eBay 允许创建测试的拍卖清单。不幸的是，eBay 商品清单的收费规定也适用于测试拍卖，因此不要使用太多测试。要创建测试拍卖，选择商品类别 Everything Else/Test Auctions/General”，然后在标题和描述中清楚地说明该清单只是测试。

**Did you know?**

程序清单 20.1 列出一些“骨架”代码，可用于创建和测试 eBay 商品清单网页。

程序清单 20.1 eBay 商品清单网页的 HTML 代码

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head><title></title></head>

<body>
<!-- EBAY HTML CODE STARTS HERE -->

<!-- EBAY HTML CODE ENDS HERE -->
</body>
</html>
```

可以看到，这段代码非常短，只在网页主体中 eBay 商品编码的地方包含两行代码。使用这个模板开发网页时，要记住不要将任何代码放在这些注释外，因为在实际的 eBay 商品清单网页中不允许这样做。这是一种模拟实际 eBay 清单的限制的好方法。

程序清单 20.2 列出了一个网页的代码，这是我在 eBay 上拍卖的一双冰上曲棍球滑板的商品描述。

程序清单 20.2 一个使用 HTML 和 CSS 来描述 eBay 商品的简单示例

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head><title></title></head>

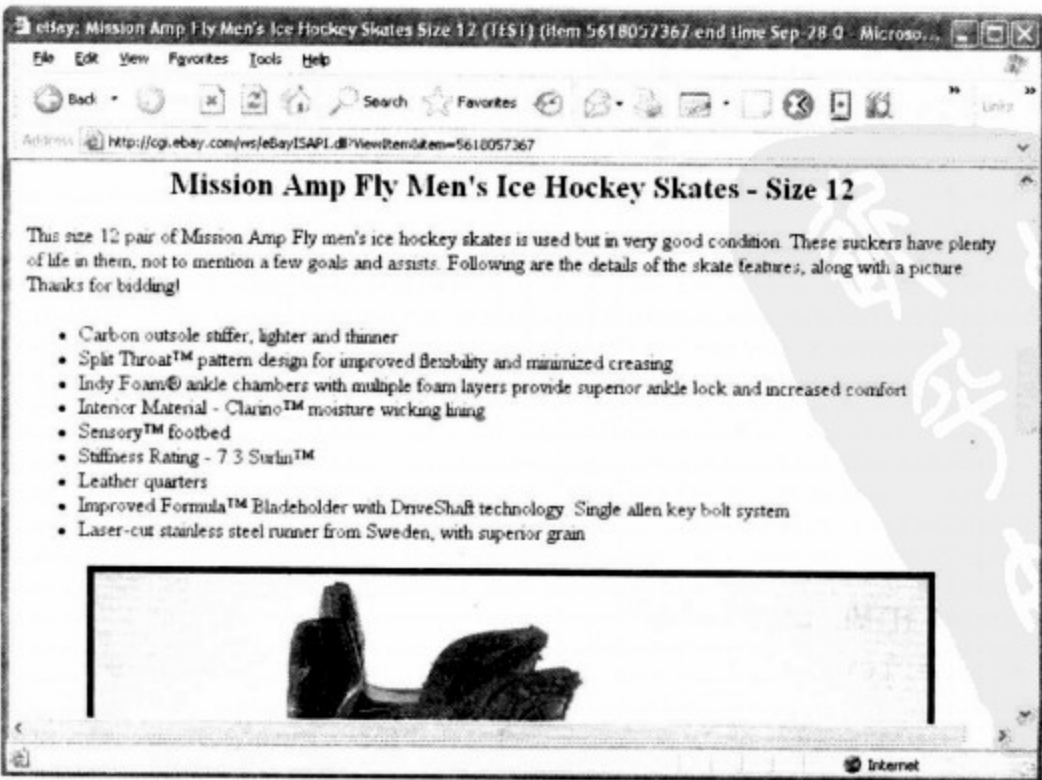
<body>
<!-- EBAY HTML CODE STARTS HERE -->
<h2 style="text-align:center">Mission Amp Fly Men' s Ice Hockey Skates - Size 12
</h2>
<div>
```



```
This size 12 pair of Mission Amp Fly men's ice hockey skates is used but in
very good condition. These suckers have plenty of life in them, not to
mention a few goals and assists. Following are the details of the skate
features, along with a picture. Thanks for bidding!
<ul>
<li>Carbon outsole stiffer, lighter and thinner</li>
<li>Split Throat&#153; pattern design for improved flexibility and
minimized creasing</li>
<li>Indy Foam&#174; ankle chambers with multiple foam layers provide
superior ankle lock and increased comfort</li>
<li>Interior Material - Clarino&#153; moisture wicking lining</li>
<li>Sensory&#153; footbed</li>
<li>Stiffness Rating - 7.3 Surlin&#153;</li>
<li>Leather quarters</li>
<li>Improved Formula&#153; Bladeholder with DriveShaft technology. Single
allen key bolt system</li>
<li>Laser-cut stainless steel runner from Sweden, with superior grain</li>
</ul>
</div>
<div style="text-align:center">
<br />
<div style="text-align:center; font-size:16pt; font-weight:bold">Side view of
skates, some scuffs but overall good shape.</div>
</div>
<!-- EBAY HTML CODE ENDS HERE -->
</body>
</html>
```

这段代码使用 HTML 和 CSS 来改进图 20.2 所示商品的纯文本版本。图 20.3 显示了程序清单 20.2 中的商品描述显示在实际的 eBay 商品清单网页中的样子。

图 20.3  
使用HTML和CSS  
来改进 eBay 商品  
清单网页的布局和外  
观



**注意：**通常可以将拍卖的商品的图像放在自己的网站上，然后在 eBay 的 HTML 代码中直接引用这些图像。要记得修改示例代码中的拍卖商品图像的 URL，引用你的 Web 服务器中的图像文件位置——还要将图像文件上传到你的 Web 服务器。也可直接将拍卖图像放在 eBay 上，但在 HTML 代码中引用图像仍需要使用图像的完整 URL。

By the Way

必须承认，这是一个相当普通的使用 HTML 和 CSS 来改善 eBay 拍卖的例子。不用担心，接下来的几节将在这个例子的基础上进行改进，使商品清单对潜在的顾客极具吸引力。

图 20.4 显示了我在 eBay 上拍卖的一对冰上曲棍球滑板。

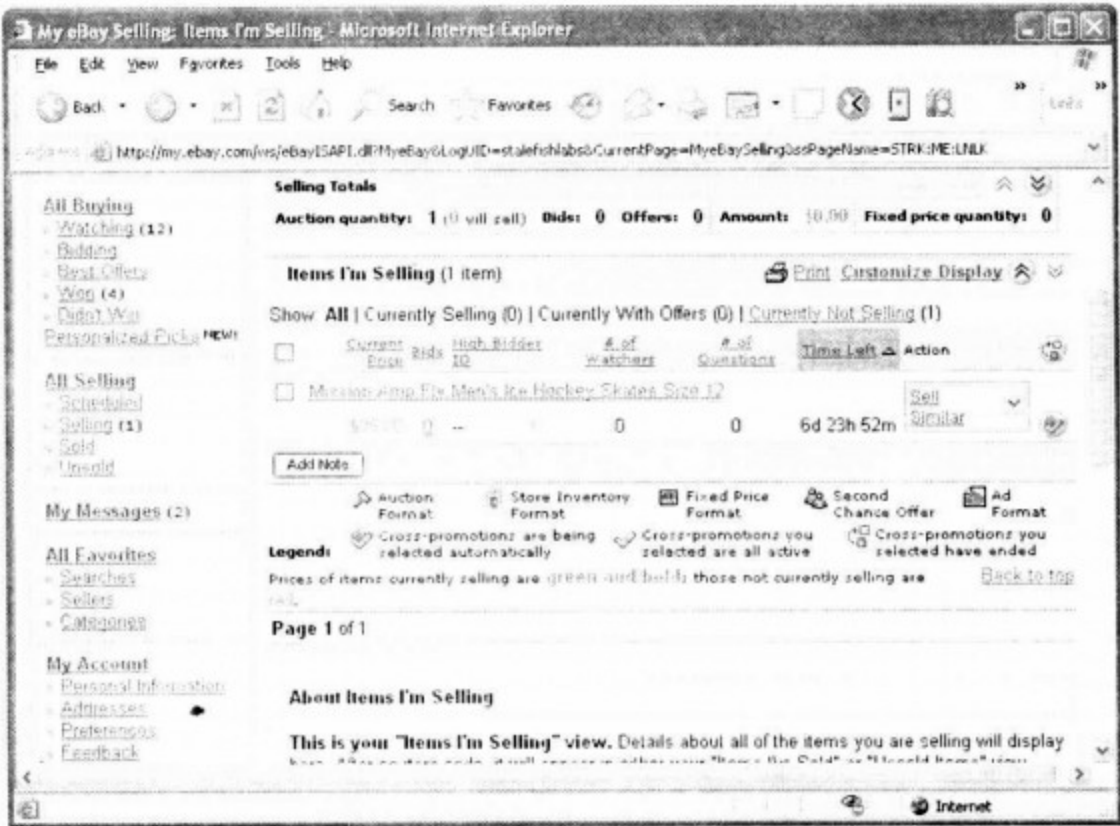


图 20.4  
我要拍卖的商品清单中的曲棍球滑板

图 20.4 表明，插入 eBay 商品清单网页中的 HTML 和 CSS 代码对商品描述外的部分没有任何影响。换句话说，潜在的顾客必须单击商品来查看它的详细清单才能看到你的代码。因此，应使用精确的有吸引力的商品描述，并使用相关的 eBay 清单特性，如陈列图像和粗体标题。

**注意：**你可能想引发竞价风暴，但这是不可能的。2005 年秋天，NASCAR 车手 Robby Gordon 就将他那有名的头盔拍卖。在一场比赛中，Gordon 和 Michael Waltrip 发生了事故，赛后，Gordon 将他的头盔砸向 Michael Waltrip。之后，Gordon 将他的头盔放在 eBay 上拍卖，并准备将拍卖所得献给慈善机构。然而，当竞价达到 1 千万马克后，eBay 停止了这场拍卖。因为看起来像是狂热的 NASCAR 车迷哄抬起来的“虚价”。

By the Way

## 20.4 提供到 About Me 网页的链接

你可能没有注意到，eBay 支持 About Me 特性，让你能够在其中包含关于自己或企业的



信息。About Me 网页与 eBay 用户 ID 相关联，可通过简单的 URL 访问，如 `http://members.eBay.com/aboutme/UserID/`。

在这个 URL 中将你的 eBay 用户 ID 插入 UserID 处，试试这个链接。你将看到自己的 About Me 网页，或者将可以创建一个新的。

我提到 About Me 网页是因为，它为潜在的买家提供了一个在浏览拍卖商品时了解你或公司的途径。你要做的只是使用 HTML 在商品清单中提供一个指向你的 About Me 网页的链接。下面是一个例子：

```
<p>
  To learn more about me, just click <a
    href="http://members.ebay.com/aboutme/stalefishlabs/"></a>.
</p>
```

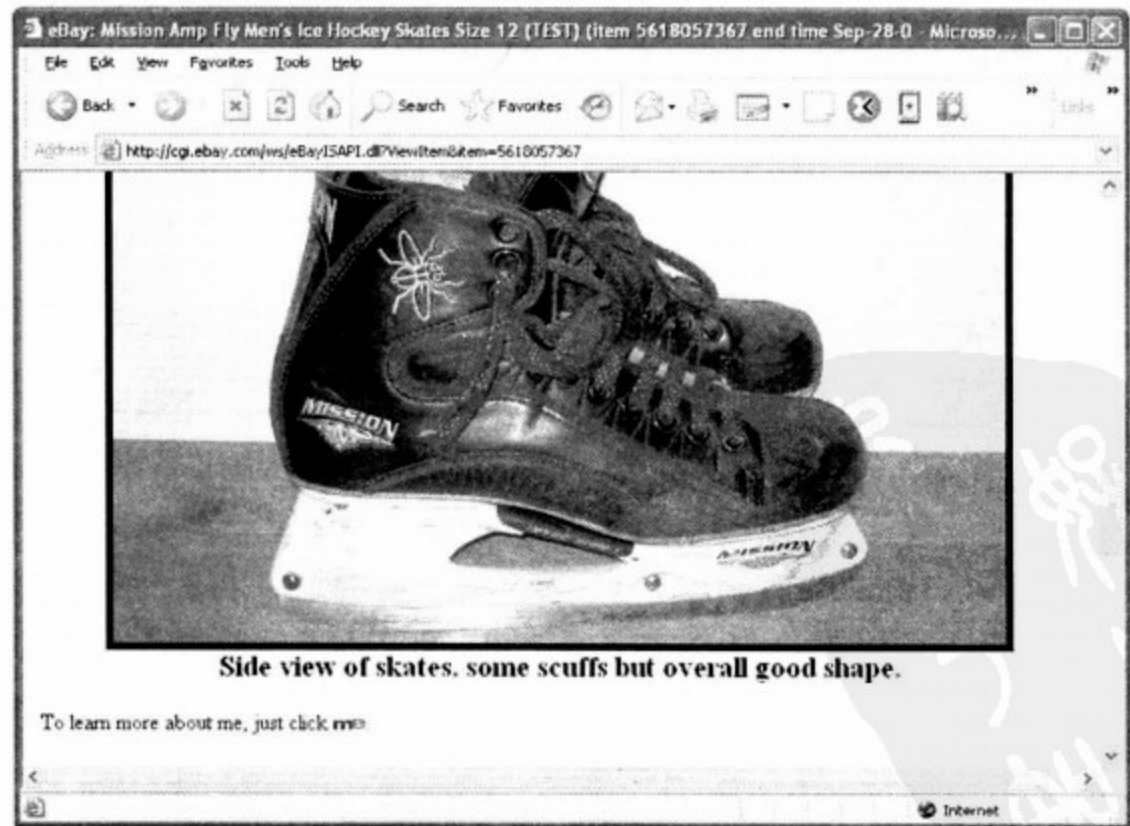
这段代码中只有一个不平常的地方，那就是它引用的图像，这是一个由彩色字母组成的单词 me 的小图像，是 eBay 的商标。这个图像位于 eBay，网址为 `http://pics.ebay.com/aw/pics/aboutme-small.gif`，因此，要使用它，只要在 `<img>` 标签的 `src` 属性中指定这个 URL。

By the Way

注意：一般来说，不提倡使用别人服务器上的图像。然而，在这个例子中，包含代码的拍卖网页正好是在 eBay 的服务器上，因此没有问题。

图 20.5 显示了 eBay 上曲棍球滑板的拍卖网页，商品描述底部包含到 About Me 网页的链接。

图 20.5  
指向 About Me 网页的图形链接使用了标准的 eBay 图像



单击图中的 me，用户就可以看到你的 About Me 网页，通过这个网页用户可以了解更多关于你或公司的信息。

## 20.5 修改商品网页的背景

在本书中的示例（更不用说第 9 章）中，网页背景都是通过样式属性 `background-color` 和 `background-image` 设置的。要将这两个样式属性应用于整个网页，必须在 `<body>` 标签内设置它们。这对于 eBay 拍卖网页不可行，因此必须借助于 JavaScript 方法来取得所需的效果。

我说的 JavaScript 方法指的是动态地设置属性 `document.body.background`，该属性存储了网页的背景。下面是如何用一行 JavaScript 代码来设置这个属性的一个例子：

```
document.body.background = "http://www.yourwebsite.com/background.jpg";
```

注意，在这段代码中，我给出了 `background.jpg` 图像的完整 URL。这是非常重要的，因为拍卖网页将放在 eBay 上，而背景图像可能存储在你自己的 Web 服务器上。因此，要定位图像，必须给出完整的地址。

程序清单 20.3 包含修改后的曲棍球滑板拍卖网页的代码，现在这个网页包含一个被动态修改的背景图像。

程序清单 20.3 这段代码动态地设置 `<body>` 标签中的背景图像

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head><title></title></head>

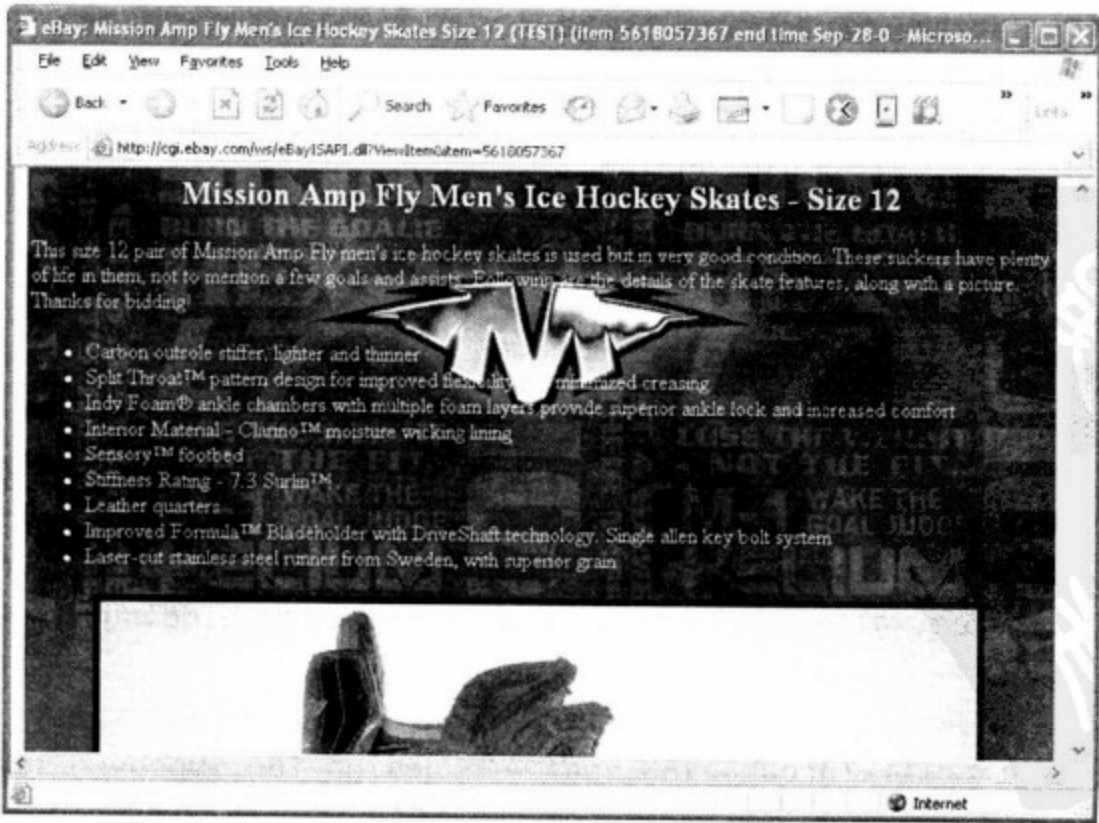
<body>
<!-- EBAY HTML CODE STARTS HERE -->
<script type="text/javascript">
    <!-- Hide the script from old browsers
    // Set the background image
    document.body.background =
        "http://www.michaelmorrison.com/ebay/missionback.jpg";
    // Stop hiding the script -->
</script>

<h2 style="text-align:center; color:white">Mission Amp Fly Men's Ice Hockey
Skates - Size 12</h2>
<div style="color:white">
    This size 12 pair of Mission Amp Fly men's ice hockey skates is used but in
    very good condition. These suckers have plenty of life in them, not to
    mention a few goals and assists. Following are the details of the skate
    features, along with a picture. Thanks for bidding!
    <ul>
        <li>Carbon outsole stiffer, lighter and thinner</li>
        <li>Split Throat&#153; pattern design for improved flexibility and
```

```
minimized creasing</li>
<li>Indy Foam&#174; ankle chambers with multiple foam layers provide
superior ankle lock and increased comfort</li>
<li>Interior Material - Clarino&#153; moisture wicking lining</li>
<li>Sensory&#153; footbed</li>
<li>Stiffness Rating - 7.3 Surlin&#153;</li>
<li>Leather quarters</li>
<li>Improved Formula&#153; Bladeholder with DriveShaft technology. Single
allen key bolt system</li>
<li>Laser-cut stainless steel runner from Sweden, with superior grain</li>
</ul>
</div>
<div style="text-align:center; color:white">
  <br />
  <div style="text-align:center; font-size:16pt; font-weight:bold">Side view of
  skates, some scuffs but overall good shape.</div>
</div>
<p style="color:white">
  To learn more about me, just click <a
  href="http://members.ebay.com/aboutme/stalefishlabs/"></a>.
</p>
<!-- EBAY HTML CODE ENDS HERE -->
</body>
</html>
```

这段代码的关键在于网页主体开头的<script>标签。在这个标签内，document.body.background 属性设置为一个给曲棍球滑板网页增添了很多趣味的图像（见图 20.6）。

图 20.6  
动态设置拍卖商品  
网页的背景可以获得生动的效果



你可能还注意到了，我不得不在这个版本的曲棍球滑板商品网页中增加一些 CSS 样式属性，使文本在深色背景中更突出。我通过 color 样式属性将该页面上的所有文本都设置为白色。

**休息一下：**现在可能放松并探索 eBay，以获得不寻常发现的好时机。下面的网站记录了通过 eBay 市场交易的不寻常商品：<http://www.weirdauctions.com/>。希望你喜欢它！

**Coffee**  
**Break**

## 20.6 创建商品清单幻灯片

要在 eBay 拍卖清单中添加一些突出的特点，可以插入一些 JavaScript 代码，将静态图像变为不断变化的幻灯片。这让你能够包含拍卖商品的多个图像而不会占满整个网页。另外，可以为每幅图像添加标题，其中包含关于商品的描述性文本。

程序清单 20.4 列出了曲棍球滑板 eBay 拍卖网页的最后版本的代码，它利用 JavaScript 在商品描述中包含幻灯片。

程序清单 20.4 这个网页使用 JavaScript 来显示动态的商品图像幻灯片

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head><title></title></head>

<body>
<!-- EBAY HTML CODE STARTS HERE -->
<script type="text/javascript">
    <!-- Hide the script from old browsers
    var slide_wait = 3; // in seconds
    var slide_text = ["Side view of skates, some scuffs but overall good shape.",
                      "Front view, toes are solid and intact.",
                      "Other side view, normal wear and plenty of life left.",
                      "Rear view, practically brand new condition."];
    var cur_slide = 1;
    var timer = 0;
    // Preload the images
    for(var i = 0; i < slide_text.length; i++) {
    var image = new Image();
        image.src = "http://www.michaelmorrison.com/ebay/skates" + i + ".jpg";
    }

    // Start the initial timer
    timer = setTimeout("nextImage();", slide_wait * 1000);
```

```

// Set the background image
document.body.background =
    "http://www.michaelmorrison.com/ebay/missionback.jpg";

function nextImage() {
    // Clear the timer and set a new one
    if (timer != 0)
        clearTimeout(timer);
    timer = setTimeout("nextImage();", slide_wait * 1000);

    // Advance to the next slide image
    if (++cur_slide > slide_text.length)
        cur_slide = 1;
    var img_element = document.getElementById( 'slide' );
    img_element.setAttribute("src",
        "http://www.michaelmorrison.com/ebay/skates" + cur_slide + ".jpg");
    img_element.setAttribute("alt", slide_text[cur_slide - 1]);

    // Update the slide description
    document.getElementById( 'caption' ).innerHTML = slide_text[cur_slide - 1];
}
// Stop hiding the script -->
</script>

<h2 style="text-align:center; color:white">Mission Amp Fly Men' s Ice Hockey
Skates - Size 12</h2>
<div style="color:white">
    This size 12 pair of Mission Amp Fly men' s ice hockey skates is used but in
    very good condition. These suckers have plenty of life in them, not to
    mention a few goals and assists. Following are the details of the skate
    features, along with a few pictures. Thanks for bidding!
    <ul>
        <li>Carbon outsole stiffer, lighter and thinner</li>
        <li>Split Throat&#153; pattern design for improved flexibility and
        minimized creasing</li>
        <li>Indy Foam&#174; ankle chambers with multiple foam layers provide
        superior ankle lock and increased comfort</li>
        <li>Interior Material - Clarino&#153; moisture wicking lining</li>
        <li>Sensory&#153; footbed</li>
        <li>Stiffness Rating - 7.3 Surlin&#153;</li>
        <li>Leather quarters</li>
        <li>Improved Formula&#153; Bladeholder with DriveShaft technology. Single
        allen key bolt system</li>
        <li>Laser-cut stainless steel runner from Sweden, with superior grain</li>
    </ul>
</div>
<div style="text-align:center; color:white">
    <img id="slide" src="http://www.michaelmorrison.com/ebay/skates1.jpg"

```



```

alt="Mission Amp Fly Ice Hockey Skates" style="border:5px solid black"
onclick="nextImage();" /><br />
<div id="caption" style="text-align:center; font-size:16pt;
font-weight:bold">Side view of skates, some scuffs but overall good
shape.</div>
</div>
<p style="color:white">
  To learn more about me, just click <a
  href="http://members.ebay.com/aboutme/stalefishlabs/"></a>.
</p>
<!-- EBAY HTML CODE ENDS HERE -->
</body>
</html>

```

你并不需要完全理解这个网页中的脚本代码，而只需知道幻灯片标题是如何创建的以及如何修改幻灯片本身的名称。下列代码用于创建幻灯片标题：

```

var slide_text = ["Side view of skates, some scuffs but overall good shape.",
                  "Front view, toes are solid and intact.",
                  "Other side view, normal wear and plenty of life left.",
                  "Rear view, practically brand new condition."];

```

这段代码创建一个标题数组，它与幻灯片展示中的每幅图像一一对应。标题数量必须与图像数量相同。

**提示：**要提高或降低幻灯片显示的速度，可修改 JavaScript 代码中的 `slide_wait` 变量。这个变量控制每个图像显示的时间。例如，下面的代码通过将每个图像显示的延迟提高到 5 秒，使幻灯片显示的速度变慢：

**Did you know?**

```
var slide_wait = 5; // in seconds
```

代码中有几处用到了幻灯片图像本身。下面一行代码显示如何根据当前的幻灯片计算每个图像的 URL：

```

img_element.setAttribute("src",
  "http://www.michaelmorrison.com/ebay/skates" + cur_slide + ".jpg");

```

要利用这段代码，必须将 URL 改为自己的图像的地址。例如，如果你的拍卖商品图像为 `putter1.jpg`、`putter2.jpg` 等，并位于 `http://www.mygolfstuff.com/images/`，下列 JavaScript 代码就能正确地设置图像：

```

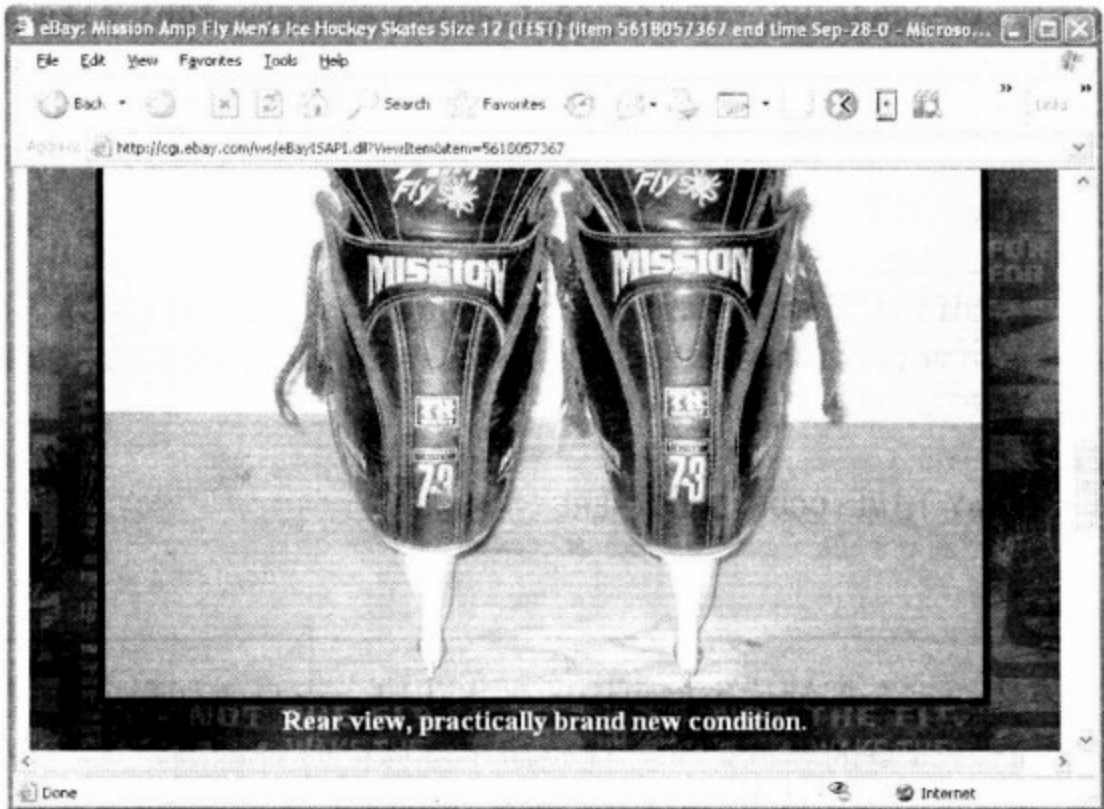
img_element.setAttribute("src",
  "http://www.mygolfstuff.com/images/putter" + cur_slide + ".jpg");

```

代码中有几个地方需要根据你的商品图像进行修改。查找 `skates`，并进行相应的修改。

图 20.7 显示了使用幻灯片的曲棍球滑板 eBay 拍卖网页。

图 20.7  
曲棍球滑板 eBay  
拍卖网页利用严格  
定时的幻灯片显示  
商品的一系列图像



包含循环显示商品图像的动态幻灯片展示的 eBay 拍卖并不多。通过结合使用 JavaScript、HTML 和 CSS，很可能提高在 eBay 上的销售数量！

## 20.7 总结

本章进入一些相当实用的领域，介绍如何使用 HTML 和 CSS 在 eBay 创建样式化的商品拍卖网页。eBay 是非常流行的在线市场，你现在知道了如何应用学到的网页设计技巧来创建有吸引力的 eBay 商品清单。你不仅知道如何利用 HTML 和 CSS 来做一些很酷的事情，还知道如何使用 JavaScript 在商品拍卖网页中加入一些动态特性，如幻灯片。

## 20.8 问与答

问：eBay 提供了一些模板可用于创建有趣的网页。为什么不使用它们？

答：想寻找简单的方法？虽然利用标准 eBay 模板确实可以创建比较美观的网页，但最终将创建出与其他许多商品网页看起来类似的商品网页。如果不利用 HTML 和 CSS 使网页别具特色，又何必辛苦地学习 HTML 和 CSS 呢？也许更重要的是，可能别人更了解你在 eBay 销售的产品的独特之处。使用这些知识，加上新学习的 HTML 和 CSS 技巧，可创造出让人眼前一亮的拍卖网页。

问：创建自己的商品图像非常痛苦。可不可以使用他人列出的类似商品的图像？

答：从技术上说是可以的；但从道德上说恐怕不行。虽然偶然会看到一些商品图像可能来自制造商的网址，但 eBay 上大部分商品照片都是卖主自己拍摄的。不管是哪种情况，这些照片都是所有者的财产，所以是禁止“借用”的。如果得到了许可或决定使用商品图像，一定要将图像复制到自己的 Web 服务器。

## 20.9 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 20.9.1 测验

1. 编写用于 eBay 商品清单的 HTML 与 CSS 时，eBay 商品网页的哪部分是可以修改的？
2. 使用什么代码可以动态地将商品网页的背景设置为图像 `background.gif`？该图像存储在 `http://www.ilovepez.com/pics/`。
3. 在 eBay 上，一架 Grumman Gulstream II 喷气机能拍卖到多少钱？

### 20.9.2 答案

1. 你只能将 HTML 和 CSS 代码放在拍卖商品的描述中，它自动放置在 eBay 商品网页的标签 `<body>` 与 `</body>` 之间；你不能访问 eBay 商品网页的 `head` 部分。
2.  
`"http://www.ilovepez.com/pics/background.gif";`
3. 大约 490 万美元——只是为了确保读者认真阅读！

## 20.10 练习

1. 现在找一件可拍卖的商品，将其放到在 eBay 上。为商品拍些相片，然后修改曲棍球滑板网页的 JavaScript 代码，创建一个带商品图像幻灯片的商品清单网页。然后，坐下来看看滚滚而来的报价！
2. 试试超出本章示例外的技巧，将商品拍卖网页变得更引人入胜。例如，考虑在拍卖网页中嵌入视频来显示商品使用时的情况（假设商品用过）。或者使用表格列出商品的各种属性，甚至将其同竞争产品进行比较。

## 第 21 章

# 创建自己的博客

回忆一下 2004 年美国总统大选的喧嚣，Internet 史无前例地在谁可能成为总统的政治争论上发挥重要作用。这场争论的大部分都是通过博客进行的。博客是指针对某个主题的在线出版物，其内容经常更新。我说的主题是指博客通常都集中于个人或单个组织的观点或比较集中的话题。

本章将探讨被托管的博客，即存储在别人的 Web 服务器上的博客。被托管的博客比自己创建的博客更有优势，因为它们通常都依赖于功能强大的博客软件，自动完成更新和发布过程。不用担心，你在本章中还将使用一些代码，学习如何使用 CSS 定制被托管的博客的外观。

### 21.1 什么是博客

在第 14 章，你学习了如何为基本的博客网页创建样式表模板。虽然这个模板管用，但它并没有体现博客的主要优点：易于发布。在更详细介绍博客的发布前，先来看看什么是博客，什么不是。

首先，博客 (blog) 一词来自 Web log 或 Weblog，指有关某人的想法或行为的在线日志。博客的问世已有多多年，但流行却是在最近一两年的事。要理解的一点是，博客的核心是网页。博客由 HTML 和 CSS 代码组成，可包含可放在普通网页中的任何内容。使网页成为博客的主要因素是，它包含经常更新的内容，并以一贯的、通常是连续的方式进行组织。

虽然博客可以包含图像，但大多数博客主要还是文本内容，通常是关于某个主题的简短文章。有些博客提供评论功能，让访问者能够对博客文章进行评论。博客还为每篇文章提供惟一的 URL，让访问者能够查看较老的博客文章。这样的链接称为 permalink，因为它们是指到博客文章的永久性链接。

**By the  
Way**

**注意：**有些博客称为照片博客，主要包含图像。在照片博客中经常贴图像，如到过的地方的照片，而不是张贴文本内容。有关照片博客的详情，请访问 <http://www.photoblogs.org/>。



图 21.1 是一个专业博客的例子，它几乎具备高水平博客所有的特色。

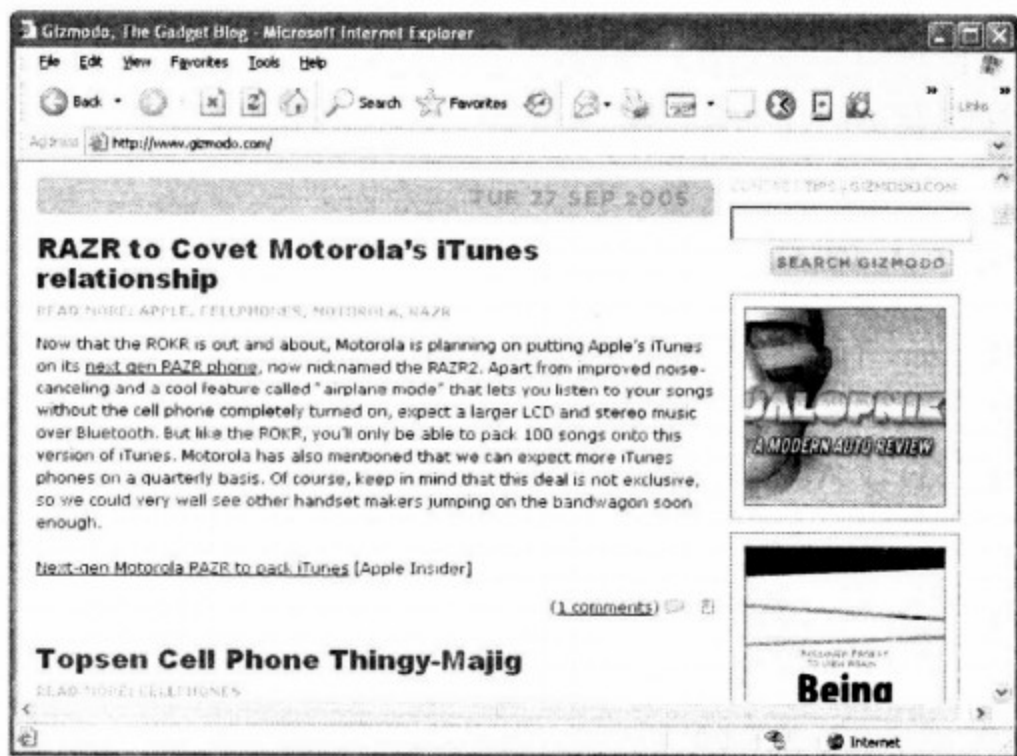


图 21.1  
Gizmodo 博客提供定期更新的有关电子产品的信息

上图中的博客是 Gizmodo，是我最喜欢的博客之一。它自称为“电子产品博客”(The Gaget blog)，是功能完善的博客典范。如果仔细查看图 21.1，将发现博客文章的右下角有个评论链接，旁边还有一个小型 **permalink** 图标。该博客还有突出的标题、格式清晰的博客文本及发表日期。

一种推动博客成为主流的有趣技术是 RSS (Really Simple Syndication)，它让对博客感兴趣的人可以定期收到新博客发表通知。RSS 让你能够使用新闻聚合软件 (news feed syndicate)，如 Feed Demon (<http://www.feedException.com/>) 同时跟踪许多博客的新内容。如果不想安装专用软件，也可以使用可在网页浏览器中使用的在线 RSS 阅读器，如 Bloglines (<http://www.bloglines.com/>)。可利用 RSS 将喜欢的博客标签保存下来，这种高效性推动了博客的流行。第 24 章将更详细地介绍 RSS，并介绍如何浏览支持 RSS 的聚合网站 (如博客)。

回到博客的管理，博客之所以流行是因为易于发布。无论按什么样的标准，第 14 章开发的博客网页都是可以接受的。然而，它的局限在于如何更新。对于需要定期 (如每天或每周几次) 更新的网站来说，手工编辑 HTML 代码，然后通过 FTP 发布网页比较费劲。可视化 Web 开发工具，如 FrontPage 和 Dreamweaver，可以使更新容易些，但仍需要手工编辑 HTML 代码。虽然本书介绍了 HTML 代码编辑，但有时使用自动化工具会更好。

有许多博客服务和软件包，让你能够通过 Web 界面编辑和更新博客，从而简化了博客的发布过程。换句话说，只要登录网站，然后输入博客内容，编辑完后就可以直接将其发布到博客中，而不需要 FTP，不需要专门的 Web 开发工具，在很多时候也不需要 HTML 代码。我为什么在关于 HTML 的书中推荐让你不需要编写 HTML 代码的工具呢？因为这样的工具可以减轻你的负担。

**注意：**我自己在博客及其发布简单性问题上有所体会。我的个人网站原来就是一个原始博客，我必须手工编写 HTML 代码，然后通过 FTP 发布文件。使用使发布过程自动化的博客软件后，我更乐于更新内容。不仅如此，由于博客是基于 Web 界面的，我还可以在任何地方增加和编辑博客文章，而不需要特殊软件。

By the  
Way



本章并不完全抛弃 HTML 和 CSS，而全部介绍博客服务或专门的博客软件。要定制博客的外观，使其与网站的其他部分风格相同，需要对 HTML 和 CSS 有一定的理解。使用这种方法，可以在已有的网页中链接到其他地方的博客，同时确保所有网页的风格一致。这可能是一种最好的方法，因为你能够全面控制你的网页，同时享受博客服务的自动化优点。

## 21.2 在 Blogger.com 上创建博客

写博客的人通常称为 blogger，有一个流行的博客服务也叫 Blogger: <http://www.blogger.com/>。本章将重点介绍该博客服务，教你学会如何创建自己的博客、定制博客并将在已有网页中链接到博客。我与 Blogger 并没有什么关系，只是发现它是最好的博客服务之一。你也可能考虑下列博客服务，它们都是免费的，并提供与 Blogger 类似的功能。

- TypePad: <http://www.typepad.com/>。
- Bloglines: <http://www.bloglines.com/>。
- LiveJournal: <http://www.livejournal.com/>。
- MySpace: <http://www.myspace.com/>。
- MSN Spaces: <http://spaces.msn.com/>。

### By the Way

**注意：**Blogger 是第一个为公众提供免费博客托管的主流博客服务之一。Blogger 服务于 2003 年初被 Google 收购，它现在还是 Google 内部的一个小组，主要致力于个人信息的组织。

回到博客。可以使用 Blogger 博客服务来创建完全由 Blogger 托管的免费博客。在 Blogger 上使用博客，不需要创建或发布任何网页。使用 Blogger 的第一步是访问 <http://www.blogger.com/signup.g>（在这个 URL 的最后一定要包含.g）创建一个账号。按照网上的指示创建免费账号。图 21.2 显示了如何在 Blogger 上输入基本的账号信息。

图 21.2

创建博客账号，主要是要指定用户名和密码并提供电子邮件地址

单击 Continue 继续创建博客的过程。图 21.3 显示了如何输入新博客的标题以及用于博客 URL 的名称——这是在网页浏览器的地址栏中输入的、用于访问该博客的地址。

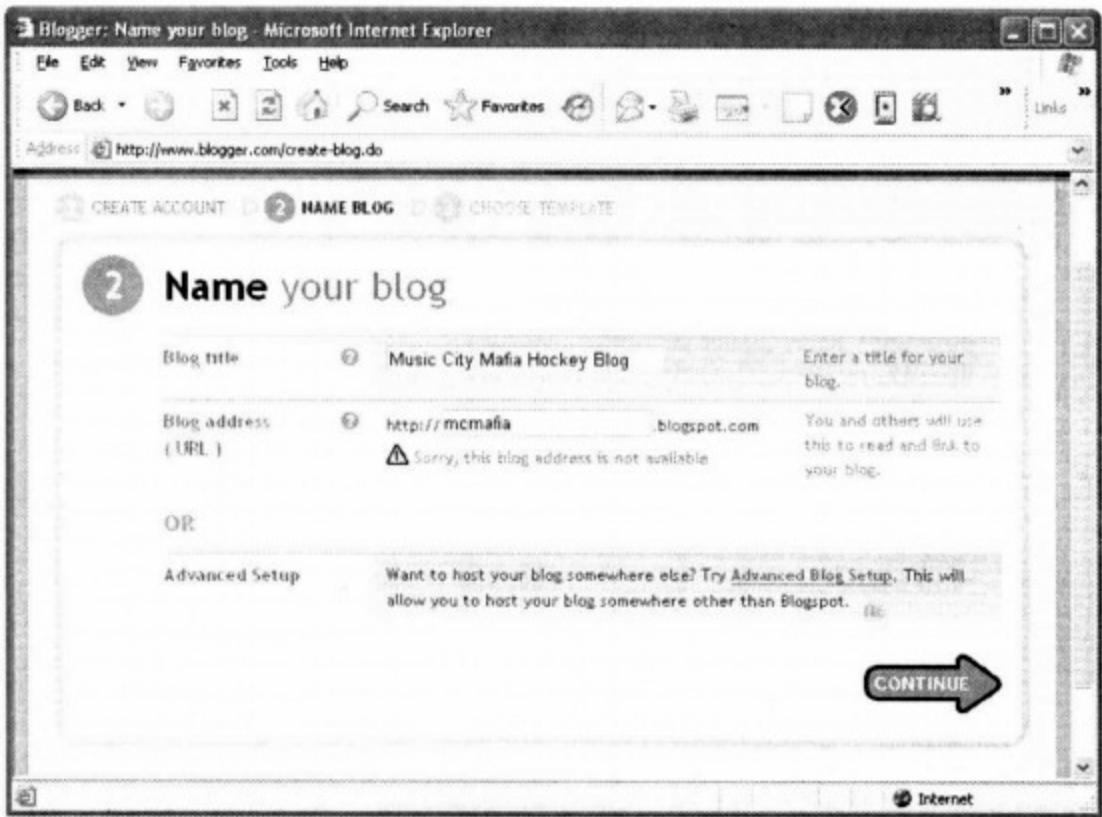


图 21.3  
指定博客标题是  
创建博客的重要  
一步

**注意：**如果发现博客名已经有人用，只要修改名称，直到找到 Blogger 接受的名称。最后，在 URL 中使用的名称不应太难记，也不应太长。

**By the  
Way**

单击 Continue 进入博客创建的下一步。图 21.4 显示了博客模板的选择，开始要求使用模板，本章稍后可以对模板代码进行修改。

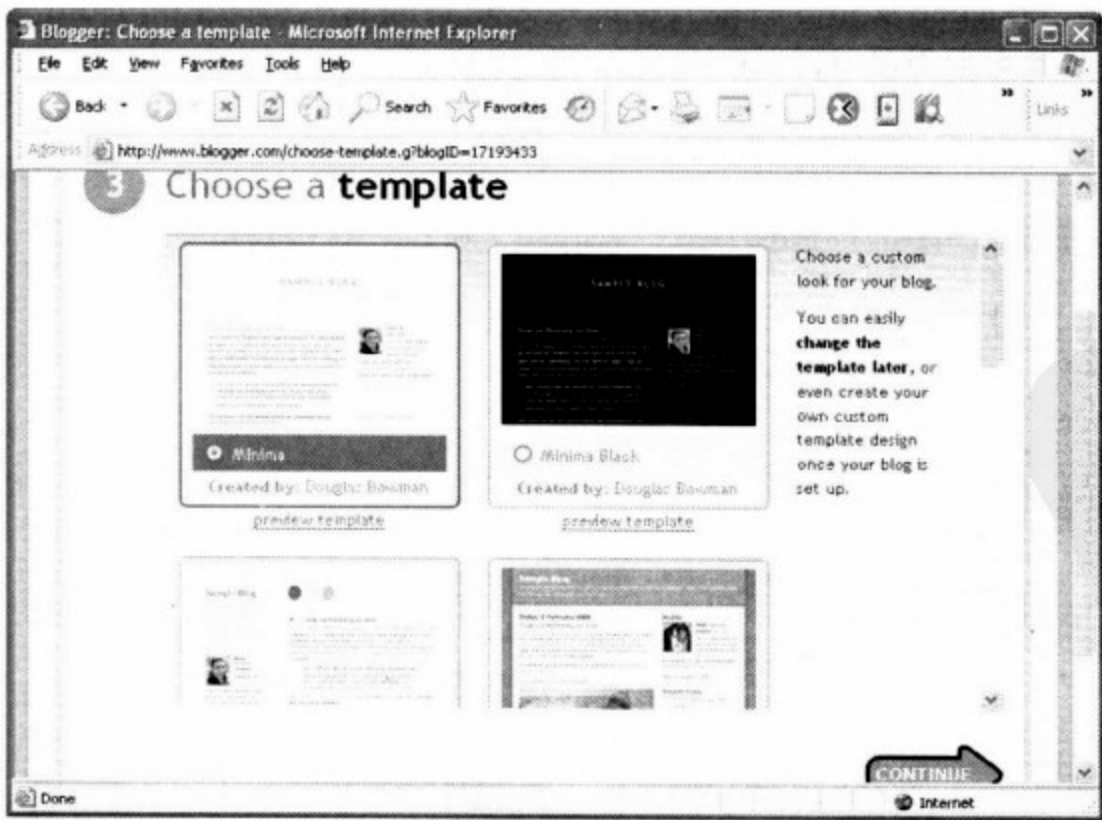


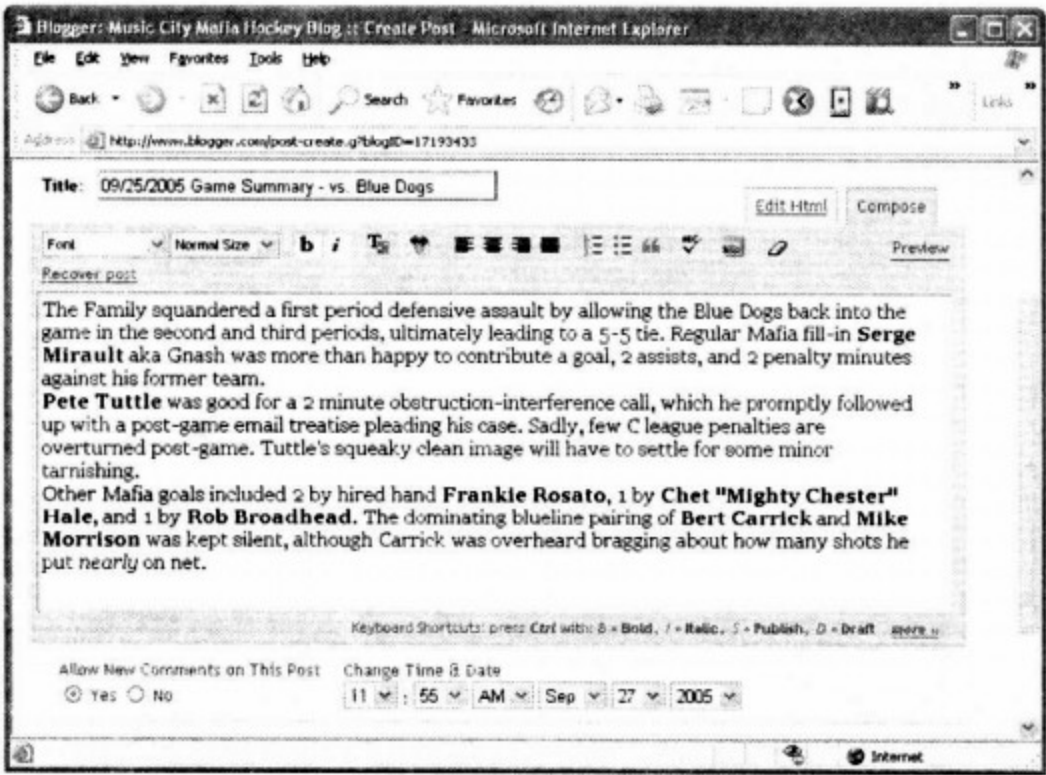
图 21.4  
开始必须选择一  
个博客模板，本  
章稍后可以自定义  
CSS 模板

单击 Continue 接受模板，并结束博客创建过程。你将看到确认屏幕，其中有一个 Start Posting 链接，单击它可以开发发布第一篇博客文章。

## 21.3 创建第一篇博客文章

现在到了通过博客将想法与全世界分享的时候。图 21.5 显示了 Blogger 的博客文章输入网页，它包含一个大型编辑框，在其中可以输入新博客文章的内容。

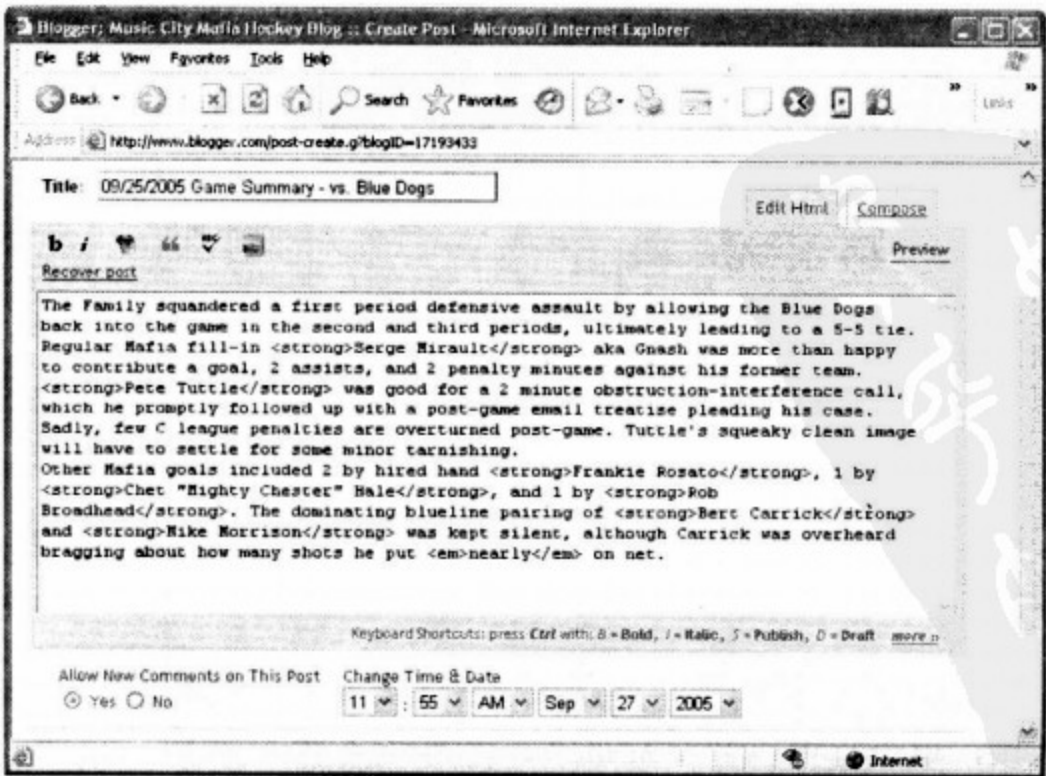
图 21.5  
Blogger 的发布区域包含一个编辑框，在其中可以输入博客文章的内容



从图中可以看到，Blogger 发布区域中的编辑框上方包括几个格式化控件，让你能够进行一些常用的格式化，如更改文本的字体和对齐方式。你也可以插入图像，甚至对网页内容进行拼写检查。

从 HTML 的角度看，Blogger 发布区域最有趣的地方是页面右上角的 Edit Html 选项卡。单击该选项卡，可以看到博客文章的 HTML 代码。图 21.6 显示了图 21.5 所示博客文章的 HTML 代码。

图 21.6  
Blogger 发布区域的 Edit Html 选项卡让你能够直接使用 HTML 代码编辑博客文章





上图表明，可以使用<strong>和<em>标签将文本指定为粗体和斜体。

**注意：**<strong>和<em>标签可将文本格式化为粗体和斜体，但不能保证文本真正显示为粗体和斜体。这是因为<strong>和<em>标签旨在强调文本，而不是指定文本应如何显示。因此，如果有视力障碍的人通过特殊的声音浏览器听网页，他可能听到<strong>和<em>强调的文本有不同的音调。<strong>和<em>标签内的文本之所以会是粗体和斜体，是因为普通网页浏览器都将这两个标签的文本显示为粗体和斜体。要保证博客中的文本有特定的外观，应使用 CSS。

By the Way

这段博客的 HTML 代码没有说明如何在博客中使用 CSS 样式。当然可以直接在 HTML 代码中输入内联样式，但这没有必要，因为博客已通过模板使用了 CSS 样式表。只要修改模板，就可以使博客的风格与整个网站风格一致。

修改博客模板前，先单击 Publish Post，完成第一篇博客文章的发布。

## 21.4 使用 CSS 修饰博客

要查看新发布的博客文章，只要单击 Blogger 中的 View Blog 或导航到你的博客地址，即 <http://yourblogname.blogspot.com/>。在这里，博客的地址是 <http://mcmafia.blogspot.com>，图 21.7 显示了该网页。



图 21.7  
Music City Mafia 博客显示了刚创建的博客文章

**注意：**在 Blogger 上，博客的 URL 是相对于 [blogspot.com](http://blogspot.com) 的，而不是 [blogger.com](http://blogger.com)。这个重要的细节要注意。

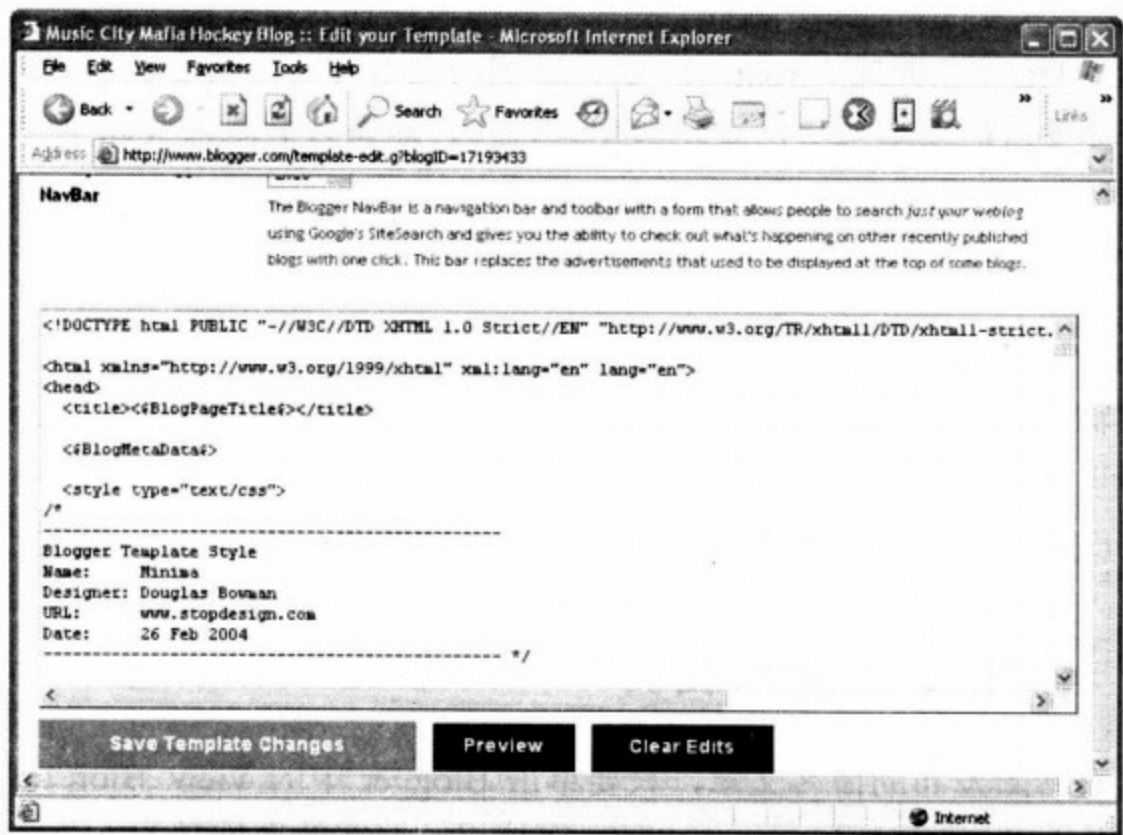
By the Way

从图中还可以看到，博客使用了前面创建博客时选择的默认模板。令人惊讶的是，这个默认模板与你在本书中多次见到的曲棍球队员网页不相称。即使如此，这个模板还是很有帮

助的，可以将这个模板调整成想要的样子。

单击 Blogger 页面上的 Template 选项卡，开始修改博客的 CSS 样式表。图 21.8 是在 Blogger 的模板区域中的 Music City Mafia 博客模板。

图 21.8  
通过 Blogger 的模板区域，可以通过 HTML 和 CSS 代码修改博客的布局和格式化



仔细看 Blogger 模板区域所示的代码，将注意到它不仅包含 CSS 代码；它实际上是一个遵循 XHTML 的完整网页，包含一些 Blogger 专用的用于插入博客内容的标签。更重要的是，它包含一个内部样式表，用作博客模板。可以修改这个样式表来改变博客的外观。

例如，下列代码包含用于博客的 body 元素的 CSS 样式规则：

```
body {
    background:#fff;
    margin:0;
    padding:40px 20px;
    font:x-small Georgia,Serif;
    text-align:center;
    color:#333;
}
```

对博客样式表的第一个修改是，修改正文的字体，使其同曲棍球队员样式表相同，后者使用 Verdana 字体。如果没有这种字体，样式表将退而求其次，使用 Arial 字体。下面是修改字体后的 body 样式规则：

```
body {
    background:#fff;
    margin:0;
    padding:40px 20px;
    font:x-small Verdana, Arial;
    text-align:center;
}
```



```
color:#333;
}
```

前面讲过的，曲棍球队员样式表也使用了特殊的链接格式。下面是博客模板默认使用的链接格式化代码：

```
a:link {
  color:#58a;
  text-decoration:none;
}
a:visited {
  color:#969;
  text-decoration:none;
}
a:hover {
  color:#c60;
  text-decoration:underline;
}
```

下面是修改后的链接格样式，这些代码来自曲棍球队员样式表：

```
a:link, a:visited, a:active {
  color:#19619A;
  font-weight:bold;
  text-decoration:none;
}
a:hover {
  background-color:gold;
  font-weight:bold;
  text-decoration:none;
}
```

新的链接样式使链接不再显示下划线。另外，链接文本显示了浅蓝色阴影，且当鼠标指向链接时链接将显示金色背景。

为使博客锦上添花，可以修改博客的标题，将文本颜色改为海蓝色，背景为金色。下面是博客 head 部分的原始 CSS 代码：

```
#header {
  width:660px;
  margin:0 auto 10px;
  border:1px solid #ccc;
}
```

下面是修改后有金色背景的 header 样式：

```
#header {
  background-color:gold;
  width:660px;
```

```
margin:0 auto 10px;
border:1px solid #ccc;
}
```

博客标题的海蓝色是用另一个样式规则#blog-title 指定的，如下所示：

```
#blog-title {
margin:5px 5px 0;
padding:20px 20px .25em;
border:1px solid #eee;
border-width:1px 1px 0;
font-size:200%;
line-height:1.2em;
font-weight:normal;
color:#666;
text-transform:uppercase;
letter-spacing:.2em;
}
```

这段代码是默认的博客标题样式规则。下面是修改后的版本，使标题文本颜色为海蓝色，且为小型大写字母，这与曲棍球队员网页样式表中的样式一致：

```
#blog-title {
margin:5px 5px 0;
padding:20px 20px .25em;
border:1px solid #eee;
border-width:1px 1px 0;
font-size:200%;
line-height:1.2em;
font-weight:normal;
color:navy;
font-variant:small-caps;
letter-spacing:.2em;
}
```

样式代码中的最后一部分是每篇博客文章的题目，这使用海蓝色、小型大写字母也更好。下面是这个元素原来的样式规则：

```
.post-title {
margin:.25em 0 0;
padding:0 0 4px;
font-size:140%;
font-weight:normal;
line-height:1.4em;
color:#c60;
}
```

这条样式规则修改后的版本只是在规则的最后添加了属性 color 和 font-variant:

```
.post-title {
margin:.25em 0 0;
padding:0 0 4px;
font-size:130%;
font-weight:normal;
line-height:1.4em;
color:navy;
font-variant:small-caps;
}
```

图 21.9 显示了应用新样式后的曲棍球博客。



图 21.9  
应用修改后的模板的曲棍球博客，与曲棍球队员网页更匹配

如果将这个页面与图 21.7 中的页面比较，将发现修改带来显著的影响。更重要的是，现在这个博客网页与要链接它的曲棍球队员网页在样式上保持一致。图 21.10 显示了这个网页。

## 21.5 在网站中链接到博客

现在，你可能已经有一些网页以及由 Blogger.com 或其他类似的博客服务托管的博客。现在的问题是，如何将这两个在线部分链接在一起。你可能想到了，只要在网页中添加链接，链接到博客即可。

下面的例子演示了如何在曲棍球队员 Terry Lancaste 的网页中，使用 <a> 标签链接到 Blogger.com 的曲棍球博客：

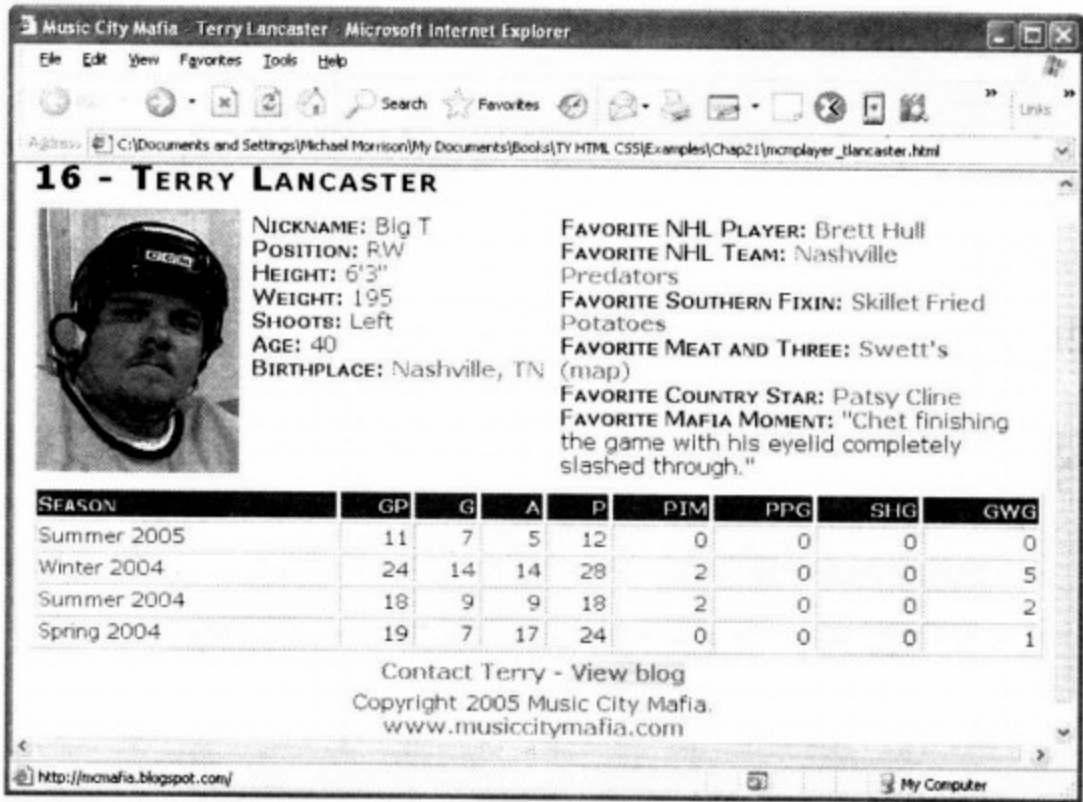
```
<a href="http://mcmafia.blogspot.com/" rel="external">View blog</a>
```

注意到在 href 属性中指定的 URL，与首次在 Blogger 上创建博客时指定的 URL 相同。另外，前面提到过，<a> 标签的 rel 属性用于避开当前浏览器的限制，以便能够在新浏览器窗口中打开链接，同时确保网页是有效的 XHTML 网页。这种技巧在第 3 章讨论如何在新浏览

器窗口中打开链接时介绍过。

图 21.10 是改进后的新的曲棍球队员网页，底部包含了被加亮显示的博客链接。

图 21.10  
博客链接现在位于  
在曲棍球队员网页  
的底部

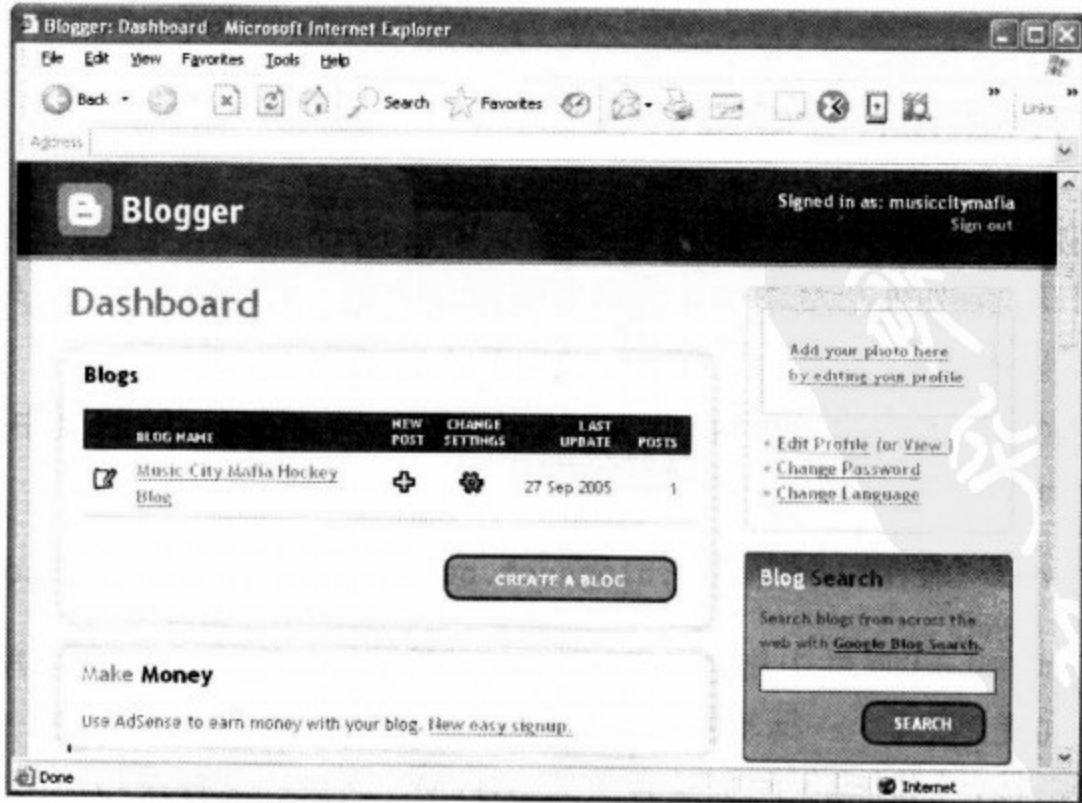


单击 View Blog 链接，将在新浏览器窗口中打开 Music City Mafia 博客，该博客如图 21.9 所示。

## 21.6 更新博客

要更新博客和添加新文章，必须返回到 Blogger 主页 <http://www.blogger.com/>。如果没有自动登录，使用前面创建的用户名和密码登录。登录后将进入 Blogger 的仪表板 (dashboard)，从中可以看到已创建的所有博客。图 21.11 显示了 Music City Mafia 博客的仪表板。

图 21.11  
通过 Blogger 的仪  
表板可以访问你创  
建的所有博客



**休息一下：**在 HTML Dog (<http://www.htmldog.com/>) 可以找到非常实用的博客示例，这对你很有帮助。该博客主要介绍 HTML 和 CSS，是优秀的网页开发在线资源。

**Coffee  
Break**

在上图中，除博客名称外，还有一个加号和一个齿轮图标。单击加号可创建新的博客文章，单击齿轮图标可修改博客的设置。单击博客名称将打开一个列出该博客下所有博客文章的网页，从中可以查看、编辑、删除或管理这些文章。

## 21.7 总结

本章暂停沉闷的 HTML 和 CSS 代码学习，介绍被托管博客的创建过程。与自己使用 HTML 和 CSS 代码手工创建的博客不同，托管的博客在发布和自动化方面提供了无可争辩的优势。因此，我介绍了使用博客服务或特殊的博客软件，而不是从头创建和管理博客。本章介绍了易用而有趣的免费博客服务 Blogger；当然还有其他博客服务，我也提及了一些。探索各种可能的选择，然后投入自己的博客中。进入博客空间后，便获得了将观点与全世界分享的机会，将其变为现实吧！

## 21.8 问与答

**问：**我听说有人通过博客赚了很多钱？如何才能赚钱呢？

**答：**确实有一些人通过博客赚了不少钱，这要感谢他们的博客文章旁边显示的广告。你可以在自己的博客中加入广告，每当有人访问你的博客并单击广告时，你就可以赚钱。Blogger 博客服务包含对 Google AdSense 的内置支持，这是一个通过在网页中放置广告来赚钱的伟大广告项目。第 17 章介绍了 Google AdSense 的基础知识。也可以通过单击 AdSense 上方的 Template 区域，在 Blogger 网站中访问和激活的 AdSense。

**问：**博客应多久更新一次？

**答：**内容越新，人们访问你的博客的可能性越大。另外，如果希望通过广告收入来使博客多赚钱，就要尽力吸引人们回来看你的博客。因此，建议一周至少更新博客文章两三次。如果不会过多地挤占空闲时间，最好每天更新一次。其实一些全职博客写手一天更新博客很多次，虽然从某种程度上说，在这么短的间隔内很难贴出高质量的内容。

## 21.9 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 21.9.1 测验

1. 博客的 permalink 有什么作用？



2. 在 Blogger 服务中, CSS 如何应用于博客?
3. 如何链接到 Blogger 上的一个名为 saltyblog 的博客, 并通过文本 “Here’s the Salty Blog!” 来链接?

### 21.9.2 答案

1. Permalink 是指向博客文章的永久性链接。这很重要, 因为在博客文章不再出现在博客主页中后, 它让用户能够通过 URL 引用特定的博客文章。

2. CSS 可应用于 Blogger, 是因为它让你能够自定义博客的模板, 让博客使用你自己定义的字体、颜色和网页布局。可以通过 CSS 甚至 HTML 来定制所有与 Blogger 模板相关的方面, 如果你有足够自信去学习特定的 Blogger 标签是如何工作的话。

3.

```
<a href="http://saltyblog.blogspot.com/" rel="external">Here's  
the Salty Blog!</a>
```

### 21.10 练习

1. 如果还没有创建自己的博客, 现在一定要创建一个。相信我, 你将发现当你定期将自己的想法放到在线网页上时, 有许多乐趣。如果你有业余爱好、运动或其他兴趣, 可以将博客上重点介绍这方面的内容。

2. 鼓励你的朋友、家人和同事访问并参与你的博客。虽然博客通常代表组织或个人的观点, 但在很大程度上说, 它也可以成为在线社区。你可能对博客上的评论感到惊讶, 这很可能是惊喜!

3. 尝试通过新闻聚合软件, 如 Feed Demon (<http://www.feeddemon.com>), 来预订博客新闻提要 (feed)。你的博客新闻提要要与博客的 URL 相同, 只是后面加上了 atom.xml。例如, 本章的 Music City Mafia 博客可以通过 <http://mcmafia.blogspot.com/atom.xml> 进行聚合。

# 第五部分

## 建立网站

第 22 章 组织和管理网站

第 23 章 帮助人们找到你的网页

第 24 章 超越传统网站





## 第 22 章

# 组织和管理网站

本书前面 21 章介绍了如何设计和创建网页以及要放在网页中的图像。现在要有更宏大的视角，不应再只考虑单个网页，而应整体考虑整个网站。

本章介绍如何组织和展示多个网页，让用户能够在这些网页中导航、浏览，而不致于造成困惑。读者还将学习如何使网站令人难忘，使访问者愿意再次访问。Web 开发人员使用“黏着”来形容人们不愿离开的网页。这一章将帮助你使网页更有黏着力！

因为网站可以是（而且通常也应该是）经常更新的，因此，创建易于维护的网页很重要。本章将教你如何在网页中增加注释和其他文档，以便你或其他同事能够理解和修改网页。

### 练习

现在，你应已经有足够的 HTML 知识，能够创建网站的绝大部分。你可能有许多网页，甚至已经将它们发布到网上。

阅读本章课程时，考虑一下你的网页是如何组织的以及如何改进。如果决定重新设计并修改几乎所有网页，不要为此感到惊讶，这种努力很可能是值得的！

**提示：**如果使用简单的文本编辑器（如 Windows 记事本或 Macintosh SimpleText）编辑 HTML 网页，这时就可以考虑使用交互式网站开发工具，如 Microsoft FrontPage 或 Macromedia Dreamweaver。

除帮助你快速并直观地编写 HTML 和 CSS 外，这些软件还能一次修改并跟踪许多网页，从而节省时间。

另一方面，这些软件并不是免费的，管理中小型网站也并不一定要使用这样的软件。除非网站至少有 50 个网页，或者计划购买包含 FrontPage 的 Microsoft Office 版本，否则，使用简单的文本编辑器或 HTML 编辑器共享软件，加上你已经知道如何使用的文件传输软件，可以节省许多时间和成本。

**Did you know?**

## 22.1 一个网页就足够时

创建和组织具吸引力的高效网站并不总是一项复杂的工作。在有些情况下，可以在一个网页中展示许多有用的信息，而不需要许多花哨的图像。实际上，单页网站有几个优点：

- 网站上的所有信息都可以尽快地下载；
- 只需一个打印命令就可以将整个网站打印到纸上，即使网页打印出来需要几张纸；
- 访问者很容易将网站保存在其硬盘上以便将来引用，特别是当网站使用很少的图像时；
- 同一网页上不同部分的链接的响应速度通常比不同网页之间的链接快。

图 22.1 显示了网页的第一部分，这作为一个长的网页比使用多页网站更好。它包含 8 页纸的文本，解释如何参与流行的电子邮件讨论列表。

图 22.1  
良好的目录可使长  
网页更易于导航



与大多数介绍性网页一样，这个网页开始简单地介绍了这个页面的主要内容和面向的读者。详细的目录让访问者能够直接跳到最感兴趣的内容（关于如何创建目录，请参见第 6 章）。

如图 22.2 所示，页面中每小节后面都有一个链接指向目录，因此，在这个网页中导航感觉就像在多个网页之间导航一样。因为该网页旨在提供方便的参考，访问者肯定更愿意将单个网页作为书签或保存下来，而不愿对 8 或 10 个网页进行操作。

**Did you know?**

**提示：**在本书见过各种花哨的图像和布局技巧后，你可能忘记了这样一点：通常良好的老式大纲是组织网站的最清晰也最有效的方法。即使网站确实需要使用多个网页，如图 22.1 所示，类似于目录的列表可能是引导人们浏览较小网站（或较大网站的一部分）的最好方法。



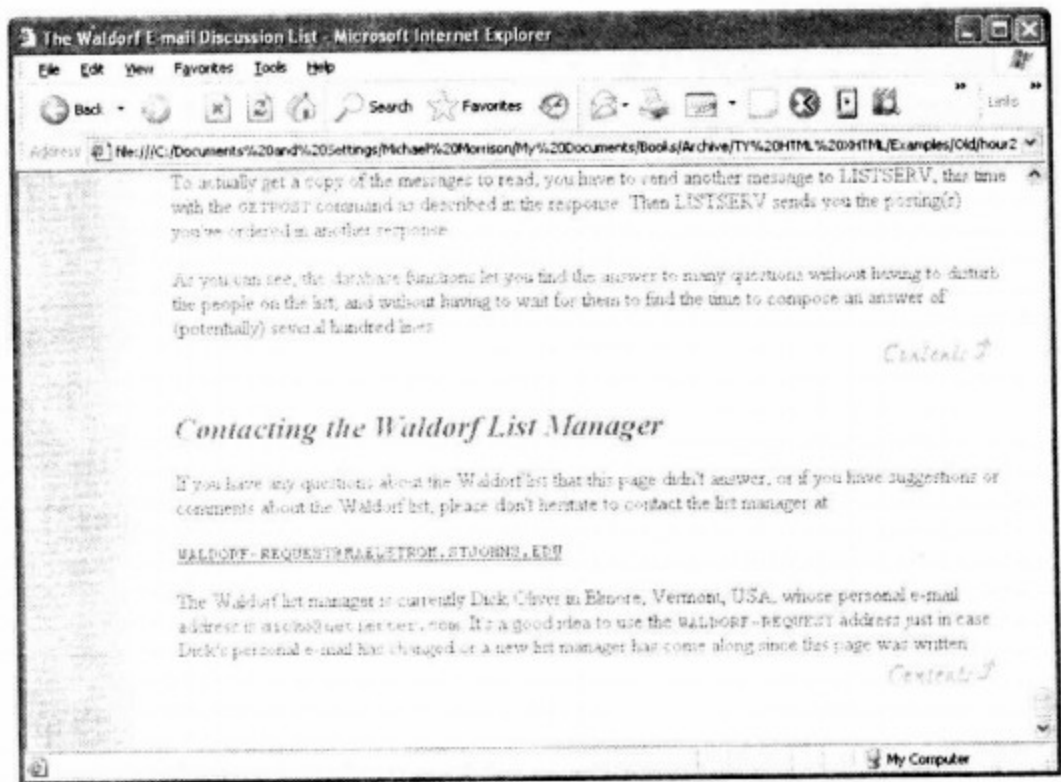


图 22.2  
在长网页中，应在  
每小节后提供指  
向目录的链接

## 22.2 组织简单的网站

虽然有些网站只有一个网页，但大部分公司和个人将网站分为能够快速阅读的多个短小网页，使用图形图标在网页间导航。这样，只需特定信息的用户不必下载整个网站。另外，还减少了需要在网页中滚动的次数，这对于使用手机设备来浏览网站或显示器分辨率比较低（小于 800×600）的用户很方便。

与现在很多网站一样，图 22.3 所示的主页旨在让人们能够通过 Internet 了解该组织。现在，许多人想了解某个组织或寻找是否有特定类型的信息时，立即求助于万维网。简单的主页应陈述足够的信息，使用户能够判断出他是否能够在这里找到更多的信息。还应提供传统的地址和电话联系信息以及电子邮件地址，这可以直接放在主页上，还可以通过突出的链接（如图 22.3 所示的网页右上角的 About Us 按钮）。

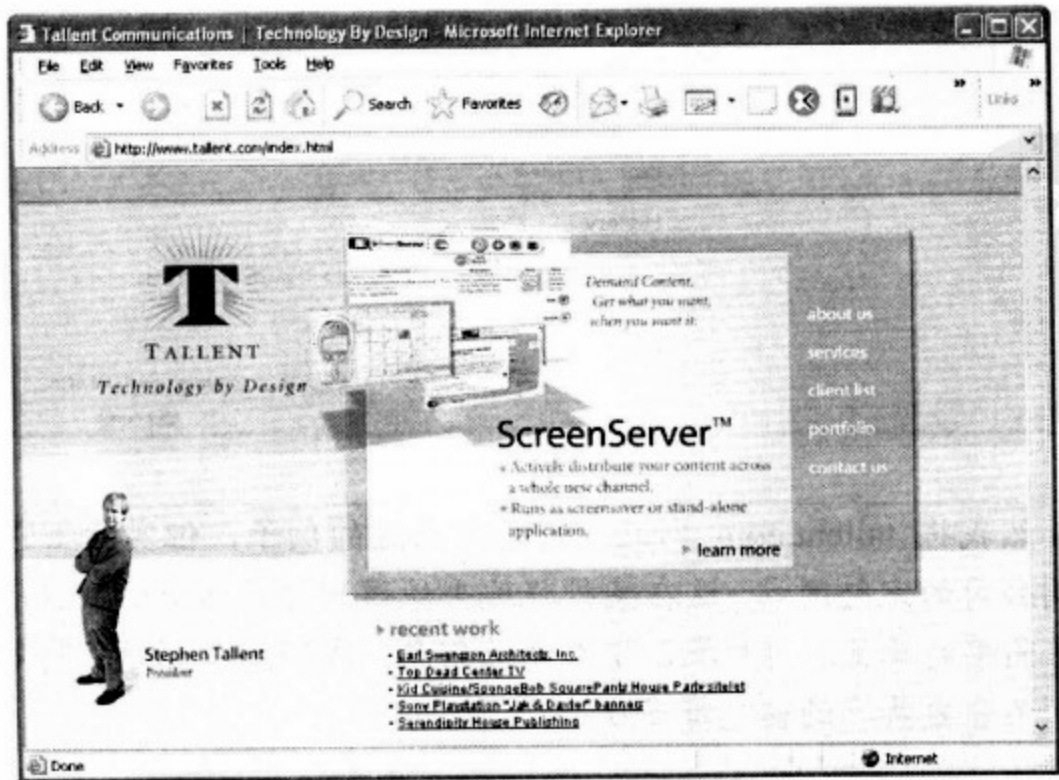


图 22.3  
这个小型企业主  
页使用突出的图  
像和有意义的文  
本，其目的和风格  
一目了然

网站开发新手最常见的错误之一是，网站上每个页面都与前面的页面看起来不一样。另一个同样严重的错误是使用其他成千上万网页制作者也在使用的公共剪贴画。别忘了，在 Internet 上，单击一下就可以将你带到全世界。要使网站突出并给人印象深刻，惟一的方法是使所有网页都遵循独特的明确无误的视觉主题。换句话说，要尽力使自己的网站与其他网站相比具有独特性，但整个网站内部是统一的。

**Did you know?**

**提示：**不管网站有多大，都应仔细组织资源。例如，我总是将网页的图像放在独立的文件夹 images 中。同样，如果有一些文件可供下载，可将其放在文件夹 download 下。这样更易于根据网页资源的类型（HTML 页面、GIF 图像等）对其进行跟踪。以图像为例，可如下引用文件夹 images 下的图像：

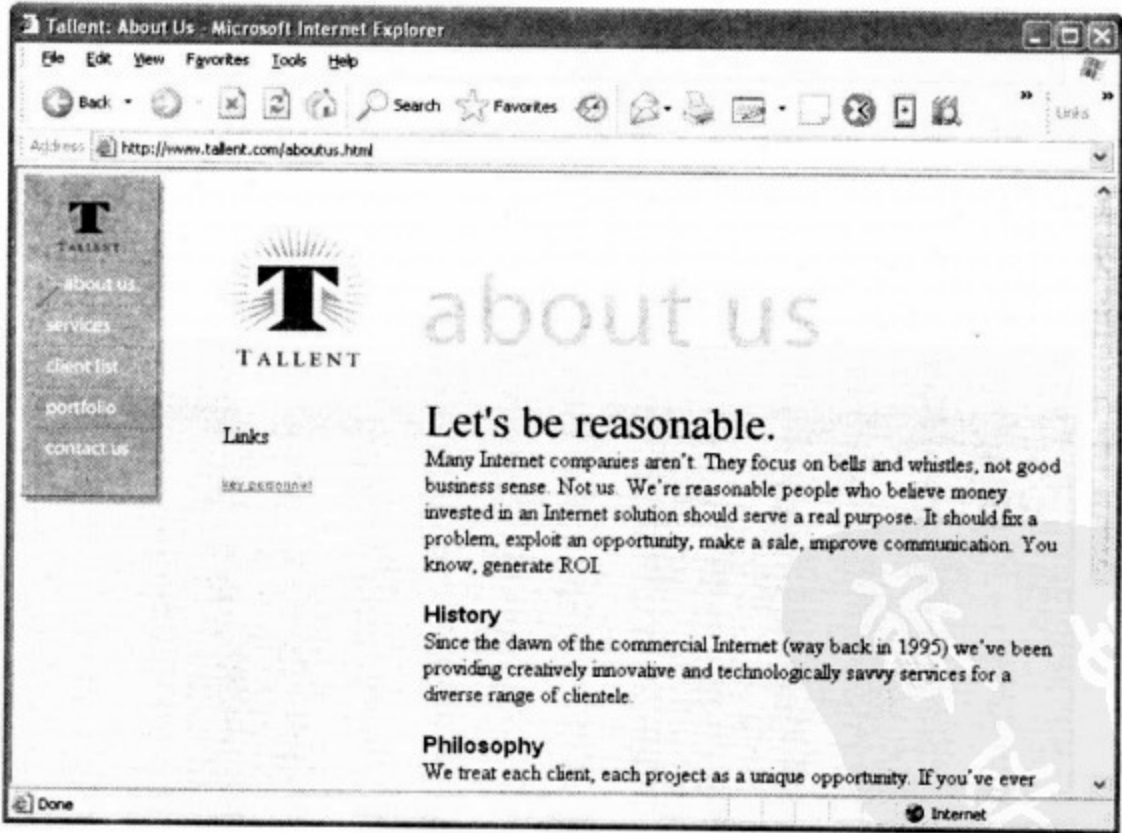
```

```

下面举个例子来说明为什么统一性可使网站具有更高的内聚性。看看当用户单击图 22.3 中的 About Us 时将发生什么。用户将进入如图 22.4 所示的网页，这个网页与前一个网页的风格非常类似。标题链接、背景、徽标和链接图像在视觉上的重复，使用户很容易认出这是同一个网站的另一个网页（复用主页中尽可能多的图像还能提高显示速度，因为这些图像已经由网页浏览器缓存在用户的计算机中）。这个网页中另一个有用的小细节是页面右上角突出的 About Us 图标——一个小指针指向 About Us（如果想在线浏览这个站点，可访问 <http://www.tallent.com/>）。

图 22.4

单击图 22.3 中的 About Us 将进入这个网页。该网页的图形主题使用户能够马上确定它属于同一个网站



**By the way**

**注意：**虽然我将 tallent.com 网站作为简单网站的例子，但实际上它是一个 Web 咨询公司的完整网站。这在某种程度上使该网站成为更好的例子，因为它保持了简单的感觉，同时展示了大量的信息。我们应以创建“简单”网站为目标，不管要共享的信息有多少。

**注意：**这里要说明一下，tallent.com 网站使用了 Flash，这是一种动画技术，可以实现一些非常有趣的、使用 HTML 和 CSS 无法实现的视觉效果。Flash 超出了本书的范围，它在这个网站的使用与网页的样式和统一性无关，而后者才是本章要讨论的。tallent.com 开发人员不用这种交互式动画，只使用 HTML 和 CSS 也可以实现同样的风格。实际上，有些用户更喜欢简单的浏览体验。

By the Way

**提示：**图 22.4 所示的网页避免了网页制作初学者经常犯的另一个错误；我将其称为“网站建设综合症”。如果你经常在 Internet 上冲浪，相信你和我一样很讨厌那些小巧的道路建设工人图标和只写着“建设中”的死网页。记住，将网页放在 Internet 上时，就已经将它们发布出去了，就像出版社出版 1 万份一样。没有出版商会在手册、书或报纸中打印出没用的一页只写着“建设中”。如果网页还没有准备好，不要将它放到网上，也不要其他网页中添加指向它的链接。开始时网站简单一点没有关系，准备好后再逐渐增加内容。

Did you Know?

## 22.3 组织较大型网站

对于复杂的网站，将精心设计的布局和图像统一地应用于所有页面，有助于组织网站和改进网站的风格。下面看一个需要展示大量信息给多类用户的网站，看看如何使艺术与组织一同发挥作用。

图 22.5 显示了我的个人网站 <http://www.michaelmorrison.com/> 的主页。我知道这听起来有点像自我推销，但是我的网页确实是在布局和组织上复杂度适当的典范。我的网站包含了许多网页，其内部包括四十多本计算机书籍、杂志文章、可下载的代码和游戏、新闻、博客、论坛和其他内容。如果没有有效而一致的整体网站规划，就不可能将所有这些信息都放在我的网站上。

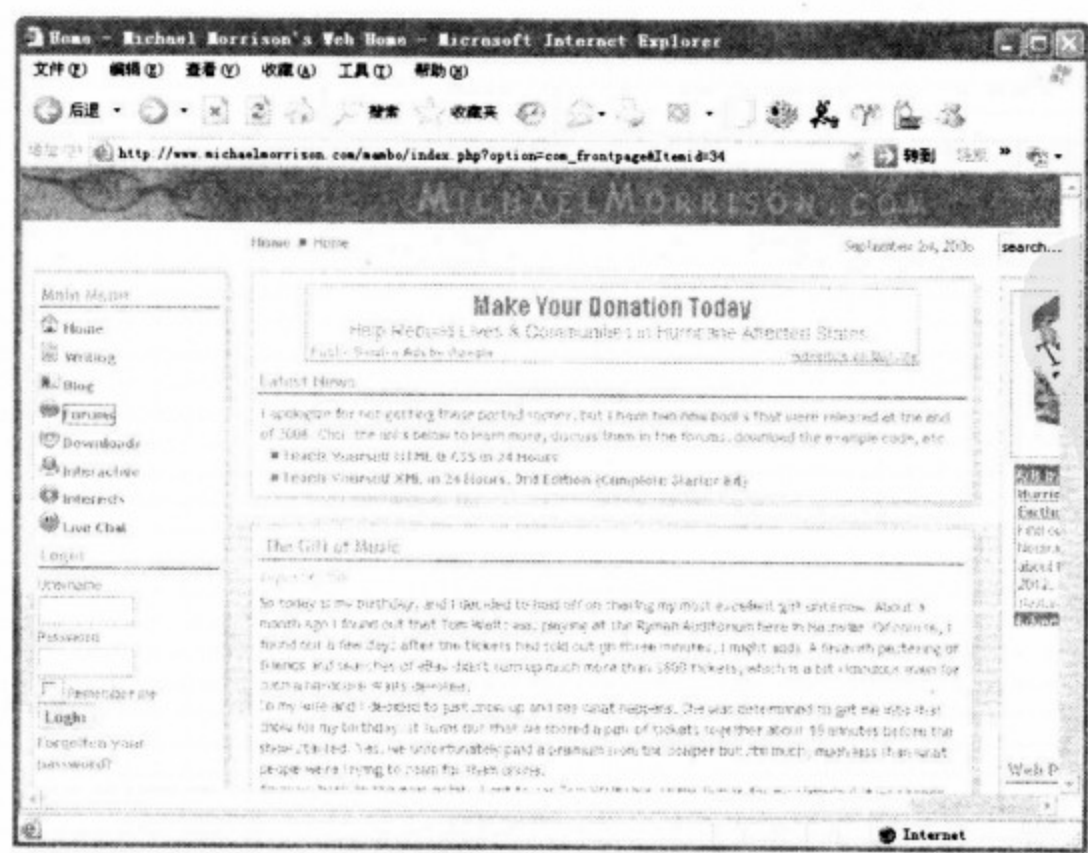


图 22.5  
我的个人网站左边的链接指向大量信息



访问者看到的第一个网页应解释网站的内容，并提供足够多的介绍性信息来吸引目标用户。在我的网站上，主页包含简要新闻，然后是最新的博客文章。这使主页的内容不断地更新，因此吸引用户定期返回来浏览。当然，不是每个用户都希望知道我住的地方的天气，但网站可以包含一些有趣的内容。

**By the Way**

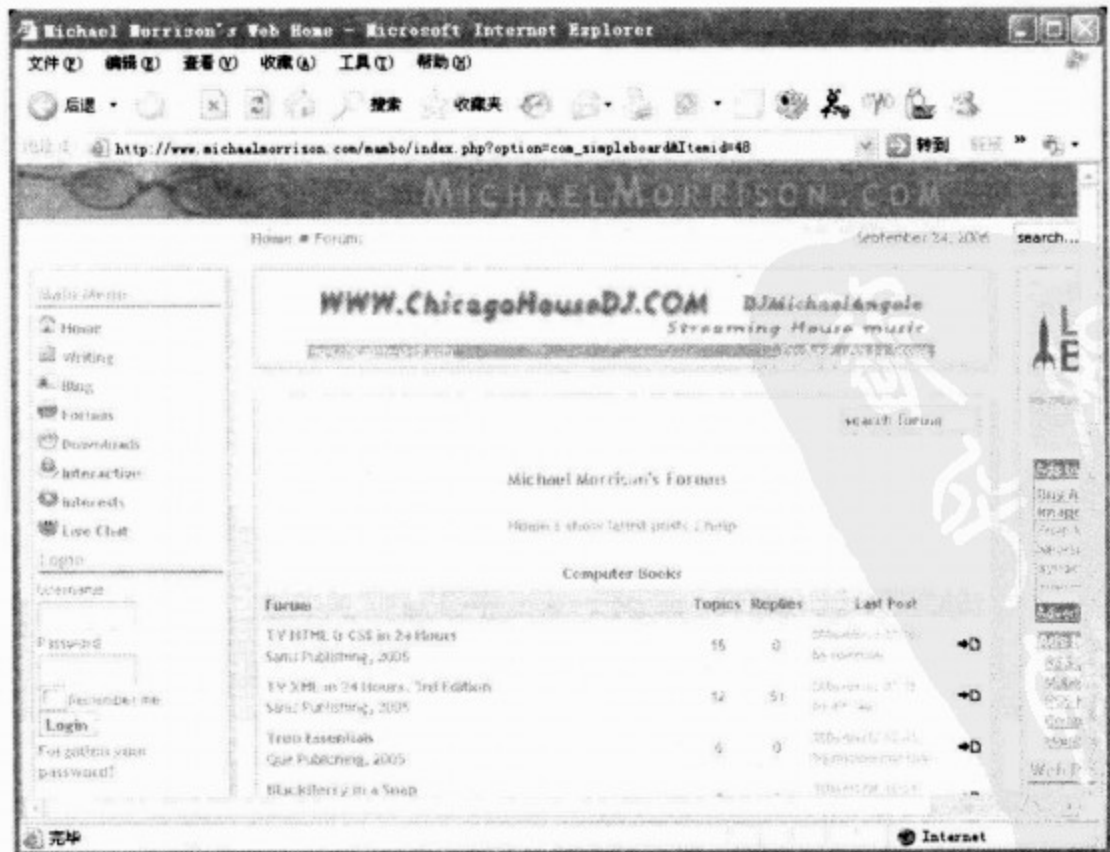
**注意：**我的网站使用了启动网页（splash page），用于在开始时显示标题或其他介绍性信息，然后重定向到网站的主要部分。除提供概要性介绍外，我启动网页还有一个很重要实际用途，让我的旧版本网站与新版本网站（图 22.5 所示）并存。要“激活”网站的新版本，只要修改启动网页的重定向即可。

**Did you Know?**

**提示：**你可能批评我的网站没有强调对访问者的好处。在网页的最开始就解释为什么它值得用户继续看下去，这是绝对必要的，特别是对于商业网站或要满足有些用户需要的网站来说。研究表明，网页只有 3~5 秒的时间来抓住访问者的注意力，否则他们可能转向别处，永远不再回来。要明智地利用这几秒钟！（提示：“欢迎来到……的主页”这一句话是不是在浪费最开始的两秒钟？）当然，对于我的网站，大多数用户都来自我的书（如本书）的读者，因此已经想浏览我的网站，看看上面都有些什么。

如图 22.5 和图 22.6 所示，我的网站组织成七个主要的分类（Home 和 Administrator 除外），从每个页面左边的菜单链接图像可以进入对应的分类。注意，之所以选择这些图标，旨在让访问者能够快速访问满足其需求的资源。应根据访问者的问题或需求来组织网站的主要链接，而不一定是根据信息本身的结构。例如，可以从介绍我的书的具体网页直接导航到特定的论坛，也可以在主页上单击 Forums 一次浏览所有论坛。这种冗余很重要，因为它让访问者能够通过多个导航路径获取信息，而不必在不感兴趣的材料上浪费时间。

图 22.6  
单击图 22.5 中的 Forum 将进入这个网页



单击图 22.5 中的 Forum 将进入图 22.6 所示的网页。这个网页的图像和颜色主题与主页一致，页面左边的菜单也与主页一致（在图 22.6 中，网页向下滚动了一点，以便更多地显示论坛）。这个网页的整体框架与网站的所有网页基本相同，只是每个网页中间主要的内容区域不同。访问者不会进入死网页或感觉在没有组织好的混乱网页结构中“迷路”。

**注意：**我的个人网站使用了 HTML、CSS 和编程语言 PHP、数据库 MySQL。这些技术一起形成一个被称为内容管理系统的软件包。之所以选择这样的系统，是因为我的旧网站是完全使用 HTML 和 CSS 创建的，而我又需要通过论坛与读者进行大量交互，因此网站难以维护。第 24 章将介绍了内容管理系统的基础知识以及是否在自己的网站中使用这样的系统。

**By the Way**

回到一般性网站设计，别忘了这样的一个事实：研究不断表明，人们同时面对遇到超过 7 种选择时，将感到困惑不安；人们面对 5 种以下选择时感觉最舒服。因此，应尽可能地避免连续展示超过 5 个链接（不管是列表还是图标），一定要避免通过使用 7 个以上的链接。需要在导航列表中展示超过 7 个链接时，可将它们分为多个列表，每个列表包含 5~7 个链接。

如果网页中不指向主页的链接不超过两个（或最多三个），也有助于访问者更好地导航；在访问者浏览完从属网页后，也应让读者能够返回到主分类网页（或主页）中。换句话说，应尽量设计比较扁平的链接结构，使大多数网页到主页的链接深度只有一两级。你肯定不希望用户严重依赖于浏览器的“返回”按钮来导航你的网站。

**提示：**在我的网站中的任何网页中，单击菜单顶部的 Home，访问者将返回主页（见图 22.5）。这个链接使用了绝对地址（<http://www.michaelmorrison.com/>）而不是相对地址（index.html），这样当用户将这个网页保存在本地硬盘中时，单击这个链接仍可以访问这个在线网站。关于绝对地址和相对地址，请参见第 3 章。

**Did you Know?**

## 22.4 编写易于维护的 HTML 代码

如果你编写过程序，就知道编写可维护的代码有多重要。如果没有任何编程经验，可维护性指的是你或其他人以后在阅读你的代码时，不会完全看不懂。困难在于要使代码尽可能容易理解。相信我，你肯定会有这样的时刻，阅读自己编写的网页时完全不知道当时想的是什么或为什么这样编写代码。幸运的是，有一种方法可以解决这种问题！

### 22.4.1 使用注释说明代码

开发 HTML 页面时，请牢记你或其他人总有一天需要对它进行修改。简单的文本网页通常很容易阅读和修改，但包含图像、表格和其他布局技巧的复杂网页就较难理解。

在 Internet Explorer 中单击“查看”菜单，然后选择“源文件”，或者在 Firefox 中单击“查看”，然后选择“网页源文件”，看看任何一个网页的源文件，你就会明白我说的。你很可能



看到一堆混乱的、难以解读的 HTML 代码。这主要是因为大量的商业网站是使用内容管理软件系统生成的，因此结构很不清晰。为方便维护网页，建议调整网页代码使其更有序。

第 17 章提到过，可以将为自己或合作者添加的注释放在标签 `<!--` 和 `-->` 之间。在浏览器中浏览时，这些注释不会显示在网页上，但是使用文本编辑器、字处理器、Web 开发工具或网页浏览器的“查看源文件”命令查看 HTML 代码时，可以看到这些注释。下面是一个如何使用注释的例子：

```
<!-- This image needs to be updated daily. -->

```

在这段代码中，`<img />` 标签之前的注释对如何使用图像进行了解释。有人读到这段代码时，就知道这是一个必须每天更新的图像。注释中的文本将完全被浏览器忽略。

**By the Way**

**注意：**要在 JavaScript 代码中包含注释，在每行注释前加上 `//`（JavaScript 注释不需要结束标签）。在样式表中，注释以 `/*` 开始，以 `*/` 结束。  
HTML 标签 `<!--` 和 `-->` 标签不能用于脚本或样式表！然而，可以也应该在 `<script>` 或 `<style>` 标签后包含 `<!--`，并在与之匹配的 `</script>` 或 `</style>` 标签前包含 `-->`。注释标签可以防止老式浏览器将脚本或样式当成普通文本显示在网页上。

**Did you Know?**

**提示：**注释的一种用途是隐藏网页中正在建设的部分。可以通过注释使正在建设的文本和图像隐藏起来，而不是显示它们然后解释它们正在建设中。这样可以逐步构建网页，只显示完成的部分。

**练习**

现在花些时间检查已创建的所有网页、脚本和样式表，添加一些有助于以后查看网页源文件的注释。下面是一些建议。

1. 在比较复杂的格式化或布局技巧使用的标签前添加注释，对其进行解释。
2. 在 `<img />` 标签前使用注释，简要地描述那些根据 alt 信息无法知道其功能的重要图像。
3. 考虑使用一条（或多条）注释总结 `<table>` 标签中的单元格是如何组织的。
4. 如果使用了十六进制的颜色代码（如 `<div style="color:#8040B0">`），插入一条注释说明它实际表示什么颜色（蓝紫色）。
5. 将注释缩进，使注释和 HTML 更易于区别和阅读。HTML 本身也要缩进，使其更容易阅读。



## 22.4.2 缩进使代码更清晰

在本书中，我一直在灌输一种 HTML 代码开发的风格，但没有明确指出。现在到了讲解

这种风格的时候了。你肯定注意到了，本书的所有 HTML 代码都使用一种缩进模式。具体地说，每个子标签都比父标签缩进两个空格。另外，标签内部超出一行的内容在标签内也缩进。

要了解缩进的重要性，最好的方法是看一些没有缩进的 HTML 代码。大家都知道这样的歌词：you don't know what you've got [']til it's gone（直到失去才意识到曾经拥有的东西）。下面是没有缩进的简单表格代码：

```
<table>
<tr><td>Cell One</td>
<td>Cell Two</td></tr>
<tr><td>Cell Three</td>
<td>Cell Four</td></tr>
</table>
```

不仅没有缩进，且表格内的行和列也没有分开。现在，将这段代码与下列描述同一各表格的代码进行比较：

```
<table>
  <tr>
    <td>Cell One</td>
    <td>Cell Two</td>
  </tr>
  <tr>
    <td>Cell Three</td>
    <td>Cell Four</td>
  </tr>
</table>
```

缩进的代码显然更清晰地表示了行和列是如何通过<tr>和<tr>标签分开的。

在我看来，对 HTML 代码的易读性和可维护性来说，一致的缩进风格甚至比注释还重要。你不一定要使用我的缩进策略。如果使用三个或四个空格，也是可以的。如果要使代码更紧凑，不缩进标签内的内容也没有问题。重要的是要形成自己的编码风格并坚持使用。

**提示：**如果你与其他人一起工作或计划与其他人一起开发网页，应考虑建立一个小组，制定一致的编码风格。

**Did you know?**

## 22.5 总结

本章提供了一些示例和解释，帮助你将网页组织成一个信息丰富、引人入胜、易于导航的网站。网络用户已经非常精明，他们希望网站设计良好，如果他们觉得你的网页因设计粗劣而难以导航，就马上会抛弃你的网页。

本章还讨论了通过加入注释和缩进使 HTML 代码易于维护的重要性。注释不仅可以在以后重看代码时提示你，而且可以指导继承代码的人。缩进可能像是美学问题，但确实有助于

一看网页就能快速地分析和理解网页的结构。

## 22.6 问与答

问：我见过的一些网页，它们要求访问者进入主页前调整浏览器窗口宽度或其他设置。这是为什么？

答：这些网页制作者的意思是，如果可以稍微控制访问者的窗口或字体大小，他们可以提供更好的展示。当然，很少人会改变其设置，所以这些网站看起来比较奇怪或者难以阅读。最好使用在这本书学到的技巧使网站在任何窗口大小和各种浏览器设置中都是容易阅读的，且是吸引人的。网站组织得越好，对访问者来说就越有用。

问：浏览网页时，较多的注释和空格会使网页加载变慢吗？

答：网页中少量的额外文本相对于其他更大的网页资源（如图像）来说，是可以忽略不计的。另外，较慢的拨号调制解调器通常会很好地压缩传输的文本，所以加入空格调整HTML格式通常根本不会影响传输时间。加入数百个注释才会在网页载入时导致一秒的额外延迟。请切记，现在很多人都拥有宽带连接（电缆、DSL等），文本传输是极其快速的。是图像使网页传输速度变慢，所以要尽量压缩图像（参阅第7章），而放心地使用文本注释。

问：你可不可以看看我的网页，然后给我一些如何改进的建议？

答：我真的很乐意去做。真的。然而，如果我查看所有读者的网站并对每个提供一个小建议，每天将花掉几个章。我看过几百个读者的网站。我的意见通常是这样的，你的网站很好看或很难看，你已经具备HTML基本知识，但还需要弄清楚：首先，在第一个句子或标题中明确指出网站是给谁看的；第二，你的网站到底能够给他们提供什么好处；第三，你希望他们看完你的网站后如何做。世界上什么好的图像和HTML技巧都不能替代清晰、一致地对网站访问者回答这三个问题。换句话说，内容是决定性的！集中于宏观规划，然后使用HTML系统性地实现它。

## 22.7 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 22.7.1 测验

1. 哪三种方法可以帮助人们体会到你所有的网页组成一个连贯一致的网站？
2. 在人们首次访问的网站看到的主页中，应包含哪两种信息？

3. 如果想对网页的未来编辑说“Don't change this image of me. It's my only chance at immortality”（不要更换这幅图像，这是我惟一的不朽机会），但不希望浏览网页的人看到这种信息，该如何做？

## 22.7.2 答案

1.

- (a) 使用一致的背景、颜色、字体和样式。
- (b) 在所链接到的网页顶部都使用相同的链接文本或图像。
- (c) 在网站每个网页中都使用相同的小标题、按钮或其他元素。

2.

- (a) 足够的信息，让他们能够立刻知道网站的名称以及网站内容是哪方面的。
- (b) 任何要传达给目标读者的重要信息，都要直接而简洁地表述出来。无论是使命声明还是营销口号，都要保证它们清晰地展示出来。

3. 在<img />标签前加入如下注释：

```
<!-- Don't change this image of me.  
      It's my only chance at immortality. -->
```

## 22.8 练习

拿一支铅笔（老式的那种）画出网站草图，草图由小矩形和它们之间的箭头组成。画出每个页面的草图，用弯线指出文本的位置，用涂鸦表示图像的位置。每个箭头从涂鸦图标开始，该图标对应于导航到箭头指向的网页的按钮。即使拥有最优秀的网站管理工具，但手工画出网站草图也可以让你非常直观地理解网站中哪些网页易于访问，以及相邻网页的布局如何协同工作，这些都应在投入时间开始编写实际 HTML，将这些网页连接到一起之前完成。信不信由你，我开始设计网站时仍像这样画出网站草图。有时没有东西能胜过铅笔和纸！



## 第 23 章

# 帮助人们找到你的网页

网页只有被用户访问时才有用；如果没有人能够找到你的网页，你辛辛苦苦地创建就是白费。本章将介绍的 HTML 标签和技巧不会使网页在视觉方面不同，但这些标签和技巧非常重要，有助于目标用户更容易看到你的网页。对于大多数网页制作者来说，本章可能是最简单的，但却是最重要的。你将学习如何在有人搜索与你的话题或公司相关的关键词时，使指向你的网页的链接出现在大多数主要的 Internet 搜索网站上。没有什么魔法可以使你出现在每个搜索列表的前面，但有许多可靠而有效的技巧可确保你的网站很容易被发现。

除给出一些提高网页“可搜索性”的技巧外，本章还将介绍如何使网页自动加载另一个网页、如何在访问者要访问的网页已经移动时将其重定向以及如何指出网页的完整 Internet 地址。

### 23.1 宣传网站

你肯定希望你的网页能够吸引别人的注意，否则你不会花功夫来创建它们。如果只是将网页放到局域网或公司内部网上，或者只是通过磁盘或电子邮件来发布网页，那么，帮助人们找到你的网页可能就不是问题。然而，如果将网页加入到 Internet 上浩如烟海的网页中，则让目标用户能够找到你的网页将是巨大的挑战。

要解决这个问题，需要对大多数人如何决定要浏览哪些网页有基本了解。下面是人们获知网站的三个基本途径：

- 有人告诉他们并提供网站的地址，他们可以直接在网页浏览器中输入该地址；
- 从其他人的链接进入你的网站；
- 在搜索网站（如 Google、Yahoo! 或 MSN Search）上的列表中找到你的网站。

如果花些时间和功夫，可使通过这三种途径找到你的网站的人更多。要使听说你的网站的人增多，需要通过口耳相传以及其他任何交流渠道。如果有联系人数据库或邮件列表，可通过它们将网站向这些人公布。将网站的地址添加到你的名片或公司的宣传册中。还有，如果有钱，可以利用电视和广播将你的 Internet 地址广而告之。总之，进行市场营销。传统



的口头营销仍是最好的方法，即使是在 Internet 上。

使其他网站链接到你的网站也很简单，但这并不意味着工作量很小。要找到每一个与你的主题相关的其他网站，并承诺如果它们添加到你的网站的链接，你也会这样做。如果存在有关你的主题有专门目录，不管是在线的还是印刷的，要确保你的网站位列其中。我不能在这本书中讲太多东西来帮助你做到这一些，惟一的办法就是付诸行动。

我可以帮助你的主要是第三点：使你的网站出现在主要的 Internet 搜索网站上。相信你至少用过一两个大型搜索网站：Google、Yahoo!、MSN Search、AllTheWeb、Alta Vista、Ask Jeeves 和 Teoma 等，这些网站的地址分别是 google.com、yahoo.com、search.msn.com、alltheweb.com、altavista.com、askjeeves.com 和 teoma.com。

**注意：**搜索引擎的流行和有用性并非一成不变的。换句话说，有些搜索引擎随着 Web 社区的发展以及它们如何执行搜索的细节而流行起来，最终又退出潮流。因此，可以访问 <http://www.searchenginewatch.com/>，看看对最流行搜索引擎的最近评估。

**By the Way**

这些网站基本上都是巨大的数据库，将 Internet 上海量的网页进行分类。它们都使用自动化程序来创建数据库，但有些网站（如 Yahoo!）强调质量，对每个列表进行人工检查。而其他网站（如 Teoma）更倾向于数量，几乎全部依靠网络机器人（robot）或网络蜘蛛（spider）。网络机器人或网络蜘蛛是一个自动化计算机程序，整天搜索 Internet 上的网页并根据它访问的网页内容创建数据库。还有一些网站（如 Google）除使用机器人方法外，还使用非常复杂的方法根据网页如何被其他网页链接对网页进行排名。

网络蜘蛛和人力不断将网页添加到数据库中，而另一个自动化程序（搜索引擎）处理人们查询特定主题网页的请求。搜索引擎在数据库中查找包含用户要查询的关键词或词组的网页，然后将包含这些词的所有网页的列表发送给用户。有人将那些主要使用人工而不是机器人创建数据库的搜索引擎称为 Internet 目录（Directory）。

**注意：**有些搜索引擎称为门户（portal），这表示它们不仅仅提供搜索功能。像 Yahoo!这样的门户，除基本的搜索功能外，通常还提供新闻、天气、购物和其他类型的信息。许多 Web 用户已经开始欣赏像 Google 这样不包括门户功能的搜索引擎。

**By the Way**

## 23.2 在主要的搜索网站列出你的网页

如果希望人们找到你的网页，就必须向每个主要的搜索网站提交请求，让它们来索引你的网页。每个搜索网站都会提供一个表格，让你填写网站的地址及简单的介绍，有时候还要求网站相关的分类或关键词列表。这些表格很容易填写，一个小时就可以填完，省下的时间可用于将你的网站在一两个专门目录中列出（如何找到这些专门目录呢？当然是通过主要的搜索网站）。

虽然在主要搜索引擎中列出很简单也很快，但也令人困惑：对于要注册网页必须单击的地方，每个搜索网站使用的术语各不相同。表 23.1 可帮助减少困惑，它列出了每个主要搜索

引擎的地址以及注册时要单击的链接词。

表 23.1 向搜索引擎注册网站

搜索引擎	在哪里注册网页地址
Google	访问 <a href="http://www.google.com/addurl/">http://www.google.com/addurl/</a> ，输入网站的地址及简单介绍，然后输入页面上显示的花体验证文本。最后单击 Add URL 按钮，将网站添加到 Google。
Yahoo! Search	访问 <a href="http://submit.search.yahoo.com/free/request">http://submit.search.yahoo.com/free/request</a> ，输入网站的地址，然后单击 Submit URL 按钮。
Yahoo! Directory	访问 <a href="http://dir.yahoo.com/">http://dir.yahoo.com/</a> ，进入网站应加入到的目录，并单击浏览器窗口右上角的 Suggest a Sit。按照说明提交网站；如果你的网站是免费的，务必选择免费选项。
MSN Search	访问 <a href="http://search.msn.com/docs/submit.aspx">http://search.msn.com/docs/submit.aspx</a> ，输入验证文本和网站地址，然后单击 Submit URL 按钮。
AllTheWeb	AllTheWeb 的搜索结果是由 Yahoo! Search 提供的，因此只要将网站提交给 Yahoo! Search，如上所述。
Alta Vista	Alta Vista 的搜索结果也是由 Yahoo! Search 提供的，因此只要将网站提交给 Yahoo! Search，如上所述。
Ask Jeeves	发送电子邮件给 <a href="mailto:url@askjeeves.com">url@askjeeves.com</a> ，要求对你的网站进行评估。
Teoma	Teoma 和 Ask Jeeves 使用相同的搜索索引，因此对 Ask Jeeves 的请求也适用于 Teoma。

URL 表示统一资源定位器（Uniform Resource Loactor），是网页地址的同义词。

**Did you know?**

**提示:**有些网站提供可自动提交给所有主要搜索引擎和一些小型搜索引擎的表单，如 <http://www.scrubtheweb.com/>、<http://www.submitexpress.com/> 和 <http://www.hypersubmit.com/>。在这些网站当中，有许多也提供将网站列在其他许多目录和索引下的收费服务。这些服务对你来说可能有价值也可能没有，这取决于你的目标用户，但我强烈推荐你直接到前面列出的主要搜索网站上，使用它们的表单来提交请求。这样可以确保准确地回答问题（每个网站的问题都有点不一样），并确保你确切地知道网站将如何列在每个网站中。

在提交请求前，先看完本章的内容，否则将遇到很严重的问题，并错过解决它的最佳时机。

要明白我所说的，想象一下这样的场景：你发布了一个销售自动蟑螂碾平机的网页。我正好遇到了蟑螂问题，且对昆虫喷雾过敏。我打开笔记本，扫除键盘上的蟑螂，登录到最喜欢的网站，然后输入“蟑螂”作为搜索关键词。搜索引擎搜索到大约 10254 个 Internet 网页，并将前 10 个列出。你提交了将网页在搜索结果中列出的请求，因此知道你的网页会显示在结果列表中。

我提过我很有钱吗？我提过有两只蟑螂在我的脚上吗？在我所处的区域，你还提供了当天送货的服务。那么，你是希望你的网页出现在列表中的第 3 位还是第 8542 位？现在你应该理解这个问题了。网页仅出现在搜索结果列表中是不够的，还需要使网页的排名靠前。

### 23.3 为搜索引擎提供提示

事实上，你绝对无法确保在主要搜索引擎中使用特定的单词或词组进行搜索时，自己的网站出现在搜索结果的前十名中，也就是说，不能从搜索网站购买广告空间。毕竟，如果可以的话，每个人都希望自己的网站列在第一名。你能够做只是避免自己的网站显示在列表的最后，使自己与其他人有同等的机会出现在最前面。

每个搜索引擎都使用不同的方法来确定哪些网站是最为相关的，因此应列在搜索结果列表的前面。但你不必太关心它们的区别，因为它们都使用一些基本的标准组合。下面的列表包含了几乎每个搜索引擎在评价哪些网页与关键词最匹配时都会考虑的因素。前三个标准是每个主要搜索引擎都使用的，这些搜索引擎中的大部分至少还使用一两个其他的标准。

- 关键词是否出现在网页的<title>标签中？
- 关键词是否出现在网页的前几行中？
- 关键词在整个网页中出现了多少次？
- 关键词是否出现在网页的<meta />标签中？
- 关键词是否出现在网页的图像文件名和 alt 文本中？
- 我的网站有多少网页链接到该网页？
- 其他网站中有多少网页链接到该网页？又有多少其他网页链接到这些网页？
- 在上一次搜索列表结果中该网页被选中了多少次？
- 该网页在人工生成的目录中排名是否靠前？

**注意：**Yahoo!和 Ask Jeeves 在搜索引擎中很独特，添加到它们的目录中的网站由人工进行分析和归类。

**By the  
Way**

很显然，未将排名提前，最重要的工作是考虑目标用户最可能输入的关键词组合。建议不要关心常用的单个单词的检索——在这种情况下，生成的列表通常很长，试图使网页出现在前面就像买彩票一样。相反，应将重点放在与你的主题相关相关性最强的、不常用词和两三个词的组合上。确保这样的单词和词组多次出现在网页中，并将最重要的词放在<title>标签以及第一个标题或介绍性段落中。

**注意：**有些激进的网页制作者在网页中将同一个单词重复几十个甚至上百次，有时候以很小的字体或难以看见的颜色显示，这样做只是为了使搜索引擎在该关键词的搜索结果中将该网页显示在前面。这种做法称为搜索引擎欺骗（spamming）。

不要做这种事情，所有主要的搜索引擎都反对这种做法，它们设置了“欺骗探测器”，检查以可疑的模式重复相同的词或词组的网页，一旦发现立刻将其从数据库中删除。网页中多次出现重要的搜索关键词是可以的（也有好处），但一定要在正常的句子或词组中使用这些关键词，这样才不会被欺骗探测策略删除。

**By the  
Way**

在上面列出的所有搜索引擎评估标准中，误解最多的可能是<meta />标签。有些人以为使用它们网页就立即能够出现在每个搜索列表的前面。另一些人却认为<meta />标签没什么用。这两种极端想法都是错误的。

<meta />标签是一个通用标签，可放在任何文档的<head>部分中，用于指定关于网页的不属于<body>文本的信息。大多数主要搜索引擎都查找<meta />标签是否提供了对网页的简单介绍和标识网页的一些关键词。例如，你的自动蟑螂碾平机订单表单可能包括下列两个

标签:

```
<meta name="description"
content="Order form for the SuperSquish cockroach flattener." />
<meta name="keywords"
content="cockroach,roaches,kill,squish,supersquish" />
```

### By the Way

**注意:** `<meta />`应放在`<head>`、`<title>`和`</title>`之后,但在结束标签`</head>`之前。根据 XHTML 标准,在每个文档的`<head>`部分中,`<title>`必须是第一个标签。

在这个例子中,第一个标签确保搜索引擎将网页显示在搜索结果列表中时,对网页有准确的描述。第二个`<meta />`标签使得搜索查询中包含指定的关键词时,你的网页排名将靠前一点。

要使网页被搜索引擎索引,应在网页中包含`<meta />`标签,并指定属性 `name="description"` 和 `name="keywords"`。这样做对网页在搜索列表中的排名可能不会有很显著的效果,也不是所有的搜索引擎都会查找`<meta />`标签,但可能有所帮助。

### Did you Know?

**提示:** 搜索引擎专家建议,在`<meta />`标签中,网页描述的理想长度是 100 ~ 200 个字符。对于关键词,推荐长度是 200 ~ 400 个字符。专家还建议不要在关键词之间浪费空格,这在蟑螂的例子中很显然。最后,不要疯狂地重复同一个关键词——如果做得过头,一些搜索引擎会惩罚你。

### Did you Know?

**提示:** 如果不希望网页包含在搜索引擎的数据库中(这种可能性不大),可将如下`<meta />`标签放在网页的`<head>`部分中:

```
<meta name="robots" content="noindex" />
```

这样会使有些搜索引擎忽略你的网页。要使用更健壮的保护措施避过搜索机器人的眼睛,可以请 Web 服务器管理员将你的网页地址包含在服务器的 robots.txt 文件中(他知道怎么做)。这样所有主要搜索引擎都会忽略你的网页。这适用于不想被公开搜索的公司内部网页。

程序清单 23.1 和图 23.1 所示的网页是一个说明如何改进搜索引擎结果的具体例子。这个网页应该很容易找到,因为它涉及一个具体的话题,并多次包含一些不常见的技术术语,让对这个主题感兴趣的人搜索。然而,还可以采取一些措施,以提高网页在搜索结果列表中的排名。

#### 程序清单 23.1 在 Internet 网站搜索中将出现一些问题的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
```

```

<head>
  <title>Fractal Central</title>
</head>

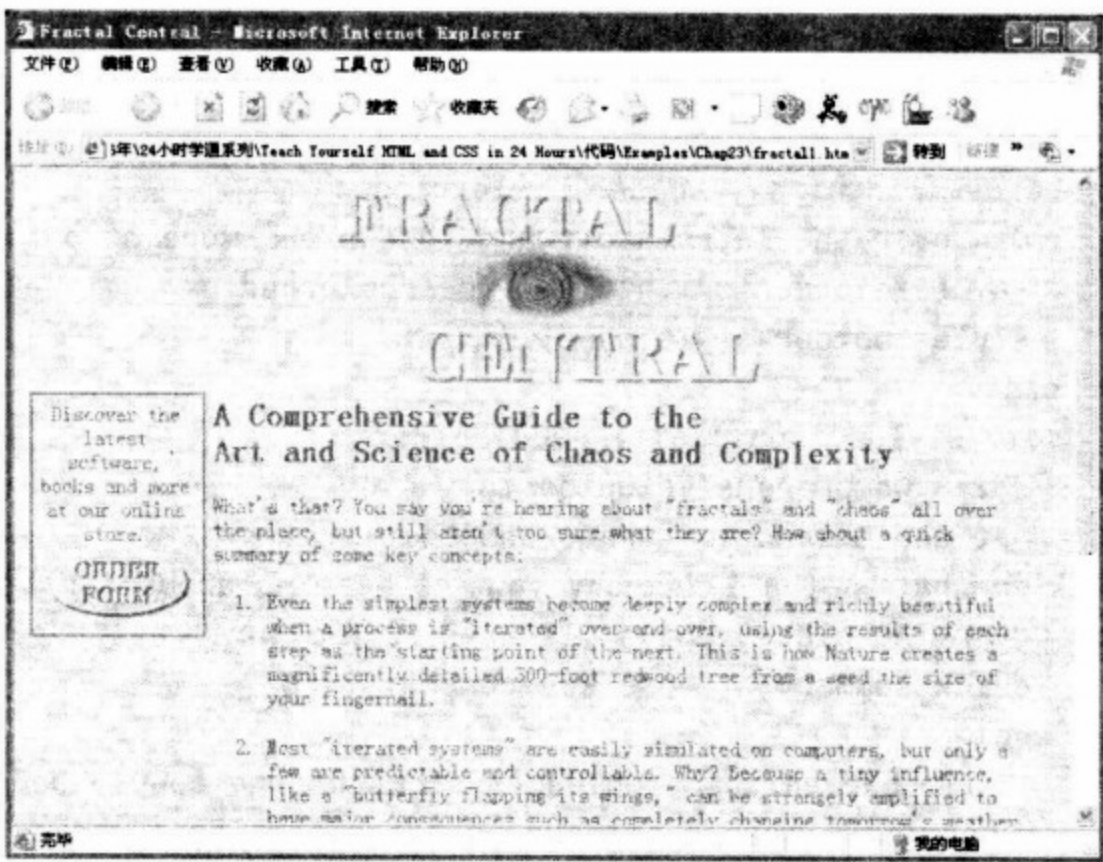
<body style="background-image:url(fractalback.jpg); color:#003399">
  <div style="text-align:center">
    
  </div>
  <div style="width:133px; float:left; padding:6px; text-align:center;
border-width:4px; border-style:ridge">
    Discover the latest software, books and more at our online store.<br />
    <a href="orderform.html"></a>
  </div>
  <div style="float:left; padding:6px">
    <h2>A Comprehensive Guide to the<br />
    Art and Science of Chaos and Complexity</h2>
    <p>What' s that? You say you' re hearing about "fractals" and "chaos" all
    over the place, but still aren' t too sure what they are? How about a
    quick summary of some key concepts:</p>
    <ol>
      <li><p>Even the simplest systems become deeply complex and richly
      beautiful when a process is "iterated" over and over, using the
      results of each step as the starting point of the next. This is how
      Nature creates a magnificently detailed 300-foot redwood tree from a
      seed the size of your fingernail.</p></li>
      <li><p>Most "iterated systems" are easily simulated on computers,
      but only a few are predictable and controllable. Why? Because a tiny
      influence, like a "butterfly flapping its wings," can be strangely
      amplified to have major consequences such as completely changing
      tomorrow' s weather in a distant part of the world.</p></li>
      <li><p>Fractals can be magnified forever without loss of detail, so
      mathematics that relies on straight lines is useless with them.
      However, they give us a new concept called "fractal dimension" which
      can measure the texture and complexity of anything from coastlines to
      storm clouds.</p></li>
      <li><p>While fractals win prizes at graphics shows, their chaotic
      patterns pop up in every branch of science. Physicists find beautiful
      artwork coming out of their plotters. "Strange attractors" with
      fractal turbulence appear in celestial mechanics. Biologists diagnose
      "dynamical diseases" when fractal rhythms fall out of sync. Even pure
      mathematicians go on tour with dazzling videos of their
      research.</p></li>
    </ol>
    <p>Think all these folks may be on to something?</p>
  </div>
  <div style="text-align:center">
    <a href="http://netletter.com/nonsense/"><img src="findout.gif"

```



```
alt="Find Out More" style="border-style:none" /></a>
</div>
</body>
</html>
```

图 23.1  
程序清单 23.1 所示网页的第一部分在网页浏览器中的显示结果



对访问者来说，程序清单 23.2 和图 23.2 中的网页内容与程序清单 23.1 和图 23.1 几乎相同。然而，对于搜索机器人和搜索引擎来说，这两个网页非常不同。下面的列表总结了其中的修改以及这样修改的原因。

- 在页面的<title>标签和第一个标题中添加了一些重要的搜索关键词。原来的网页甚至没有在这两个关键的地方包含单词 fractal。
- 添加了<meta />标签，为搜索引擎提供描述和关键词。
- 在第一个<img />标签中添加了极具描述性的 alt 属性。不是所有搜索引擎都会读取并索引 alt 文本，但的确有一些搜索引擎这样做。
- 删除了技术术语（如“fractal”和“iterated”）的引号，因为有些搜索引擎认为“fractal”和 fractal 是两个不同的单词。使用字符实体&quot;来表示引号，搜索引擎不会考虑这个字符实体。这是个好主意，因为 XHTML 鼓励 Web 开发人员使用&quot;实体而不是直接使用引号。
- 在订单表单框的文本中两次添加了关键词 fractal。

程序清单 23.2 对程序清单 23.1 所示网页的改进

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
```

```

<head>
  <title>Fractal Central: A Guide to Fractals, Chaos, and Complexity</title>
  <meta name="description" content="A comprehensive guide to fractal
  geometry, chaos science and complexity theory." />
  <meta name="keywords" content="fractal, fractals, chaos science, chaos
  theory, fractal geometry, complexity, complexity theory" />
</head>

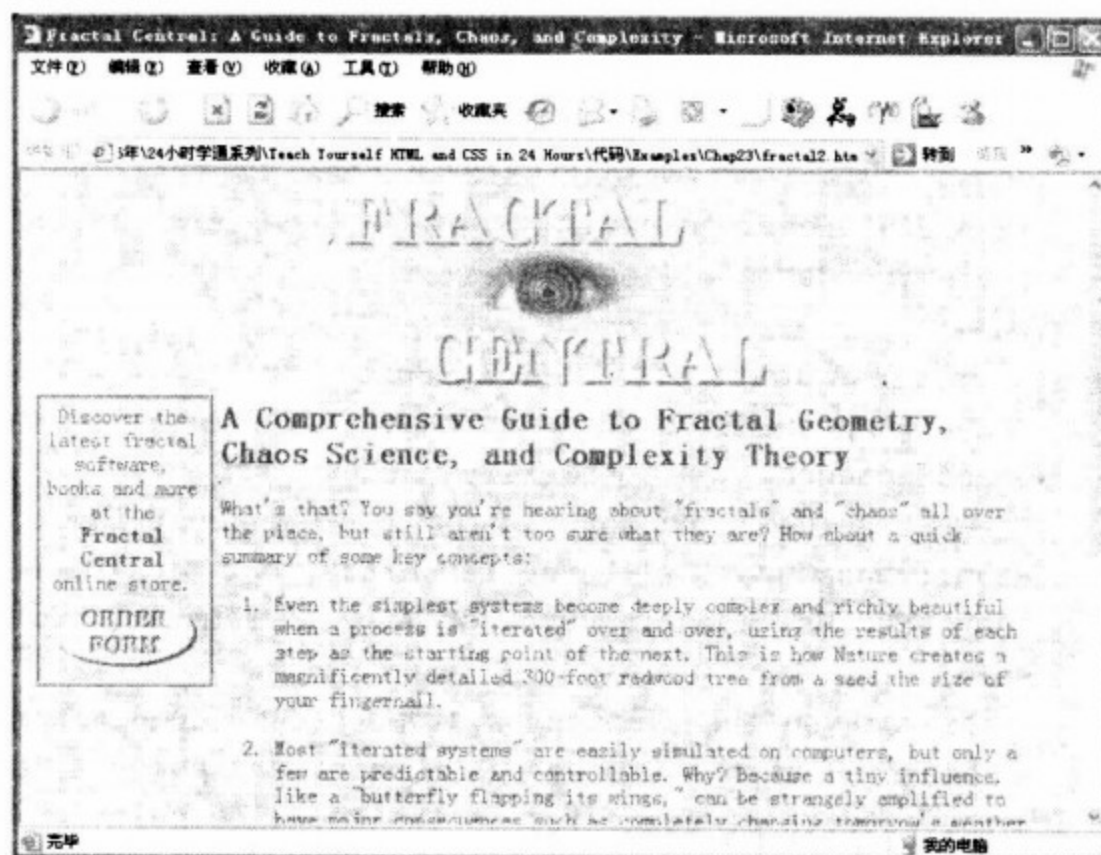
<body style="background-image:url(fractalback.jpg); color:#003399">
  <div style="text-align:center">
    
  </div>
  <div style="width:133px; float:left; padding:6px; text-align:center;
  border-width:4px; border-style:ridge">
    Discover the latest fractal software, books and more at the
    <span style="font-weight:bold">Fractal Central</span> online store.<br />
    <a href="orderform.html"></a>
  </div>
  <div style="float:left; padding:6px">
    <h2>A Comprehensive Guide to Fractal Geometry,<br />
    Chaos Science, and Complexity Theory</h2>
    <p>What's that? You say you're hearing about &quot;fractals&quot; and
    &quot;chaos&quot; all over the place, but still aren't too sure what
    they are? How about a quick summary of some key concepts:</p>
    <ol>
      <li><p>Even the simplest systems become deeply complex and richly
      beautiful when a process is &quot;iterated&quot; over and over, using
      the results of each step as the starting point of the next. This is
      how Nature creates a magnificently detailed 300-foot redwood tree from
      a seed the size of your fingernail.</p></li>
      <li><p>Most &quot;iterated systems&quot; are easily simulated on
      computers, but only a few are predictable and controllable. Why?
      Because a tiny influence, like a &quot;butterfly flapping its
      wings,&quot; can be strangely amplified to have major consequences
      such as completely changing tomorrow's weather in a distant part of
      the world.</p></li>
      <li><p>Fractals can be magnified forever without loss of detail, so
      mathematics that relies on straight lines is useless with them.
      However, they give us a new concept called &quot;fractal
      dimension&quot; which can measure the texture and complexity of
      anything from coastlines to storm clouds.</p></li>
      <li><p>While fractals win prizes at graphics shows, their chaotic
      patterns pop up in every branch of science. Physicists find beautiful
      artwork coming out of their plotters. &quot;Strange attractors&quot;
      with fractal turbulence appear in celestial mechanics. Biologists
      diagnose &quot;dynamical diseases&quot; when fractal rhythms fall out

```

```
of sync. Even pure mathematicians go on tour with dazzling videos of
their research.</p></li>
</ol>
<p>Think all these folks may be on to something?</p>
</div>
<div style="text-align:center">
<a href="http://netletter.com/nonsense/"></a>
</div>
</body>
</html>
```

图 23.2

程序清单 23.2 所示网页的第一部分在网页浏览器中的显示结果



无法量化人们在搜索 `fratal` 和 `chaos` 时，能够找到程序清单 23.2 和程序清单 23.1 所示网页的概率差，但这些修改肯定能够提高搜索引擎找到该网页的可能性。通常来说，针对搜索引擎所做的改进也可能使网页的主题对于人来说更容易理解。这使得针对搜索引擎的网页优化是双赢的！

### By the Way

**注意：**如果你读过最近几年一些流行的计算机杂志，可能见到有人称 XML（“HTML 的未来”）可以使在 Internet 搜索信息更容易。你可能想知道如何使你创建的网页实现这种新的搜索奇迹。

好消息是，XML 确实能够使在线搜索更容易、更高效；坏消息是现在甚至在不久的将来 XML 及其衍生物 XHTML 并不能使人们更容易找到你的网页。依赖于 XML 来改进 Web 搜索的技术现在还在开发中，未来几年内可能不会成为主流。使用 XHTML 和 CSS 来代替旧的 HTML 是朝着更有组织因而也可搜索的 Web 方向发展的第一步。

## 23.4 自动加载另一个网页

管理网站时，可能需要将有些网页从一个地方移到另一个地方。例如，可能决定更换服务提供商或整个网站的域名。可能只是重新组织网站，将一些网页移到另一个目录下。

那么，如果用户将你的网页加入收藏夹，然后你移动了这个网页，将发生什么呢？如果不希望访问者面对“没有可以显示的页面”时的无助，应在旧地址放上一个网页，指出“该网页已移至……”，并链接到新地址。

你很可能在 Internet 上遇到过类似的消息，其中的一些网页用到接下来将介绍的技巧；自动使访问者在几秒后进入新地址，即使访问者没有单击链接。

实际上，可以让任何网页在指定的时间后自动加载其他网页，其秘诀是在网页的<head>部分使用<meta />标签的变种，如下所示：

```
<meta http-equiv="refresh" content="5; url=nextpage.html" />
```

其中，5 表示加载下一个网页前等待的时间，可将其替换为所需的秒数；并将 nextpage.html 替换为要加载的网页的地址。

除将访问者定向到新网页地址外，自动页面加载也可显示网页标题一段时间，然后使访问者进入网站的主要区域。例如，请看程序清单 23.3 的代码，这个网页在 Mozilla Firefox 中显示的结果如图 23.3 所示。5 秒后，<meta />标签导致网页 <http://www.stalefishlabs.com/news.html>（如程序清单 23.4 和图 23.4 所示）被加载并显示。

**注意：**我也针对没有耐心的人提供了一个指向 news.html 页面的链接，他们可以在 5 秒钟结束前单击这个链接。

**By the  
Way**

程序清单 23.3 <meta />标签使网页浏览器在 5 秒后自动加载图 23.4 所示的网页

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Stalefish Labs. Dead Serious About Fun.</title>
    <meta name="description" content="Stalefish Labs.
    Dead Serious About Fun." />
    <meta name="keywords" content="games, toys, fun, social, entertainment,
    tall tales, tall tale, topper, zinger" />
    <meta http-equiv="refresh"
    content="5; url=http://www.stalefishlabs.com/news.html" />
    <base href="http://www.stalefishlabs.com/" />
  </head>

  <body style="text-align:center">
```

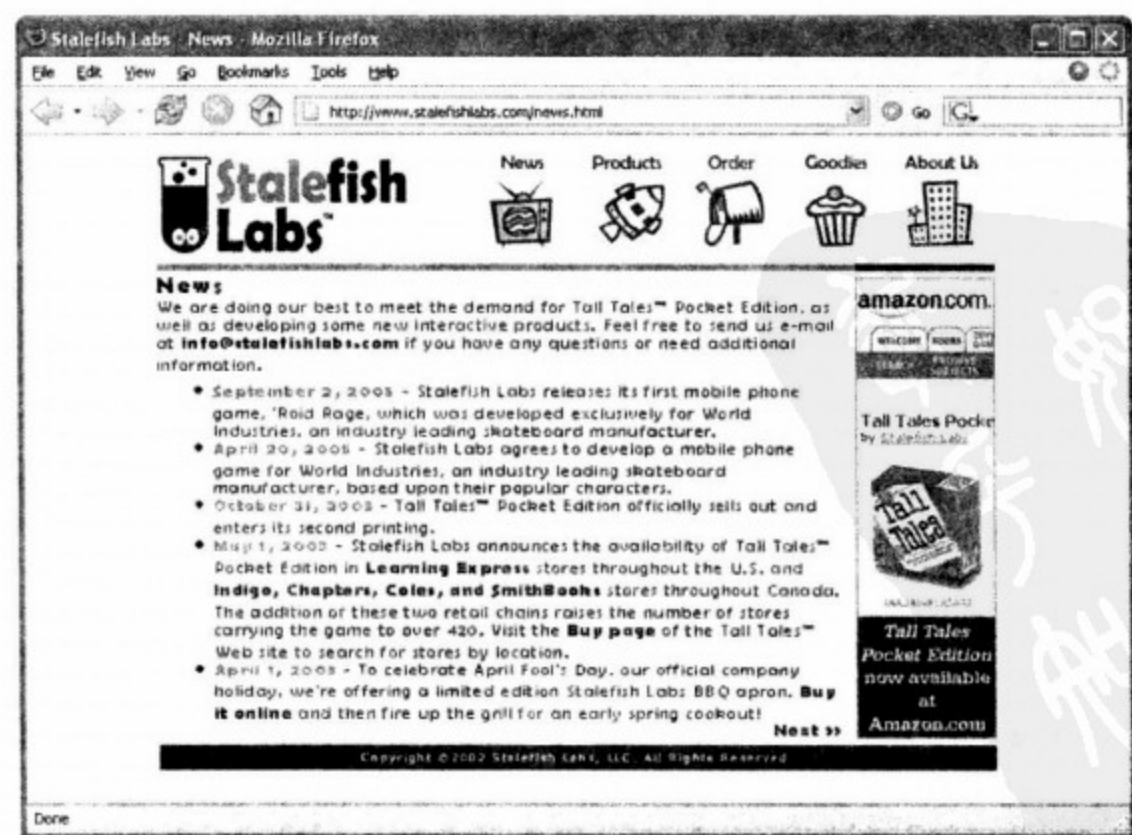


```
<div style="width:100%; margin-top:100px; text-align:center">
  <a href="news.html"></a>
</div>
<div style="width:100%; text-align:center">
  <a href="news.html"></a>
</div>
</body>
</html>
```

图 23.3  
这是程序清单 23.3 所示的网页，它对网站进行介绍



图 23.4  
程序清单 23.3 和图 23.3 所示的网页在 5 秒后自动加载这个网页





介绍 Stalefish Labs 的网页包含其他的一些 HTML 小技巧，在你继续下面的学习前需要指出。首先，使用前面介绍的<meta />标签的描述和关键词，使网站在搜索引擎中的搜索效果更好；其次，第一个<div>标签使用了样式属性 margin-top，在页面的标题图像上方留出一些空间。这改善了网页的整体外观，不管浏览器窗口的大小如何。

这个网页的最后一个技巧是使用<base />标签，它位于网页的<head>中。下面将介绍为什么这个标签很重要。

## 23.5 指出网页的完整地址

假设你创建了一个为业务做广告的网页，客户很喜欢该网页并将它保存在本地硬盘上。几天后，他想向朋友介绍你的网站，这时将发生什么？他没有将网页加入书签，当然页面没有包含指向自身的链接。他单击指向你的订单表单的链接，然而，这些链接中只包含文件名（如<a href="orderform.html">）；因此在他的硬盘上无法运行，除非订单表单也在他的硬盘上。你就这样失去了两个顾客。

避免这种情形的一种方法是，在所有链接中都使用以 http://开始的完整地址。然而，这使网页难以测试和维护。如果移动网站或更换域名，所有链接都将失效。

也可以在每个网页（包括主页）中包含一个指向主页的链接。然而，还有一种更好的方法使网页记住它来自哪里。

<base />标签让你能够在网页的<head>部分包含其地址，如下所示：

```
<html>
  <head>
    <title>My Page</title>
    <base href="http://www.myplace.com/" />
  </head>
  <body>
    ...The actual page goes here...
  </body>
</html>
```

对于维护该网页的 HTML 开发人员来说，<base />标签提供该网页应放在哪里的文档信息。

更重要的是，该网页中所有的链接都将像网页位于<base />指定的地址处那样发挥作用——即使它并不在这个地址中。例如，如果将程序清单 23.3 所示的网页保存在本地硬盘上，然后在网页浏览器中打开它，该页面的所有图像将从在线网站 http://www.stalefishlabs.com/ 加载，而不是从本地硬盘加载。实际上，这些图像保存在 Web 根目录下的 images 文件夹中，这就是为什么该程序清单中的链接在图像文件名前加上了 images/ 的原因。指向 news.html 的链接也指向根目录 http://www.stalefishlabs.com/ 下的网页，而不是硬盘上的本地网页。

**警告：**对于在<meta />标签中指定的在一定延迟后自动加载的网页的 URL，<base />标签没有影响。这就是为什么在程序清单 23.3 中，<meta />标签使用了 news.html 的完整 URL 的原因。

**Watch  
Out!**

By the Way

**注意：**很少有网页制作者使用（甚至知道）<base />标签。很多知道它的人也不修改它，因为这样在测试本地硬盘上的网页（包括图像和链接）时比较麻烦。我在本章尽量提供足够的信息，读者可以自己决定是否值得使用<base/>标签——你的决定很可能是不值得。

如果你选择使用<base />标签，要等到准备将网页上传到 Web 服务器时再将它放到网页中。这样可以在本地硬盘上对所有图像和链接进行测试，然后在最后时刻再将<base />标签加进去，从而在网页发布后利用它提供的优点。

23.6 总结

本章介绍了一些极其重要的主题：如何提供提示给搜索引擎（如 Google、Yahoo!、MSN Search、Alta Vista、Ask Jeeves 和 Teoma），让人们能够更容易地在 Internet 找到你的网页。你还学习了如何让网页记下自己的地址以及如何使网页自动加载另一个网页。表 23.2 列出了本章介绍的标签与属性。

表 23.2 第 23 章介绍的标签与属性

标签/属性	功能
<meta />	指定文档的元信息（关于文档本身的信息）。最常见的用途是加入网页描述、指定关键词以及使网页自动加载另一个网页或刷新自己。在文档的<head>中使用。
属性	
http-equiv="服务器命令"	发送命令给 Web 浏览器或服务器。例如，http-equiv="refresh"使 Web 浏览器自动加载一个新网页。
name="名称"	用于指定 content 属性中包含关于文档的哪类信息。例如 name="key-words"表示 content 中包含网页的关键词。
content="值"	http-equiv 或 name 中指定的信息类型的实际信息或值。例如，如果 http-equiv 属性设置为 refresh，content 属性应该设置为要等待的秒数，接着是分号以及要载入的网页地址。
<base />	指定当前文档在 Web 浏览器中的位置的根 URL。这是<head>中的可选标签。
属性	
href="URL"	文档的根 URL

23.7 问与答

问：我的网站有许多网页，我需要在搜索网站为每个网页填写单独的表格吗？

答：不需要。如果只提交主页（它可以链接到所有其他网页），搜索蜘蛛将搜索这个网页的所有链接（还有被链接网页的所有链接，这样不断重复下去），直到它将网站中所有网页都编入索引为止。

问：我提交了在搜索引擎中列出网站的请求，但我的网页从未出现，即使我输入公司的特有名称。我该怎么办呢？

答：大多数大型搜索引擎提供一个表格，填写后你可以立即查询特定地址是否包含在其数据库中。如果发现还没有，可以再提交申请。有时要在提交申请几天甚至几星期后，网络蜘蛛才将你的网页编入索引。Yahoo!对索引申请的处理以慢著称，因为它雇人来检查列出的

每个网页。

问：在<meta />标签中加入关键词时，需要包含每种可能的拼写或大小写吗？

答：不必担心大小写；几乎所有的搜索都是以小写输入的。然而，要包含所有明显的变体或常见拼写错误，将其作为关键词。虽然从理论上说很简单，但在利用<meta />标签时，还有很多本章没有介绍的高级策略。关于<meta />标签更深入的讨论和网站推广，请访问<http://www.selfpromotion.com/>。

问：可以使用<meta />标签使网页每隔几秒或几分钟自动地刷新自己？

答：可以，但通常不需要这样做，除非在 Web 服务器上设置了某种程序或脚本为网页提供新信息。然而，这种技术在有些情况下很有用。例如，有些要显示体育比分等实时信息的网站经常使用这种技术，通过自动刷新，每隔几分钟显示新的比分。

## 23.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 23.8.1 测验

1. 如果发布了一个关于小狗收养 (puppy adoption) 的网页，如何确保在主要的 Internet 搜索网站输入 puppy、dog 与/或 adoption 时可以找到该网页？
2. 假如你最近将网页从 <http://mysite.com/oldplace/thepage.html> 移到了 <http://mysite.com/newplace/thepage.html>，但你不太确定是否会一直将该网页放在那里。如何自动地将进入旧地址的访问者引导到新地址，而不需要发送信息告诉他们这种修改呢？
3. 哪三种方法可以确保将你的网页保存在硬盘中的访问者即使忘记将其加入书签或收藏夹中，也可以通过硬盘中的网页找到你的网站？

### 23.8.2 答案

1. 首先，确保 puppy、dog、adoption 经常出现在主页中（你可能已经这样做了），并使 Puppy Dog Adoption 出现在网页标题中。然后，在网页的<head>部分加入下列<meta />标签：

```
<meta name="description"
content="dog adoption information and services" />
<meta name="keywords" content="puppy, dog, adoption" />
```

最后，将网页发布到网上，并访问主要搜索引擎的网站提交页面（本课前面已列出），填写每个网站的提交表单。

2. 将下面的网页放在 <http://mysite.com/oldplace/thepage.html>：

```
<html>
```

```

<head>
  <meta http-equiv="refresh" content="0;
    http://mysite.com/newplace/thepage.html">
</head>
<body>
</body>
</html>

```

3.

(a) 在每个页面中包含一个指向该网站的链接，要使用完整的 Internet 地址，下面是一个例子：

```

The address of this page is:
<a href="http://www.mysite.com/home.html">
http://www.mysite.com/home.html</a>

```

(b) 页面之间的所有链接都使用完整的 Internet 地址，下面是一个例子：

```

This is my home page. From here you can
<a href="http://mysite.com/personal.html">
find out about my exciting personal life</a>, or
<a href="http://mysite.com/work.html">
find out about my boring work life</a>.

```

(c) 使用<base />标签指定网页的 Internet 根地址，下面是一个例子：

```

<head>
  <title>My Home Page</title>
  <base href="http://mysite.com/" />
</head>

```

## 23.9 练习

你能够想出一些有趣和有用的使用网页自动修改的方法（使用<meta http-equiv="refresh">标签）？你肯定能够。

## 第 24 章

# 超越传统网站

在将来的许多年之内，读者在本书学到的所有知识都可能完美无缺地应用于 HTML 兼容的软件。现在有数千万的网页是标准 HTML 编写的，即使标准发展变化，未来的网络浏览器和商业软件仍能够浏览现在的网页。本书重点介绍 HTML 的最新版本：XHTML，它作为网页创作语言正在逐步替代 HTML。

然而，一些最令人兴奋的 HTML 高级应用仍在快速发展。本章介绍最新的 HTML 扩展，帮助读者理解利用这些新功能可以做什么。

---

Web 技术总是在不断改变，这意味着不应只依赖于书籍来了解 HTML 或 XHTML 的最新进展。因此，我希望读者通过如下网站努力跟上 Web 技术的发展步伐。

- 了解最新 HTML 标准（提议的未来标准）的两个最佳资源是 World Wide Web Consortium 网站（<http://www.w3.org/>）和 HTML Compendium 网站（<http://www.htmlcompendium.org/>）。
- 要了解这些标准在最新的网络浏览器中实际上是如何实现的以及有哪些非标准 HTML 扩展可用，可访问微软的网站（<http://www.microsoft.com/>）、Mozilla Firefox 网站（<http://www.getfirefox.com/>）和 Opera 网站（<http://www.opera.com/>）。Macintosh 用户要获悉 Safari 网页浏览器的最新消息，可访问 Apple 网站（<http://www.apple.com/>）。

还可以从这些网站获得网页浏览器的最新更新。

---

### 24.1 超越 Web 的 HTML

不管是现在还是将来，读者通过阅读本书获得的 HTML 知识都是最重要的（且可获利的）技能之一。然而，你在有生之年将创建的大部分 HTML 页面可能都不是网页。



要理解这是为什么以及 HTML 的发展前景, 请看最新 HTML (XHTML) 标准中的下列特性。

- 通过样式表和脚本, HTML 让你能够精确地控制任何文本或图像信息的外观与功能。
- 所有主要的编程语言、交互式媒体和数据库格式都可以与 HTML 无缝地集成。
- 世界上几乎所有人类语言都可以使用 HTML 扩展字符集和字体来交流。
- 数据安全标准使得通过 HTML 进行金融和其他敏感交易、管理机密和限制对信息的访问变成现实。
- 微软 Windows 操作系统的所有未来版本都将继续使用 HTML 作为用户界面的重要组成部分。现在几乎所有提高办公效率的软件都支持 HTML, 其中一些还高度依赖于 HTML (与 XML) 作为标准的数据格式。

所有这些都表明, 在不久的将来, HTML 将毫无疑问地在全世界的几乎所有计算机和计算机网络的信息显示和交换中扮演重要角色——甚至是惟一的重要角色。这听起来很重要, 因为它确实很重要。然而, 本章还将指出, HTML 甚至还将扮演比这更重要的角色。为让读者理解这些, 有必要回过头来看看更广阔的图景: 计算机自身在我们社会中扮演的角色在不断改变。

## 24.2 从计算器到无线通信设备

计算机曾被认为是计算和处理数字的设备, 后来发展成为处理所有信息类型的设备, 从文字和数字到图像和声音。在现在与将来, 计算机首先是通信设备, 其主要用途是在人们之间传送信息。

在当今的大部分工作场所, 只需知道如何单击链接和滚动页面, 就可以使用计算机访问商业信息, 而无需知道哪些信息来自你的计算机的, 哪些来自楼下的服务器, 哪些来自千里之外的其他服务器。

用户已经习惯于在计算机屏幕查看非常容易阅读且具有吸引力的信息页面, 他们已经完全不能容忍含义不清的图标、没有修饰的文本信息以及古怪的迷宫式菜单结构。他们希望计算机屏幕总是像 Web 一样容易阅读与交互。

通过提供计算机软件赚钱的人非常清楚地意识到了这一点, 并为实现这一目标进行了空前的研究与开发。这样发展下去, 关于计算机交互的比喻从 1980 年代的“桌面窗口”变成了 1990 年代“万维网页面”。这个比喻在新千年的前十年已变为满足用户的需要, 无线手持设备与无线网络迅速成为日常生活的必要组成部分。

## 24.3 HTML 作为新的用户界面

随着计算机角色的发展, HTML 在我们使用计算机处理的几乎所有事情中变得越来越中心化了。在将所有类型的信息以可预测和可展示的方式连接起来这方面, HTML 事实上已成为全球标准。

HTML 让你能够以简单、可靠的方式组合和排列文本、图像、声音、视频和交互式程序。

HTML 不同于老式专用页面布局标准，它一开始就是为全球各种计算机之间的高效通信而设计的。

然而，HTML 无处不在并不意味着在未来几年中网络浏览器将是主要的软件应用类型。事实上，作为单独程序的网络浏览器几乎已经消失。例如，Microsoft Internet Explorer 的功能不仅仅局限于从万维网获取网页，它让你能够使用 HTML 页面作为界面，用于组织和浏览自己计算机中的信息，包括文件夹和 Windows 桌面。通过与 Microsoft Office 等支持 HTML 的软件协作，HTML 已成为文字处理、电子表格和数据库的通用标准界面。Mozilla Firefox 也不仅仅是个网页浏览器，它使用 HTML 将所有类型的媒体集成到 Email、讨论组、日程管理、商业文档及协作项目管理中。

**注意：**最新的 Windows 操作系统 Windows Vista，完美地将网页浏览器集成到桌面计算的体验中。换句话说，Web 浏览器作为单独应用程序的概念已成为过去，因为 Web 浏览器已经集成到了计算机体验的每个角落。

**By the Way**

同时，对 HTML 的支持实际上已经包含在每个主要的软件中，使计算机上几乎所有程序都可以采用 HTML 页面的形式导入和导出信息。总之，HTML 就像粘合剂，将计算机上不同类型的信息组合起来，确保信息能够以标准形式展示出来，以相同的外观显示给世界上的每个人。

当 HTML 不能提供足够的结构来准确描述信息时，XML 可作为一种通用数据格式。因此，HTML 足以作为不同类型的可视信息的粘合剂，而 XML 作为不容易看见的信息的粘合剂，如电子定单、自动资金转账、新闻 feed、手机设置等。在 Web 信息的展示方面，XML 提供了一种取代 HTML 的高度结构化选择。

在商业领域，快速、有效地交流被视为职工最常见、最重要的任务，HTML 与 XML 的“信息粘合剂”不仅仅能够连接不同类型媒体，它还是连接企业与客户、将员工组成高效团队的粘合剂。了解如何应用这些粘合剂——即你在本书学到的技巧——将让你能够在任何现代组织中扮演最有价值的角色。

## 24.4 数字媒体革命不会出现在电视上

在未来几年内，最重要的变化不是 HTML 本身，而是通过 HTML 网页可以接触的受众。许多网站开发者认为基于 Internet 的内容能够成为继电视与广播后的大众市场；而不太乐观的观察家则指出，全球通信网络还要发展很长一段时间才能够广泛传送电视质量的视频或在世界上大部分地区广泛使用。尽管如此，高速无线网络（其中许多是免费的）正在快速增长，已经开始支持为每个需要的人提供广泛的高速 Internet 接入的概念。

所有通信行业，从电视到电话，都迅速地向数字技术转移。这样，通信网络间的界线已变得模糊。新的 Internet 协议有望优化多媒体传送，与此同时，新协议让无线广播能够支持双向互动传输。一个小型卫星天线就让你能够接入 Internet，同时传输高清晰度电视。

另外，HTML 是全球范围惟一得到广泛支持、用于组合文本和任何其他数字媒体的标准。无论数字通信未来发生什么令人出乎意料的转变，也很难想象不是以 HTML 作为主导而 XML

作为辅助！

数百万用户已经开始不用“真正的计算机”访问 Internet——通过电视机顶盒、专用 Internet 设备，更不用说所有支持 Web 访问的手机与寻呼机。以后将有更多的设备能够为不使用计算机的人提供 HTML 内容。成熟的手持计算机，如 Treo、BlackBerry 与 Pocket PC 设备，让你能够在任何地方通过无线网络上网。对于不太了解设备的人，流行的家庭视频游戏系统，如 Xbox 360，除在线游戏服务外也提供对 Web 访问。

### By the Way

**注意：**对 HTML 制作者来说，HTML 进入大众市场无疑是极佳的机会。然而，这也给 HTML 页面设计带来了众多的挑战，因为很多人可能在低分辨率的电视屏幕或小型手持设备上查看网页。有关如何确保 HTML 网页能够被广大的受众查看和理解的指引，请参阅下一节。

## 24.5 为未来做准备

如果读者阅读了本书的大部分内容，应具备了将来在新的数字世界取得成功的最重要的元素之一：扎实的 HTML、XHTML 和 CSS 知识。

当前你学习 HTML 的主要原因很可能是为了创建一些网页，但希望本章能够使你相信，今后你将不止在这方面使用 XHTML。下面是你在规划和创建网站时应考虑的一些因素，这样才能为明天做好准备。

- 如果要在网页中实现某种功能，但目前使用 HTML 无法实现，就在网页中包含注释，以便将来可能时添加这种特性。相比于文本和图像部分，网站的多媒体和交互式部分可能需要更多的修改，以新技术的步伐。尽可能将网站中比较先进的内容分离出来，使用注释标签 `<!--` 和 `-->` 对其进行详细说明。
- 即使高带宽的交互性媒体可能成为未来的主流，但不要过分追求。即使拥有快速 Internet 连接的人也可能不希望等待流视频开始播放。在浏览或收听交互式媒体方面，一定要为网站访问者提供选择。
- 样式表让你能够完全控制网页的布局 and 位置，因此如果不熟悉基本布局，现在就应该学习。长期以来，要创建真正专业级的印刷版面，必须理解并使用诸如行距（leading）、字距（kerning）、字号及首字母下沉（drop caps）等。这些很快也将成为创建优秀网页的基本要求。
- 设计网页时，不要总是假设查看网页的每个人使用的都是大显示器。电视、手机、游戏控制台和许多其他设备也可以访问网页。有些设备的屏幕分辨率很低（有些手机为 200×200 甚至更低）。虽然要设计在这样的分辨率显示效果良好的网页很困难，但这样做可以使网页访问者范围尽可能广泛。

### By the Way

**注意：**决定不考虑为屏幕很小的设备设计网页没错。实际上，要设计出在 200×200 和 1280×1024 的分辨率下显示效果都很好的网页通常不太现实。然而，可以考虑为小屏幕设备提供另一个网页版本。本章稍后将学习 XHTML Mobile，它提供了这样的选项。

- World Wide Web Consortium 已经发布了几个相对较新的标准，这些标准对 Web 的影响将随着时间的推移而逐渐增大。这些标准包括：

XHTML Mobile;

Portable Network Graphics (PNG);

Really Simple Syndication (RSS);

Synchronized Multimedia Interface Language (SMIL);

Scalable Vector Graphics (SVG);

Mathematics Markup Language (MathML);

eXtensible Style Sheet Language (XSL)。

在隐私和安全方面，新标准包括：

Platform for Internet Content Selection (PICS);

Platform for Privacy Preferences (P3P);

Digital Signature standard (Dsig)。

这些改进可能提高网站的潜力并改变当前用于创建网页的一些方法，因此应访问 w3.org 网站，花些时间了解这些标准。

**注意：**可以在 Sams Publishing 网站 <http://www.sampublishing.com/> 找到一些在线参考和学习资源。

这个网站除提供一种简单的方法，让读者复习本书介绍的所有示例网页和 HTML 技巧外，还提供许多由于篇幅限制而没有放在书中的示例网页。

还可以找到许多由本书的读者创建的网站链接。你肯定能够从中找到一些好主意，用在自己的网页上！

**By the  
Way**

## 24.6 未来的 HTML 应用

HTML 和 XHTML 普遍的兼容性促使人们将任何重要的文档格式化为网页，即使没有计划要立即将它发布到万维网上。可以创建单个网页，用于打印在纸上、作为电子邮件发送、在会议上显示或将其发布到公司内部网以供参考。也可以使用传统的方法，针对每种应用对页面进行格式化——当信息需要更新时使用不同的软件程序对每个文件进行编辑。由于大多数商业软件都支持 HTML 标准，许多组织要求员工对于所有重要的文档都使用 HTML。

可移植到 HTML 的不是以前认为的只有“文档”。结合使用 XML、样式表、JavaScript 和其他技术，基于 HTML 的表示在很多情况下可以取代以前各种专用数据格式、专用软件和传统的编程语言。下面是 HTML 在 Web 外的一些其他领域的应用。

- 包含基于 HTML 的交互内容的信息亭现在如雨后春笋般涌现。它们看起来像自动取款机，可帮助销售唱片和电影票、拓展商场柜台，甚至自动收取停车费用。
- 信息丰富的 CD-ROM 或 DVD-ROM 正快速地向 HTML 转移。大不列颠百科全书现



在完全基于 HTML，这使得公司可用 CD-ROM、Web 或这两种方式的结合快速、实时更新地提供内容。因为 CD-ROM 驱动器显示多媒体内容比大多数 Internet 连接快，因此，因媒体密集而不能在今天的万维网上实现的动态 HTML 显示将成为可能。DVD-ROM 更快，且可存储更多内容，这使它特别适合于大型多媒体“网站”。

### By the Way

**注意：**当然，即使基于 CD-ROM 的百科全书已经过时，HTML 也在更新。在第 6 章学习了如何将网页链接到 Wikipedia，这是一个公众可编辑的在线百科全书，在我看来它根本地改变了管理信息的方式。Wikipedia 理论上允许所有人使用 HTML 对它进行编辑。如果你已经忘了它有多酷，可参见 HTML<http://www.wikipedia.org/>。

- 基于 HTML 的企业新闻稿现在通常是以 HTML 创建的，用于企业内部网，并打印在纸上发送给不能在 Web 上看到它们的员工或客户。在线与纸张呈现的区别在于，用于打印的图像必须是高分辨率的、黑白的，用于计算机屏幕的图像是低分辨率的。然而，现在彩色打印机已经比较便宜，它们使得低分辨率的彩色图像在基于 HTML 的新闻稿中的效果很好。
- 教师们现在发现，测试和学生作业单使用 HTML 页面更容易管理，且可以包含无法包含在纸上的多种交互式内容。可以将简单的 HTML 文档放在可写的 CD 或存储卡中发送给不能访问 Internet 的学生。
- 第 21 章介绍了博客及如何创建自己的博客。随着越来越多人依赖于 HTML 创建的博客来共享信息和讨论从政治、技术到运动等各种话题，博客已经成为美国文化中的一股强大力量。
- 与博客类似，用 HTML 创建的在线照片共享服务，如 Flickr (<http://www.flickr.com/>) 和 Google's Hello (<http://www.hello.com/>)，让人们能够向朋友、家人和公众提供一种前所未有的了解其生活的途径。
- 垂直市场 (vertical market) 用户购买计算机通常旨在运行为特定客户而设计的程序。提供这些系统的增值销售商和系统集成商现在提供的机器都配置成开始就显示 HTML 页面。这可以在用户使用机器的过程中提供帮助，或用一种更容易交互、更完善的界面来取代老式傻瓜菜单，而不会影响易用性。
- Google 发布功能强大到不可思议的在线地图应用 Google Maps 时，整个 Internet 世界都震惊了，该应用是使用 HTML、XML 和 JavaScript 创建的。在第 3 章中，你学习了如何将网页链接到 Google Maps。当卡特里娜飓风在墨西哥海湾肆虐，主要的新闻网如 CNN 和 MSNBC 显示飓风过后的详细卫星图像时，使用了 Google Maps 的另一个应用 Google Earth，从中可以看出这种技术的影响有多深远。

我可以列出更多 HTML 在普通网页外的创造性应用，但我要说明的已经很清楚：要呈现任何类型的信息，考虑使用 HTML，而不是几年前使用的专用软件或编程工具。

本章接下来将介绍一些具体的 Web 技术，它们让你能够创建超越基本 HTML 限制的网页。虽然这些技术肯定超出了 HTML，但它们全都依赖于 HTML。



### 24.6.1 RSS 和网页聚合

Really Simple Syndication (RSS) 是一种让你能够使用 XML 标记语言来创建及聚合新闻提要 (feed) 的技术。RSS 有多种用途, 你可以在自己的网站上显示来自其他网站的 RSS 摘要; 可以创建自己的 RSS 摘要库, 然后使用称为新闻集合器的软件定期浏览新闻; 还可以使用 RSS 聚合自己的网站, 让其他人能够浏览你的摘要。

当我说 RSS 让你能够聚合 Web 内容时, 指的是可以向网站订阅, 从而不必访问该网站就可以很容易地找出网站的更新。使用称为新闻聚合器的专门软件, 可以监视来自多个站点的新闻摘要, 可以高效地获得大量信息的更新, 而不必定期在每个网站上停留。可以将 RSS 理解为网页数据的“存货闹钟”——当你喜欢的网站发布了新的信息时, 它让你能够及时地得到更新。

学习 RSS 最简单的方法是安装新闻聚合器软件并自己探索新闻摘要。现在最流行的新闻聚合器是 FeedDemon, 在 <http://www.feedException.com/> 可下载免费的试用版本。FeedDemon 是一个独立的桌面应用, 完全独立于网页浏览器。

**注意:**其他流行的在线新闻聚合器包括 Bloglines(<http://www.bloglines.com/>)、NewsIsFree (<http://www.newsisfree.com/>) 和微软试验性的 start.com (<http://www.start.com/>)。Google News (<http://news.google.com/>) 也是一种聚合器, 虽然它不像真正的 RSS 摘要管理器那样提供那么大的灵活性。Google News 可能是功能最强大的摘要生成器, 因为用它很容易地聚合 Google News 分类甚至 Google 搜索。

By the  
Way

图 24.1 显示了用于浏览新闻摘要的 FeedDemon 桌面聚合器。

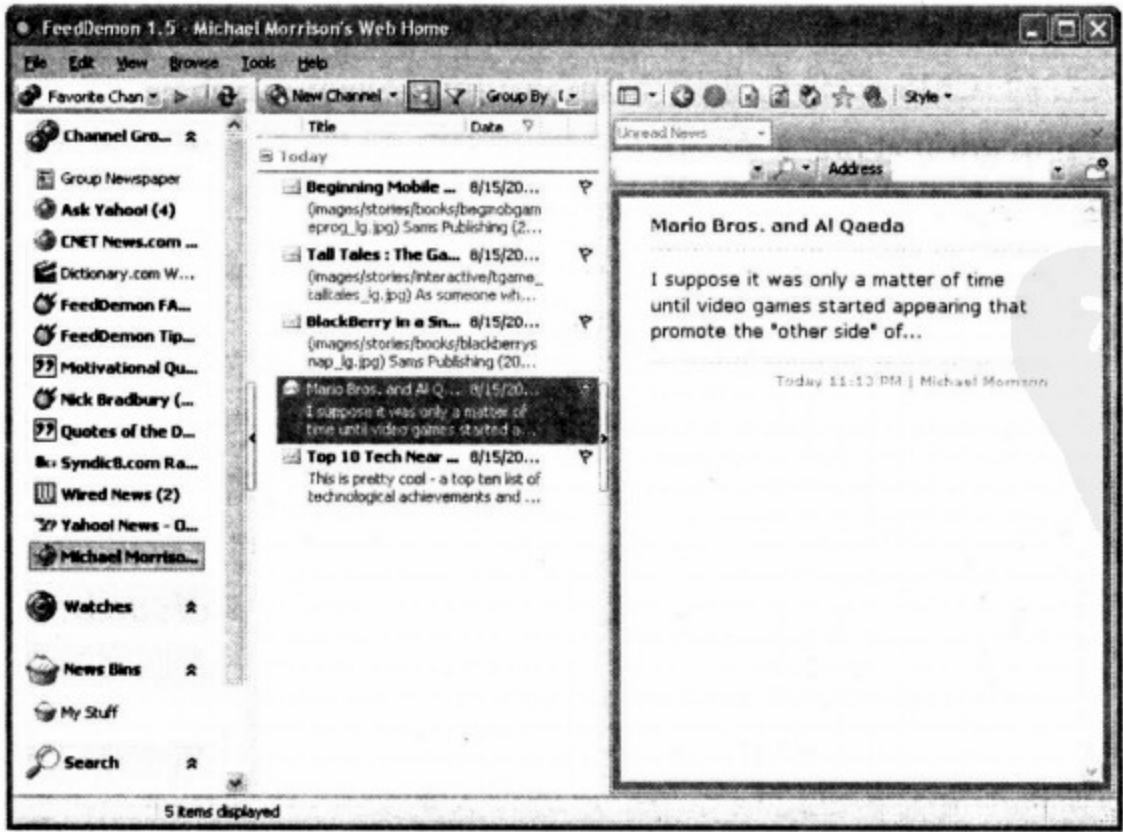


图 24.1  
FeedDemon 是主要的桌面新闻聚合器之一, 其价格合理

如果仔细观察上图，将发现其中的新闻摘要来自于我位于 [michaelmorrison.com](http://michaelmorrison.com) 的新闻摘要。使用 FeedDemon 很容易管理多个新闻摘要来源，而不会感觉被信息淹没。习惯使用 FeedDemon 这样的新闻聚合器后，很快将认识到访问和处理 Web 内容有多么高效。现在，我通过新闻摘要就能获得更新，因此很少自己访问不同的站点。

### By the Way

**注意：**不幸的是，如何创建自己的 RSS 新闻摘要不在本书的范围内。然而，如果读者对此感兴趣，建议访问 [http://en.wikipedia.org/wiki/Really\\_Simple\\_Syndication](http://en.wikipedia.org/wiki/Really_Simple_Syndication)，阅读 Wikipedia 的 Really Simple Syndication 项，以了解更多的知识。

## 24.6.2 使用内容管理系统

你肯定访问过一些有大量用户交互的网站，这些用户交互可能是讨论组、私有消息、投票和其他形式的动态内容和交流功能。你可能知道，仅使用 HTML 很难创建这样的网站。即使结合使用 XHTML、CSS 和 JavaScript，在动态内容管理方面也存在一些局限性，有大量双向用户交互的网站通常要求动态内容管理。

创建高交互式网站通常都需要大量的开发工作。具体地说，大多数这样的网站都要求使用数据库管理系统 (DBMS) 和服务端脚本技术，如 Microsoft 的 Active Server Pages (ASP) 或 PHP (PHP Hypertext Preprocessor)。总之，要创建这样的网站，需要大量本书没有介绍的技能，更不要说要花相当多的时间和资源。

你也可以下载一些免费软件轻松地完成这项工作——利用他人辛勤劳动的成果。用于创建高交互性、数据驱动的网站软件称为内容管理系统 (CMS)。虽然有一些昂贵的商业软件，但也有许多开源的 (免费的) 内容管理系统供你使用。要使用这样的系统，首先要下载软件并安装到 Web 服务器上。然后必须完成配置过程，确保 CMS 成功地连接到数据库。这样，就可以将大部分时间花在组织网站和开发网站内容上——通常是通过基于 Web 的用户界面。

### By the Way

**注意：**流行的 CMS 软件包包括 Mambo (<http://www.mamboserver.com/>)、Drupal (<http://drupal.org/>) 和 e107 (<http://e107.org/>)。这些软件包我用过许多，我的个人网站 <http://www.michaelmorrison.com/> 实际上就是使用 Mambo 创建的。因为我需要通过论坛、实时聊天、投票和定期更新的博客与读者进行交互，因此从传统网站转向 CMS 是非常值得的。

CMS 与传统网站相比具有一些显著的优点，因为 CMS 网站通常是完全由数据库中的数据驱动的。例如，CMS 网站中的大多数网页都根据存储在数据库中的代码动态生成。这让你能够在任何地方直接在网页浏览器中编辑并创建网页。我在旅行时仍定期更新我的网站，只要用 Internet Explorer 登录并访问网站的管理员区域就即可。要创建更生动的 Web 发布系统而不是静态的 HTML 页面，确实需要考虑使用 CMS。

如果 CMS 是你需要的，那为什么还要花时间来学习一整本介绍 HTML 的书呢？因为 CMS 并不适合于每个人。如果不需要通过论坛等与用户进行大量交互，可能根本不需要 CMS。另外，CMS 网站难以进行“装饰”使其具有独特的风格。许多 CMS 网站看起来非常相似，

这令人很不愉快。如果花时间和精力为 CMS 网站创造独特的风格，也需要牢固的 HTML 和 CSS 知识。

这引出了最后一个与 CMS 网站相关的问题。即使决定将 CMS 软件包用于网站，而不是自己创建传统的 HTML 页面，懂得 HTML/XHTML 和 CSS 也非常重要。你无疑希望在 CMS 内创建一些不寻常的网页，甚至可能还想修改整个网站的风格。我见过的每个 CMS 软件包在很大程度上都依赖于 CSS 来创建页面的布局和颜色主题。很显然，所有页面本质上都是 HTML 或 XHTML。因此，简单地说，只要进行 Web 开发，HTML/XHTML 和 CSS 知识就不会无用武之地。

要体验一些 CMS 软件包，看看它们是如何工作的，可参见 <http://www.opensourcecms.com/>。这个网站提供许多开源 CMS 软件包的运行模拟，你可以登录到这些系统进行体验。另外，我的个人网站 <http://www.michaelmorrison.com/> 是一个 MamboCMS 站点，而 <http://www.musiccitymafia.com/> 是一个 e107 CMS 站点。

### 24.6.3 使用 XHTML Mobile 编写网页

虽然我谈到支撑 Web 的语言时通常指的是 HTML，但实际上你在本书中已经学习了 XHTML。XHTML 还有另外一个版本我没有提到，那就是 XHTML Mobile，它是满足移动设备有限需求的 XHTML 精简版。移动设备包括手持或便携技术——手机、寻呼机、联网小管家（connected organizers）、手持 PC 等。与桌面计算机相比，这种设备通常都有比较小的屏幕、较少的内存以及限制更大的用户界面（通常没有键盘或鼠标）。

在 XHTML Mobile 出现前，主要有两种移动 Web 服务：WAP 和 iMode。iMode 由 NTTDoCoMo 创建，主要流行于日本和欧洲的部分地区。WAP 由多家移动行业巨头创建，是移动网页的全球标准。这个 WAP 版本为 WAP 1.0，尽管获得了成功，它仍遭到了许多批评。WAP 1.0 和 iMode 依赖于自己的标记语言来编写网页代码。具体地说，WAP 1.0 基于 WML（Wireless Markup Language，无线标记语言），而 iMode 基于 cHTML（Compact HTML，精简 HTML）。这两种语言都是移动网页的基本标记语言，然而，当发展到更强大的基于 XML 的无线 Web 时，它们在很多方面都不能够胜任。

XHTML Mobile 包含 WML 和 cHTML 的许多核心特性。当前大多数移动浏览器都支持 XHTML Mobile，且 XHTML Mobile 支持使用 CSS。你在本书中已经见过多次，CSS 可以控制网页的格式化和显示的许多方面。我这里说 CSS 也适用于 XHTML Mobile，实际上指的是 CSS 的一个子集，称为 WCSS（Wireless CSS），它是经过精简的 CSS 版本。换句话说，WCSS 之于 XHTML Mobile 犹如 CSS 之于 XHTML。

XHTML Mobile 的优点在于它与 XHTML 没有区别——除受到的限制更多外。因此，如果你了解 XHTML，也就了解了 XHTML Mobile，只是可使用的工具更少，因为 XHTML Mobile 比 XHTML 的限制的更多。一般来说，XHTML Mobile 可以完成你想在移动网页上实现的所有功能。

程序清单 24.1 是一个在无线应用中使用 XHTML Mobile 的例子。它包含了一个电影描述移动网页的代码，你可以在手机上浏览这个网页，然后决定要不要去看电影《金刚》。

## 程序清单 24.1 描述电影《金刚》的 XHTML Mobile 网页

```
1: <?xml version=" 1.0" encoding=" UTF-8" ?>
2:
3: <html>
4:   <head>
5:     <link rel=" stylesheet" type=" text/css" href=" movie.css" />
6:   </head>
7:
8:   <body>
9:     <h1 align=" center" >King Kong</h1>
10:    <p>This remake of the 1933 classic follows an expedition to the
11:      mysterious Skull Island, where a legend of a giant gorilla draws
12:      explorers and filmmakers. The legend, however, is both real and
13:      dangerous, living in a massive jungle that has protected him and
14:      other prehistoric creatures for decades. Kong finds solace in a
15:      beautiful woman (Naomi Watts), and is subdued enough to be captured
16:      and brought back to New York. However, as the captors and the public
17:      will learn, it takes a lot more shackles to hold back an animal of
18:      such monstrous size.</p>
19:   </body>
20: </html>
```

这段代码与你在本书中见过的 XHTML 网页的代码类似。第一行声明了使用的 XML 版本，第 5 行将该页面与一个 WCSS 样式表关联起来。movie.css 样式表为介绍电影《金刚》移动网页的内容指定了颜色和字体。文档的其他部分很简单，遵循常规 XHTML 语法。例如，文档的主题包含了普通 XHTML 代码，使用了熟悉的<h1>标签和<p>标签。

程序清单 24.2 包含了 movie.css 样式表的代码，它用于格式化介绍电影《金刚》的 XHTML Mobile 网页。

## 程序清单 24.2 介绍电影《金刚》的网页的 WCSS 样式表

```
1: body {
2:   background: #FFFFFF;
3: }
4:
5: h1 {
6:   font-size: x-large;
7:   color: #660000;
8:   text-align: center;
9:   text-decoration: underline;
10: }
11:
12: p {
13:   display: block;
14:   border: 1px #330000 solid;
15:   background: #660000;
16:   color: #FFFFFF;
```



```
17: text-align: left;
18: font-size: medium;
19: padding: 4px;
20: }
```

图 24.2 显示了介绍电影《金刚》的 XHTML Mobile 网页在 Opera 网页浏览器的“小屏幕”视图中显示的结果。

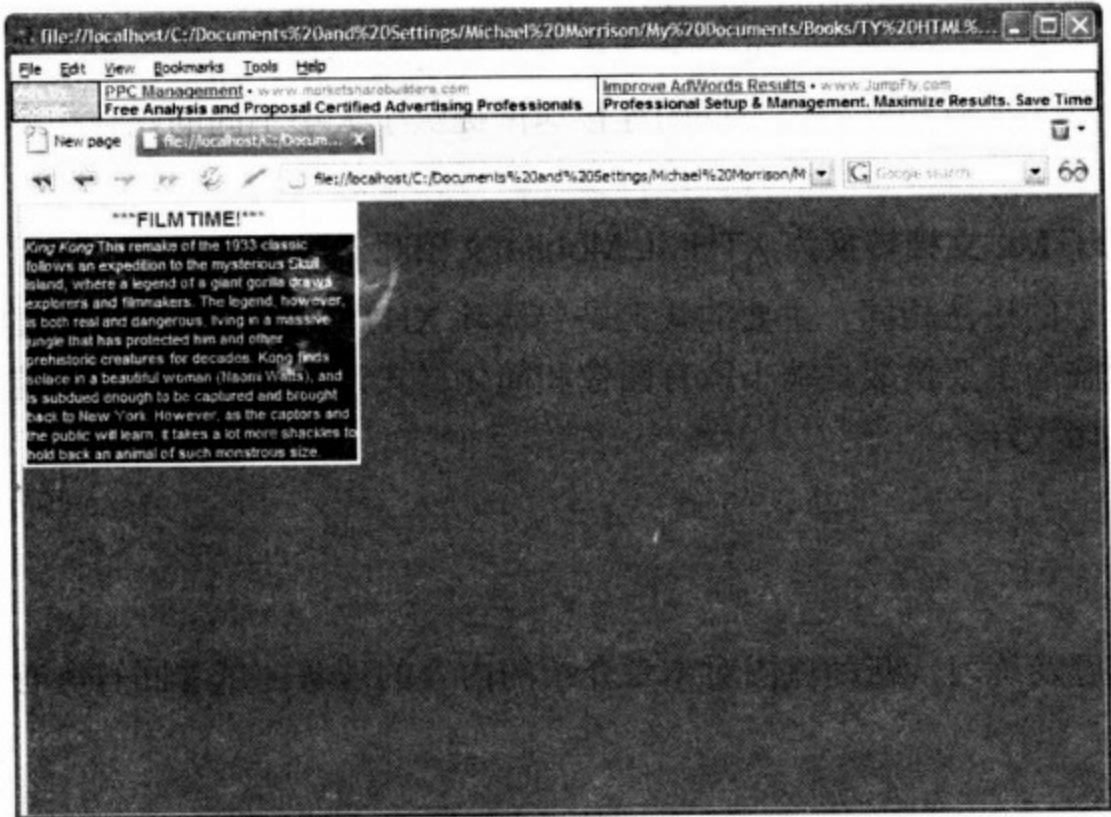


图 24.2  
XHTML Mobile 让你能够创建可在手机和手持计算机等设备上浏览的移动网页

Opera 网页浏览器支持“小屏幕”模式，这一点很独特，它让你能够在桌面浏览器中模拟很小的屏幕。在 Opera 中选择菜单“View/Small Screen”就可以进入小屏幕模式。使用 Opera 的小屏幕模式，无需真正的移动设备就可以测试 XHTML Mobile 网页。然而，最好一使用真正的移动设备来测试 XHTML Mobile 网页，这很重要。

**注意：**Opera 网页浏览器可从 <http://www.opera.com/>免费下载。

**By the Way**

XHTML Mobile 是比较新的技术，然而，随着越来越多的移动设备开始 Web 浏览，它也变得越来越重要。

## 24.7 总结

本章对 HTML 的未来进行了展望，讨论了 HTML 将在全球通信中扮演的新角色，简要地介绍了 HTML 同一些新兴技术的关系，这些新技术基于 HTML 并与 XML 标准紧密相关。本章还给出了一些现在如何规划和组织网页以便适应将来的建议。

### 问与答

问：数字通信与其他通信到底有什么区别？数字通信就是指它使用了 HTML 吗？



**答：**信息是以离散的二进制位（就是数字）的方式传输时，就称为数字方式（digital）。以数字方式传输时，储存、获取与处理信息更容易，而不会丢失或改变信息。计算机的所有信息（包括 HTML）本质上都是数字的，在不远的将来，所有电话、电视、收音机甚至电影制作都将是数字的。许多移动电话网络已经是数字的了，高清晰度电视也是数字的。

**问：**还要多久我才能设计不受使用拨号调制解调器限制的 Internet 网页？

**答：**这取决于你希望什么人阅读你的网页。不幸的是，在将来的若干年内还有许多人使用 56kbit/s 的拨号调制解调器。但好消息是，使用 1Mbit/s（1000kbit/s）或更快的电缆、铜-光纤与无线宽带连接的人数在迅速增加。18 倍的速度之差导致越来越多的网页发布者分别提供高速与低速的网站；许多人已经开始为不同的连接速度提供了大小不同的媒体内容。只要让用户能够绕过视频与音乐等耗费带宽的内容，就没什么可担心的。

**问：**如何将常规 XHTML 文档转换为 XHTML Mobile 文档呢？

**答：**由于这两种语言的语法相同，主要的诀窍是要删除 XHTML Mobile 不支持的特性，如框架。除此之外，面临的主要挑战是缩小所有图像和简化样式，让网页在颜色可能较少的小屏幕上显示的效果也比较好。

## 24.8 作业

作业包含一些测试题及练习，帮助巩固对本章介绍的内容的理解。试着回答所有问题，然后再看后面的答案。

### 测验

你已经在 24 章中做了 23 个测验，这里就不再测验了，建议你恭喜自己出色地完成了任务，休息一下，用一些特别的东西犒劳一下自己。这是你应得的。

## 24.9 练习

现在你已经学习了 HTML、XHTML 和 CSS，且拥有自己的在线网站。本书还可以帮助你将网站做得更好。你可能要复习本书中的问题与答案部分以及附录 A。别忘了在附录 B 的开头夹个书签。做一下你在第一次可能跳过的那些“练习”，将帮助提高 HTML 技能。

相信你还没有探索完 Sams Publishing 网站上所有的娱乐示例与教程提示。其网址为 [www.sampublishing.com](http://www.sampublishing.com)，在那里 JavaScript 非常热门，而 HTML 从不停息。

# 第六部分

## 附录

附录 A 读者最常问的问题

附录 B 完整的 XHTML 1.1 和 CSS 2 快速参考

附录 C 从 HTML 迁移到 XHTML



# 计算机精品学习资料大放送

软考官方指定教材及同步辅导书下载 | 软考历年真题解析与答案

软考视频 | 考试机构 | 考试时间安排

Java 一览无余: Java 视频教程 | Java SE | Java EE

.Net 技术精品资料下载汇总: ASP.NET 篇

.Net 技术精品资料下载汇总: C#语言篇

.Net 技术精品资料下载汇总: VB.NET 篇

撼世出击: C/C++编程语言学习资料尽收眼底 电子书+视频教程

Visual C++(VC/MFC)学习电子书及开发工具下载

Perl/CGI 脚本语言编程学习资源下载地址大全

Python 语言编程学习资料(电子书+视频教程)下载汇总

最新最全 Ruby、Ruby on Rails 精品电子书等学习资料下载

数据库精品学习资源汇总: MySQL 篇 | SQL Server 篇 | Oracle 篇

最强 HTML/xHTML、CSS 精品学习资料下载汇总

最新 JavaScript、Ajax 典藏级学习资料下载分类汇总

网络最强 PHP 开发工具+电子书+视频教程等资料下载汇总

UML 学习电子书下载汇总 软件设计与开发人员必备

经典 LinuxCBT 视频教程系列 Linux 快速学习视频教程一帖通

天罗地网: 精品 Linux 学习资料大收集(电子书+视频教程) Linux 参考资源大系

Linux 系统管理员必备参考资料下载汇总

Linux shell、内核及系统编程精品资料下载汇总

UNIX 操作系统精品学习资料<电子书+视频>分类总汇

FreeBSD/OpenBSD/NetBSD 精品学习资源索引 含书籍+视频

Solaris/OpenSolaris 电子书、视频等精华资料下载索引

## 附录 A

# 读者最常问的问题

我阅读并仔细整理了本书以前版本的读者发给我的 1000 多个问题和建议。这些反馈信息全面影响了本版，从总体布局到具体地注意、提示和测试题。我尽力最大的努力，将读者提出的问题的答案放在各个课程中其他读者可能提出相同问题的地方。

读者可能忘记或忽略了本书中的要点，本附录就是为此而提供的。该附录还回答不适合放在任何主题下的问题，这些问题按照被问及的频率排列：第一个是最常问的问题，第二个其次，以此类推。

如果本书对问题有详细的解释，将指出相关的部分。在读者可能需要更多信息的情况下，将提供简洁的答案，并提供可能会有所帮助的在线资源。

## 最常见的 24 个问题

### 1. 下一步应阅读什么书？

尝试阅读《JavaScript 基础、案例与应用》（已由人民邮电出版社翻译出版）。

### 2. 我在创建第一个网页时遇到了困难，它不管用。我做错了什么？

第一个网页总是最难的。如果查看网页（在网页浏览器中选择菜单“文件/打开”）时看到的是 HTML 代码或网页开头看到怪异的字符，表明没有将文件保存为纯文本或 ASCII 文本格式。如果不知道如何在字处理器存储为这种格式，使用计算机自带的记事本或 SimpleText 编辑器（Wordpad 在这方面最容易出问题）。另外，确保保存文件时使用了扩展名.htm 或.html。

有关创建第一个网页的更多指导，请细心阅读第 2 章开头的“练习”和 2.1 节。

另外，要编辑和浏览硬盘上的网页，不必连接到 Internet。如果每当启动网页浏览器时它都试图连接到 Internet，可将浏览器的主页设置为硬盘中的一个网页。

### 3. 图像或多媒体文件不能在线显示。

将图像放到网页中时，可能遇到几个常见的陷阱。

- 确保图像文件与引用它的 HTML 文档在相同的文件夹中；如果要试图引用其他文件夹中的图像，复习第 3 章中的“指定网页的地址”一节。
- 确保图像文件存储为 GIF、JPEG 或 PNG 格式。使用 Paint Shop Pro、Picasa 或其他图像软件打开文件，并使用菜单“File/Save As”重新保存。
- 确保文件名的大小写和<img />标签中的“src=属性”匹配。对于对于大多数 Web 服务器来说，MyImage.gif 和 myimage.GIF 是不同的！
- 为删除图像周围的蓝色边框，在<img />标签中加入 style="border:none"。
- 下面这种情况虽然有可能出现，但可能性不大：在 Mozilla FireFox 中，是否在 Tools/Options/Web Features 下关闭 Load Images；在 Microsoft Internet Explorer 中，是否在“工具/Internet 选项/高级”中关闭了“显示图片”。
- 真实的故事：一个读者花了 4 天时间试图弄清楚图像为什么不显示，最后发现，在应输入“<img src=”的每个地方都输入了“<img scr=”。不要笑，务必检查输入的代码是否正确。

参考第 8 章。如果在网页上放置图像时遇到麻烦，不仅在第 8 章可以找到有用的提示，在第 9 和第 10 章也可以找到。

音频和视频文件比较棘手，更容易出问题。无法使它们在每种流行浏览器的每个版本都能够正常运行，但可参阅第 19 章获得尽可能多的帮助。

#### 4. 如何使表单能够在服务器上正常运行？

请求 ISP 帮助设置表单处理脚本。如果 ISP 不能这样做，需要找到一个愿意提供这种服务的 ISP，或使用第三方的表单处理服务，如 <http://www.freedback.com>。也可以将表单信息发送到你的电子邮件地址，但这种方法不那么可靠，因为它要求所有用户都通过网页浏览器正确地配置其电子邮件。

#### 5. 如何在网页中放置计数器？

你大概不需要计数器，因为大多数 ISP 每周都给您发送详细的统计报告，总结了你的每个网页被访问多少次。你应该要求提供这种服务，但有些 Web 托管服务商（尤其是免费的服务或北美以外地区的服务商）不提供这种报告。在这样的情况下，需要在服务器上建立一个 CGI 脚本。这不太难，在 <http://www.developer.com> 和其他 Web 开发人员网站上可以找到有关如何完成这项工作的代码和建议。

#### 6. 我对框架不太明白。我的框架不管用，我不知道原因。你知道吗？

框架比较棘手。可能需要阅读第 16 章两三篇并做些练习，才能全面理解它。下面的提示可能会有所帮助。

- 可以在任何框架中右击，并选择“This Frame/View Frame Source”（Mozilla FireFox）或“查看源文件”（Microsoft Internet Explorer），以查看框架的 HTML 代码。从主菜单中选择“查看/源文件”将显示框架集文档的 HTML 代码。
- 使链接同时修改多个框架内容的惟一方法是，链接到新的框架集，并在链接标签<a>中包含 target="\_top"。



➤ 要删除所有框架回到常规单页文档时，也使用 `target="_top"`。

7. 如何在网页设计领域谋职，为他人制作一个网页应收费多少呢？

如同在任何竞争性行业（网页设计是竞争激烈的行业）一样，需要详实的营销计划才能成功。如果已经找到一些客户，收费多少显然要靠谈判来确定。通常，经验丰富的 Web 开发人员的收费标准在每章 25~50 美元之间。如果你还处于学习阶段，收费将更低些，除非你是专业图像或出版设计人员，拥有忠实的客户群。

8. 如何制作用密码保护的网页？

向 ISP 询问它们可提供哪些安全措施。要实现真正的密码保护，需要使用安全网页或至少某种脚本，限制只有输入了有效密码的用户才能访问。

9. 可以在哪里查找 Java 小程序和编写好的 JavaScript？

查看 <http://www.developer.com>、<http://www.javascript.com> 和 <http://javascript.internet.com>；或在任何主要的 Internet 搜索网站使用关键字 Java 或 JavaScript 进行搜索。

10. 链接无法正常运行，可能是什么地方有问题？

检查 href 链接及其链接到的文件的拼写和大小写。如果大小写不匹配，有些链接在硬盘上能够正常运行，但在 Web 服务器上不行，这是因为 Windows 不区分文件名的大小写，但 UNIX 区分。另外，复习第 3 章和第 6 章，确保理解了相对地址、绝对地址和链接锚点。

11. 使用 JavaScript 代码时遇到了麻烦，尽管语法绝对没有问题。

绝对不要认为 Microsoft 和 Netscape 的网页浏览器存在问题，尤其是 JavaScript 模块。有问题的是你。

可以考虑使用 JavaScript 调试器用自己的智慧解决遇到的 JavaScript 问题。Mozilla 提供了一个名为 Venkman 的免费调试器，它被集成到 Firefox 中，更详细的信息请参阅 <http://www.mozilla.org/project/venkman/>。Internet Explorer 也内置了一个脚本调试器，可以在“Internet 选项”对话框的“高级”选项卡中，取消选中复选框“禁用脚本调试(Internet Explorer)”来启用它。

12. 可在哪里得到有关创建图像和多媒体的更多帮助？

如果使用 Paint Shop Pro 处理图像，可参阅 <http://mardiweb.com/web/> 处的在线教程；如果使用的是 Picasa，Google 在其 Picasa 网站的支持部分 (<http://picasa.google.com/support/>) 提供了“Getting Startd Guide”。

13. 如何把更多的内容放在网站中（聊天室、点击计数器、密码保护、交互声音、下拉链接列表等）？

这些都涉及 JavaScript 或 CGI 脚本编程（高级技术）。为帮助读者学习本书没有介绍的知识，在 <http://www.sampublishing.com> 提供了一个高级开发人员资源的列表。

14. 如何使一条消息在网页底部滚动？

在 <http://www.developer.com> 和 <http://www.javascript.com> 都能够找到这样的 JavaScript 脚本。

15. 如何将文件放在 Web 网站上供人下载？

只需将文件上传到放置网页的地方，并使用一个常规 HTML 连接，像下面这样：

```
<a href="bigfile.zip">Click here to download bigfile.zip.</a>
```

以这种方式链接图像时，单击链接可能在浏览器窗口中打开图像，这不是你将图像放到网上供人下载的目的。对此，解决方案是用户在浏览器中打开图像后，将其存储到本地计算机中。

16. 如何将浏览器放在存储卡或光盘中？通过便携介质发布网页时，需要这样做吗？

当前，大多数人的计算机上都有网页浏览器，但如果想提供网页浏览器以防万一，需要获得浏览器公司的许可。我推荐 Opera (<http://www.opera.com>)，它很小且允许免费分发有时间限制的评估版。

17. 下载您推荐的 Paint Shop Pro 或 FTP 软件时，下载非常慢甚至完全停止。您能帮忙吗？

除建议以后再试外，我恐怕帮不上其他忙。可以想象一下交通阻塞时的汽车。

18. 应使用 Java 小程序和其他高级功能吗？

除非需要做一些采用其他方法都无法完成的事情，否则不应使用。基本 HTML 代码的速度更快、兼容性更高且更容易维护。

19. 如何制作表单让人填写并打印？

可以填写并打印任何 HTML 表单，只需告诉他们可以这样做，请参阅 18.10 节。确保表单元素位于标签 `<form>` 内，即使你不打算以电子方式提交表单。

20. 如何调整文本使其两端对齐？

如果使用的是平板显示器，可尝试使用剪刀和胶水（译者注：这是作者在开玩笑）。也可以像下面这样使用 CSS 样式表 `text-align: style="text-align:justify"`。

21. 如何为网站做宣传？如何为网站找到广告商？

23.1 节会有所帮助，但多半需要根据具体情况制定自己的营销/PR 计划。

有很多 Web 广告服务和公司，它们向独立 Web 发布者付费以播放广告或以其他可能盈利的方式与独立 Web 发布者合作。每当访问者单击它们的广告时，付给你少量的钱（详情请参阅 <http://www.sitecash.com/guide.htm>）。一种更好的选择是 Google AdSense，它是 Google 开发的一个项目，在网站中显示相关的广告，每当有人单击广告，你都将得到一定的收入。更详细的信息，请参阅 Google AdSense 网站，其网址为 <http://www.google.com/adsense>。

22. 如何在电子邮件内创建 HTML 网页或链接？

只需像创建网页那样输入 HTML 代码。当今的大多数高级电子邮件程序都允许以 HTML 方式创建和查看电子邮件。就像格式化文档那样进行格式化，不需要任何 HTML 经验。通常，如果电子邮件的 Web 地址以 `http://` 打头，大多数电子邮件程序自动将其转换为链接。

23. 如何在新窗口中打开链接？

在 href 链接标签中使用 `target="_blank"`。从技术上说，这种方法与 XHTML 1.1 不兼容。

为避开与 XHTML 1.1 不兼容的问题，请阅读 3.4 节。

#### 24. 如何链接到数据库让访问者能够搜索我的网站？

对于这个问题，要获得良好的解决方案，可能需要向软件公司付点钱。你的 Web 托管服务商可能已经支持某种特定的数据库，能够向你提供有关创建能够访问数据库的网页的指南。你可能想学习 PHP 和 MySQL，这是两种让你能够在网站中添加数据库功能的流行技术。



## 附录 B

# 完整的 XHTML 1.1 和 CSS 2 快速参考

作为 XML 的一种应用，XHTML 1.1 是对 HTML 的改进，使得更易于定义和实现对语言的扩展。本附录提供所有 XHTML 1.1 元素和属性以及 CSS 2 样式属性的快速参考。

为方便读者查找信息，本附录根据功能将 HTML 元素放在不同的小节中，并按如下顺序排列各个小节：

- 结构；
- 文本短语和段落；
- 文本格式化元素；
- 列表；
- 链接；
- 表格；
- 框架；
- 嵌入内容；
- 样式；
- 表单；
- 脚本。

在每节中，元素按字母顺序排列，并提供如下信息。

- 用法：元素的总体描述。
- 开始/结束标签：指出标签是否是必需的、可选还是非法。
- 属性：列出元素的属性，并简要说明其作用。
- 是否可为空：指出元素是否可以为空。
- 备注：使用元素时需要特别考虑的因素。

CSS 样式属性以类似的方式排列，只是其中包含的是可接受的值，而不是属性。

**注意：**XHTML 1.1 包含几个基本属性，可用于很多元素中。在每个元素中，这些属性被称为属性组 core、i18n 和 events。列出所有 XHTML 元素后，将详细说明这些属性组，指出每个属性组包含哪些属性。

By the  
Way

## B.1 XHTML 结构

XHTML 依靠几个元素来定义文档结构（与结构化文档内的文本相对），并提供给浏览器或搜索引擎使用的信息。

### <bdo>...</bdo>

用法	用于有选择性地关闭默认文本方向的双向算法元素
开始/结束标签	必需/必需
属性	core ang="language": 文档的语言 dir="direction": 文本方向 (ltr、rtl)，强制属性
是否可为空	否

### <body>...</body>

用法	包含文档的内容
开始/结束标签	可选/可选
属性	core、i18n 和 events onload="eventcode": 文档加载时触发的内部事件 onunload="eventcode ": 文档卸载时触发的内部事件
是否可为空	否
备注	只能有一个<body>，必须紧跟在<head>的后面，<body>元素可由<frameset>元素代替

### 注释<!--...-->

用法	用于插入注释或脚本，浏览器不显示它们
开始/结束标签	必需/必需
属性	没有
是否可为空	是
备注	注释在不限于一行，而是可以为任意长。结束标签不必与开始标签位于同一行

### <div>...</div>

用法	用于给文本块添加结构
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	不能在 p 元素内使用

**注意：**你可能遇到这样的 HTML 网页，即使用带 align 属性的<div>元素。在 XHTML 中，该属性已被删除，并使用 CSS 样式属性 text-align 来对齐文本。该样式属性将在本附录后面介绍。

By the  
Way



<!doctype>

用法	版本信息出现在 HTML 文档的第一行，它是 SGML 声明而不是元素
----	-------------------------------------

<h1>...</h1>到<h6>...</h6>

用法	6 种标题（h1 最重要的），用于在 body 中以层次方式组织信息
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	浏览器根据标题的重要程度使用不同的字号，<h1>最大，而<h6>最小

<head>...</head>

用法	这是文档头，包含向用户和搜索引擎提供信息的其他元素
开始/结束标签	可选/可选
属性	i18n profile="url": 指定元数据位置的 URL
是否可为空	否
备注	每个文档只能有一个<head>，必须在开始标签<html>后面，且在<body>的前面

<hr />

用法	用于分隔网页各部分的水平线
开始/结束标签	必需/非法
属性	core、i18n 和 events
是否可为空	是

<html>...</html>

用法	html 元素包含整个文档
开始/结束标签	可选/可选
属性	i18n
是否可为空	否
备注	声明<!doctype...>中包含版本信息，因此在这里不是必需的

<meta />

用法	提供有关文档的信息
开始/结束标签	必需/非法
属性	i18n http-equiv="servercmd": HTTP 响应报头的名称 name="name": 元信息的名称 contemt="value": 元信息的内容 scheme="scheme": 指定用于解释元数据的模式
是否可为空	是

<span>...</span>

用法	通过定义文本跨度来组织文档
----	---------------

开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
<b>&lt;title&gt;...&lt;/title&gt;</b>	
用法	给网页指定的名称。<title>标签位于在<head>标签中，其内容显示在浏览器窗口的标题栏中
开始/结束标签	必需/必需
属性	i18n
是否可为空	否
备注	每个文档只能有一个标题

## B.2 XHTML 文本短语和段落

可根据特定的用途结构化文本短语（块），如创建段落。不应将其同修改文本格式混为一谈。

<b>&lt;abbr&gt;...&lt;/abbr&gt;</b>	
用法	用于定义缩写
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	该标签中包含的是缩写格式，完整的内容有其属性定义。
<b>&lt;acronym&gt;...&lt;/acronym&gt;</b>	
用法	定义首字母缩略词
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
<b>&lt;address&gt;...&lt;/address&gt;</b>	
用法	为作者或联系人信息指定特殊格式
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	通常在<address>元素中使用 元素将地址分行
<b>&lt;blockquote&gt;...&lt;/blockquote&gt;</b>	
用法	用于显示长引用
开始/结束标签	必需/必需
属性	core、i18n 和 events cite="url": 要引用的文本的地址
是否可为空	否

**<br />**

用法	强行换行
开始/结束标签	必需/非法
属性	core、i18n 和 events
是否可为空	是

**<cite>...</cite>**

用法	引用参考
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<code>...</code>**

用法	标识要显示的代码段
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<del>...</del>**

用法	显示最后一次修改从文档中删除的文本
开始/结束标签	必需/必需
属性	core、i18n 和 events cite="url": 链接的文档的 URL, 该文档解释了所做的修改 datetime="datetime": 指出修改的日期和时间
是否可为空	否

**<dfn>...</dfn>**

用法	定义包括的项
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<h1>...</h1>到<h6>...</h6>**

用法	文本标题
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<ins>...</ins>**

用法	显示最后一次修改在文档中插入的文本
开始/结束标签	必需/必需
属性	core、i18n 和 events cite="url": 链接的文档的 URL, 该文档解释了所做的修改

	datetime="datetime": 指出修改的日期和时间
是否可为空	否

<kbd>...</kbd>

用法	指出用户将输入的文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<p>...</p>

用法	定义段落
开始/结束标签	必需/可选
属性	core、i18n 和 events
是否可为空	否

<pre>...</pre>

用法	显示预先格式化的文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<q>...</q>

用法	用于显示不需要分段的短引用
开始/结束标签	必需/必需
属性	core、i18n 和 events cite="url": 引用文本的 URL
是否可为空	否

<samp>...</samp>

用法	标识示例输出
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<strong>...</strong>

用法	更强的强调
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<sub>...</sub>

用法	创建下标
开始/结束标签	必需/必需

属性	core、i18n 和 events
是否可为空	否

<sup>...</sup>

用法	创建上标
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<var>...</var>

用法	一个变量
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

### B.3 XHTML 文本格式化元素

文本特征如字号、粗细和样式可使用这些元素来修改，但更好的方法是使用 CSS 样式属性。本附录后面包含有关这些属性的完整参考，这些属性让你能够对文本进行难以置信的控制。

<b>...</b>

用法	粗体文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<big>...</big>

用法	大文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<i>...</i>

用法	斜体文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<small>...</small>

用法	小文本
开始/结束标签	必需/必需
属性	core、i18n 和 events



是否可为空	否
<b>&lt;tt&gt;...&lt;/tt&gt;</b>	
用法	电传打字机（等宽）文本
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

## B.4 XHTML 列表

通过创建列表，可以将文本组织成结构化程度更高的大纲。列表可嵌套。

<b>&lt;dd&gt;...&lt;/dd&gt;</b>	
用法	在<dl>（定义列表）元素中使用的定义描述
开始/结束标签	必需/可选
属性	core、i18n 和 events
是否可为空	否
备注	可包含块级的内容，如<p>元素

<b>&lt;dl&gt;...&lt;/dl&gt;</b>	
用法	创建定义列表
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	必须包含至少一个<dt>或<dd>元素

<b>&lt;dt&gt;...&lt;/dt&gt;</b>	
用法	<dl>（定义列表）中使用的定义项
开始/结束标签	必需/可选
属性	core、i18n 和 events
是否可为空	否
备注	必须包含文本（文本用文本标记元素进行修改）

<b>&lt;li&gt;...&lt;/li&gt;</b>	
用法	定义列表中的列表项
开始/结束标签	必需/可选
属性	core、i18n 和 events
是否可为空	否

<b>&lt;ol&gt;...&lt;/ol&gt;</b>	
用法	创建有序列表
开始/结束标签	必需/必需
属性	core、i18n 和 events

是否可为空	否
备注	必须包含至少一个列表项

<ul>...</ul>

用法	创建无序列表
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	必须包含至少一个列表项

B.5 XHTML 链接

超级链接是 XHTML 的基础。这些元素让你能够链接到其他文档、当前文档的其他地方或外部文件。

<a>...</a>

用法	用于定义链接和锚
开始/结束标签	必需/必需
属性	core、i18n 和 events charset="encoding": 源的字符编码 name="name": 定义锚 href="linkurl": 链接源的 URL rel="linktype": 前向链接类型 rev="linktype": 逆向连接类型 shape="value": 让你能够使用定义的形状 (default、rect、circle、poly) 定义客户端图像映射 coords="values": 使用像素或百分比设置形状的大小
是否可为空	否

<base />

用法	文档中的其他所有 URL 都相对于这个位置进行解析
开始/结束标签	必需/非法
属性	href="linkurl": 链接源的 URL
是否可为空	是
备注	位于文档的<head>中

<link/>

用法	定义链接和源之间的关系
开始/结束标签	必需/非法
属性	core、i18n 和 events charset="encoding": 源的字符编码 href="linkurl": 源的 URL rel="linktype": 前向链接类型 rev="linktype": 逆向连接类型

	type="contenttype": Internet 内容类型
	media="media": 定义目标介质 (screen、print、projection、braille、speech、all)
	target="placement": 指定在哪里显示源 (用户定义的名称、blank、parent、self、top)
是否可为空	是
备注	位于文档的<head>中

## B.6 XHTML 表格

表格是一种以表格方式显示数据的方法。在 XHTML 之前, 表格被广泛地用于页面布局, 但随着样式表的出现, W3C 不再鼓励这样使用表格。

### <caption>...</caption>

用法	显示表题
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	可选的

### <col />

用法	为共享属性值将列编成组
开始/结束标签	必需/非法
属性	core、i18n 和 events span="numcols": 组包含的列数 width="width": 列宽, 可以是百分比、像素数或最小值 align="alignment": 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bootom、baseline)
是否可为空	是

### <colgroup>...</colgroup>

用法	定义一个列组
开始/结束标签	必需/非法
属性	core、i18n 和 events span="numcols": 组中包含的列数 width="width": 列宽 align="alignment": 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bootom、baseline)
是否可为空	否

### <table>...</table>

用法	创建表格
开始/结束标签	必需/必需

属性	core、i18n 和 events width="width": 表宽 cols="numcols": 列数 border="borderwidth": 表格边框的宽度 (单位为像素) frame="frame": 设置表格边框的可视性 (void、above、below、hsides、lhs、rhs、vsides、box、border) rules="rules": 设置表格内的可视规则 (none、group、rows、cols、all) cellspacing="cellspacing": 单元格之间的间距 cellpadding="cellpadding": 单元格内的间距 summary="description": 提供表格的文本描述, 以方便使用
是否可为空	否
<b>&lt;tbody&gt;...&lt;/tbody&gt;</b>	
用法	定义表体
开始/结束标签	可选/可选
属性	core、i18n 和 events align="alignment": 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline)
是否可为空	否
<b>&lt;td&gt;...&lt;/td&gt;</b>	
用法	定义单元格的内容
开始/结束标签	必需/可选
属性	core、i18n 和 events abbr="name": 缩写的名称 axis="axisnames": 列出单元格对应的行头和列头 rowspan="numrows": 单元格跨越的行数 colspan="numcols": 单元格跨越的列数 align="alignment": 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline) headers="headers": 单元格的表头信息 scope="scope": 指出单元格是否为其他单元格提供表头信息
是否可为空	否
<b>&lt;tfoot&gt;...&lt;/tfoot&gt;</b>	
用法	定义表格脚注
开始/结束标签	必需/可选
属性	core、i18n 和 events align="alignment" 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline)
是否可为空	否

<th>...</th>

用法	定义表头单元格的内容
开始/结束标签	必需/可选
属性	core、i18n 和 events abbr="name": 缩写的名称 axis="axisnames": 列出单元格对应的行头和列头 rowspan="numrows": 单元格跨越的行数 colspan="numcols": 单元格跨越的列数 align="alignment": 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline) headers="headers": 单元格的表头信息 scope="scope": 指出单元格是否为其他单元格提供表头信息
是否可为空	否

<thead>...</thead>

用法	定义表头
开始/结束标签	必需/可选
属性	core、i18n 和 events align="alignment" 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline)
是否可为空	否

<tr>...</tr>

用法	定义表格中一行单元格
开始/结束标签	必需/可选
属性	core、i18n 和 events align="alignment" 水平对齐单元格的内容 (left、center、right、justify、char) char="charalignment": 设置列与之对齐的字符 charoff="charoffset": 一行中到第一个对齐字符的偏移量 valign="verticalalignment": 垂直对齐单元格的内容 (top、middle、bottom、baseline)
是否可为空	否

B.7 XHTML 框架

框架用于在网页浏览器窗口中创建“面板”，用于显示来自不同源文档的内容。框架让你能够创建有多个其他网页组成的网页。XHTML 1.1 不支持框架，但仍可以通过 XHTML 1.0 Frameset DTD 和 XHTML 1.0 Transitional DTD 使用框架。

<frame />

用法	定义框架
开始/结束标签	必需/非法



属性	core name="name": 框架的名称 src="sourceurl": 要显示在框架中的源文档 frameboder="border": 显示/隐藏框架的边框 (0、1) marginwidth="width": 设置框架边框和内容之间的间隔 marginheight="height": 设置框架边框和内容之间的间隔 noresize="noresize": 禁止调整大小 scrolling="scrolling": 指定滚动条的出现方式 (auto、yes、no) longdesc="descurl": 指向框架描述的 URL, 用于不支持框架的浏览器
是否可为空	是
备注	XHTML 1.0 Frameset DTD

<frameset>...</frameset>

用法	定义窗口内的框架布局
开始/结束标签	必需/必需
属性	core rows="numrows": 包含的框架行数 cols="numcols": 包含的框架列数 onload="eventcode": 文档加载时触发的内部事件 onunload="eventcode": 文档卸载时触发的内部事件
是否可为空	否
备注	XHTML 1.0 Frameset DTD: 框架集可嵌套

<iframe>...</iframe>

用法	创建内部框架
开始/结束标签	必需/必需
属性	core name="name": 框架的名称 src="sourceurl": 要显示在框架中的源文档 frameboder="border": 显示/隐藏框架的边框 (0、1) marginwidth="width": 设置框架边框和内容之间的间隔 marginheight="height": 设置框架边框和内容之间的间隔 scrolling="scrolling": 指定滚动条的出现方式 (auto、yes、no) height="height": 高度 width="width": 宽度 longdesc="descurl": 指向框架描述的 URL, 用于不支持框架的浏览器
是否可为空	否
备注	XHTML 1.0 Transitional DTD

<noframes>...</noframes>

用法	不支持框架时的替代内容
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否
备注	XHTML 1.0 Frameset DTD

## B.8 XHTML 嵌入内容

也被称为包含 (inclusion)，嵌入内容适用于图像、图像映射、Java 小程序、Flash 动画以及其他多媒体或编程内容，它们放在网页中提供额外功能。

### <area />

用法	用于定义链接和锚
开始/结束标签	必需/非法
属性	core、i18n 和 events shape="value": 让你能够使用定义的形状 (default、rect、circle、poly) 定义客户端图像映射 coords="values": 使用像素或百分比设置形状的大小 href="linkurl": 链接源文档的 URL nohref="nohref": 表示区域没有动作 alt="alttex": 替代文本 onfocus="eventcode": 元素获得焦点时发生的事件 onblur="eventcode": 元素失去焦点时发生的事件
是否可为空	是

### <img />

用法	在文档中包含一幅图像
开始/结束标签	必需/非法
属性	core、i18n 和 events src="sourceurl": 图像的 URL alt="alttext": 替代文本 width="height": 图像的宽度 height="width": 图像的高度 border="border": 边框宽度 hspace="horizontal space": 图像同其他内容的水平间距 vspace="vertical space": 图像同其他内容的垂直间距 usemap="mapurl": 指向客户端图像映射的 URL ismap="ismap": 指定服务器端图像映射 longdesc="desurl": 指向图像描述的 URL，供不显示图像的浏览器使用
是否可为空	是

### <map>...</map>

用法	与<area>元素一起使用时，创建客户端图像映射
开始/结束标签	必需/必需
属性	core、i18n 和 events name="name": 要创建的图像映射的名称
是否可为空	否

### <object>...</object>

用法	包含一个对象
开始/结束标签	必需/必需

属性	core、i18n 和 events declare="declare": 声明但不创建对象的标记 classid="objecturl": 对象的 URL codebase="codebaseurl": 用于解析其他属性指定的 URL 的 URL data="dataurl": 对象数据的 URL type="datatype": 数据的 Internet 内容类型 codetype="codetype": 代码的 Internet 内容类型 standby="waitmsg": 加载时显示消息 height="height": 图像的高度 width="width": 图像的宽度 border="border": 在对象周围显示边框 hspace="horizontal space": 对象边界同其他其他页面内容之间的间距 vspace="vertical space": 对象的顶部和底部同其他页面内容之间的间距 usemap="mapurl": 图像映射的 URL shape="shapes": 让你能够定义搜索超级链接的区域, 如果对象是图像 name="nameurl": 作为表单的一部分提交的 URL
是否可为空	否

<param />

用法	初始化对象
开始/结束标签	必需/非法
属性	neme="name": 定义参数名 value="value": 对象参数的值 valuetype="valuetype": 定义值类型 (data、ref、object) type="contenttype": Internet 介质类型
是否可为空	是

B.9 XHTML 样式

通过使用<style>元素, 可以将样式表 (内联和外部的) 加入到 HTML 文档中。

<style>...</style>

用法	创建内部样式表
开始/结束标签	必需/必需
属性	i18n type="contenttype": Internet 内容类型 media="media": 定义目标介质 (screen、print、projection、braille、speech、all) title="title": 样式的名称
是否可为空	否
备注	位于<head>元素中

B.10 XHTML 表单

表单创建让用户能够选择选项、输入信息并将数据发送给 Web 服务器进行处理的界面。

<button>...</button>

用法	创建按钮
开始/结束标签	必需/必需
属性	core、i18n 和 events name="name": 按钮名 value="value": 按钮的值 type="type": 按钮类型 (button、submit、reset) disabled="disabled": 将按钮状态设置为禁用 onfocus="eventcode": 元素获得焦点时发生的事件 onblur="eventcode": 当元素失去焦点时发生的事件
是否可为空	否

<fieldset>...</fieldset>

用法	将相关控件编组
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

<form>...</form>

用法	创建一个表单用于放置用户输入控件
开始/结束标签	必需/必需
属性	core、i18n 和 events action="actionurl": 服务器动作的 URL method="post/get": HTML 方法 (get、post)，不提倡使用 get enctype="mediatype": 指定 MLME (Internet 媒体类型) onsubmit="eventcode": 提交表单时发生的内部事件 onreset="eventcode": 当重置表单时发生的内部事件 accept="contenttype": 服务器能够接受的内容类型列表 accept-charset="encodings": 字符编码列表
是否可为空	否

<input/>

用法	定义在表单中使用的控件
开始/结束标签	必需/非法
属性	core、i18n 和 events type="controltype": 输入控件的类型 (txet、password、checkbox、radio、submit、reset、file、hidden、image、button) name="name": 控件的名称 (除 submit 和 reset 外都需要) value="value": 控件的初始值 (单选按钮和复选框都需要) checked="checked": 将单选按钮的状态设置为被选中 disabled="disabled": 禁用控件 readonly="readonly": 适用于文本密码类型 size="size": 除文本和密码控件外其他控件的宽度 (像素数)，文本和密码控件的宽度用字符数指定

	maxlength="maxlength": 可输入的最大字符数
	src="imageurl": 图像控件类型的 URL
	alt="alttex": 替代文本描述
	usemap="mapurl": 到客户端图像映射的 URL
	onfocus="eventcode": 元素获得焦点时发生的事件
	onblur="eventcode": 元素失去焦点时发生的事件
	onselect="eventcode": 控件被选中时发生的内部事件
	onchange="eventcode": 控件的内容变化时发生的内部事件
	accept="filetypes": 允许上传的文件类型
是否可为空	是

**<label>...</label>**

用法	给控件加上标签
开始/结束标签	必需/必需
属性	core、i18n 和 events
	for="control": 将控件与标签关联起来
	onfocus="eventcode": 元素获得焦点时发生的事件
	onblur="eventcode": 元素失去焦点时发生的事件
是否可为空	否

**<legend>...</legend>**

用法	为 fieldset 指定标题
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<optgroup>...</optgroup>**

用法	用于将<select>元素中的表单元素编组
开始/结束标签	必需/必需
属性	core、i18n 和 events
	disabled="disabled": 不使用
	label="label": 定义分组的标签
是否可为空	否

**<option>...</option>**

用法	指定<select>元素中的选项
开始/结束标签	必需/可选
属性	core、i18n 和 events
	selected="selected": 指定选项是否被选中
	disabled="disabled": 禁用控件
	label="label": 为选项组定义标签
	value="value": 控件提交时的值
是否可为空	否



**<select>...</select>**

用法	创建一些选项供用户选择
开始/结束标签	必需/必需
属性	core、i18n 和 events name="name": 元素的名称 size="size": 以行为单位的宽度 multiple="multiple": 允许多项选择 disabled="disabled": 禁用控件 onfocus="eventcode": 元素获得焦点时发生的事件 onblur="eventcode": 元素失去焦点时发生的事件 onselect="eventcode": 控件被选择时发生的内部事件 onchange="eventcode": 控件内容发生变化时发生的内部事件
是否可为空	否

**<textarea>...</textarea>**

用法	创建一个区域供用户输入多行文本
开始/结束标签	必需/必需
属性	core、i18n 和 events name="name": 元素的名称 rows="numrowss": 以行为单位的宽度 cols="numcols": 以列为单位的高度 disabled="disabled": 禁用控件 readonly="readonly": 设置显示的文本为只读状态 onfocus="eventcode": 元素获得焦点时发生的事件 onblur="eventcode": 元素失去焦点时发生的事件 onselect="eventcode": 控件被选中时发生的内部事件 onchange="eventcode": 控件内容改变时发生的内部事件
是否可为空	否
备注	将要显示的文本放在开始和结束标签之间

B.11 XHTML 脚本

使用脚本可以处理数据和执行其他动态事件。使用<script>元素将脚本包含的网页中，该元素还指出了使用的脚本语言（JavaScript、VBScript 等）。

**<noscript>...</noscript>**

用法	提供浏览器不能执行脚本时使用的替代内容
开始/结束标签	必需/必需
属性	core、i18n 和 events
是否可为空	否

**<script>...</script>**

用法	<script>元素包含由浏览器执行的客户端脚本
开始/结束标签	必需/必需

属性	<code>type="scripttype"</code> : 脚本语言的 Internet 内容类型 <code>src="scripturl"</code> : 外部脚本的 URL <code>defer="defer"</code> : 指出脚本不修改文档内容
是否可为空	否
备注	可在<meta/>元素中设置默认的脚本语言

## B.12 XHTML 常用属性和事件

在前面几节中，下面 6 个属性被统称为 core。

- `id="id"`: 全局标识符。
- `class="styleclasses"`: 由空格分隔的类列表。
- `style="styles"`: 样式信息。
- `title="title"`: 为特定元素提供其他信息；而<title>元素指定整个网页的标题。
- `accesskey="shortcut"`: 设置用于访问元素的快捷键。
- `tabindex="taborder"`: 设置元素的 Tab 顺序。

在前几节，下面两个用于国际化 (i18n) 的属性统称为 i18n。

- `lang="lang"`: 语言标识符
- `dir="textdir"`: 文本方向 (ltr,rtl)

下列内部事件统称为 events。

- `OnClick="eventcode"`: 指针设备（如鼠标）被单击。
- `OnDblClick=" eventcode"`: 指针设备（如鼠标）被双击。
- `OnMouseDown=" eventcode"`: 按住鼠标按钮。
- `OnMouseUp=" eventcode"`: 松开被按住的鼠标按钮。
- `OnMouseOver=" eventcode"`: 鼠标移到对象上。
- `OnMouseMove=" eventcode"`: 鼠标移动。
- `OnMouseOut=" eventcode"`: 鼠标离开对象。
- `OnKeyPress=" eventcode"`: 按下键然后松开。
- `OnKeyDown=" eventcode"`: 键被按住。
- `OnKeyUp=" eventcode"`: 按住的键被松开。

## B.13 CSS 尺寸样式属性

很多 CSS 样式规则依赖于一个或另一个表单的尺寸属性。如果不使用这些尺寸属性，将难以指定表单中元素的尺寸。

### height

用法	设置元素的高度
----	---------

值	auto、length、%
<b>line-height</b>	
用法	设置元素行之间的距离
值	normal、length、%
<b>max-height</b>	
用法	设置元素的最大高度
值	none、length、%
<b>max-width</b>	
用法	设置元素的最大宽度
值	none、length、%
<b>min-height</b>	
用法	设置元素的最小高度
值	length、%
<b>min-width</b>	
用法	设置元素的最小宽度
值	length、%
<b>width</b>	
用法	设置元素的宽度
值	auto、length、%

## B.14 CSS 文本和字体样式属性

CCS 的核心是文本和字体样式属性，它们让你能够以难以置信的方式控制网页文本的外观。

<b>color</b>	
用法	设置文本的颜色
值	color
<b>direction</b>	
用法	设置文本的方向，从左到右还是从右到左
值	ltr、rtl
<b>font</b>	
用法	让你能够在一个地方设置所有字体属性
值	font-style、font-variant、font-weight、font-size/line-height、font-family

font-family

用法	用于元素的顺序字体名和/或通用字体名列表
值	family-name、generic-family

font-size

用法	设置字体大小
值	xx-small、x-small、small、medium、large、x-large、xx-large、smaller、larger、length、%

font-style

用法	设置字体样式
值	family-name、generic-family

font-variant

用法	以小型大写字体或常规字体显示文本
值	normal、small-caps

font-weight

用法	设置字体的粗细
值	normal、bold、bolder、lighter、100、200、300、400、500、600、700、800、900

letter-spacing

用法	增大或缩小文本的字符间距
值	normal、length

text-align

用法	对齐元素中的文本
值	left、right、center、justify

text-decoration

用法	将修饰应用于文本
值	none、underline、overline、line-through、blink

text-indent

用法	缩进元素中文本的首行
值	length、%

text-transform

用法	控制文本的大小写
值	none、capitalize、uppercase、lowercase

white-space

用法	指定如何处理元素中的空格
----	--------------

值	normal、pre、nowrap
<b>word-spacing</b>	
用法	增大或缩小字间距
值	normal、length

## B.15 CSS 背景样式属性

有几个 CSS 样式属性可用于修改页面的背景以及页面上各个元素的背景。

### background

用法	让你能够在一个地方设置所有的背景属性
值	background-color、background-image、background-repeat、background-attachment、background-position

### background-attachment

用法	指定背景图像是固定的还是随其他页面内容滚动
值	scroll、fixed

### background-color

用法	设置元素的背景色
值	color-rgb、color-hex、color-name、transparent

### background-image

用法	将图像用作背景
值	url、none

### background-position

用法	设置背景图像的起始位置
值	top left、top center、top right、center left、center center、center right、bottom left、bottom center、bottom right、x-% y-%、x-pos y-pos

### background-repeat

用法	设置是否以及如何重复背景图像
值	repeat、repeat-x、repeat-y、no-repeat

## B.16 CSS 边框样式属性

每个块元素都有样式可设置的边框。虽然可以让边框不可见，但有多种样式可应用于元素的边框。

### border

用法	让你能够在一个地方设置所有四条边框的所有属性
----	------------------------



值	<i>border-width</i> 、 <i>border-style</i> 、 <i>border-color</i>
<b>border-bottom</b>	
用法	让你能够在 一个地方 设置下边框的所有属性
值	<i>border-bottom-width</i> 、 <i>border-style</i> 、 <i>border-color</i>
<b>border-bottom-color</b>	
用法	设置下边框的颜色
值	<i>border-color</i>
<b>border-bottom-style</b>	
用法	设置下边框的样式
值	<i>border-style</i>
<b>border-bottom-width</b>	
用法	设置下边框的宽度
值	<i>thin</i> 、 <i>medium</i> 、 <i>thick</i> 、 <i>length</i>
<b>border-color</b>	
用法	设置四条边框的颜色
值	<i>color</i>
备注	可以使用 1~4 种颜色指定
<b>border-left</b>	
用法	让你能够在 一个地方 设置左边框的所有属性
值	<i>border-left-width</i> 、 <i>border-style</i> 、 <i>border-color</i>
<b>border-left-color</b>	
用法	设置左边框的颜色
值	<i>border-color</i>
<b>border-left-style</b>	
用法	设置左边框的样式
值	<i>border-style</i>
<b>border-left-width</b>	
用法	设置左边框的宽度
值	<i>thin</i> 、 <i>medium</i> 、 <i>thick</i> 、 <i>length</i>
<b>border-right</b>	
用法	让你能够在 一个地方 设置右边框的所有属性

值	<i>border-right-width</i> 、 <i>border-style</i> 、 <i>border-color</i>
<b>border-right-color</b>	
用法	设置右边框的颜色
值	<i>border-color</i>
<b>border-right-style</b>	
用法	设置右边框的样式
值	<i>border-style</i>
<b>border-right-width</b>	
用法	设置右边框的宽度
值	<i>thin</i> 、 <i>medium</i> 、 <i>thick</i> 、 <i>length</i>
<b>border-style</b>	
用法	设置四个边框的样式
值	<i>none</i> 、 <i>hidden</i> 、 <i>dotted</i> 、 <i>dashed</i> 、 <i>solid</i> 、 <i>double</i> 、 <i>groove</i> 、 <i>ridge</i> 、 <i>inset</i> 、 <i>outset</i>
备注	可以使用 1~4 种样式来指定
<b>border-top</b>	
用法	让你能够在一个地方设置上边框的所有属性
值	<i>border-top-width</i> 、 <i>border-style</i> 、 <i>border-color</i>
<b>border-top-color</b>	
用法	设置左边框的颜色
值	<i>border-color</i>
<b>border-top-style</b>	
用法	设置左边框的样式
值	<i>border-style</i>
<b>border-top-width</b>	
用法	设置左边框的宽度
值	<i>thin</i> 、 <i>medium</i> 、 <i>thick</i> 、 <i>length</i>
<b>border-width</b>	
用法	让你能够在一个地方设置四条边框的宽度
值	<i>thin</i> 、 <i>medium</i> 、 <i>thick</i> 、 <i>length</i>
备注	可以使用 1~4 个宽度值来指定

## B.17 CSS 边距样式属性

边距让你能够在元素的外边缘周围（元素边框的外面）增加空间。

margin

用法	让你能够在 一个地方 设置所有 边距属性
值	margin-top、margin-right、margin-bottom、margin-left

margin-bottom

用法	设置元素的 下边距
值	auto、length、%

margin-left

用法	设置元素的 左边距
值	auto、length、%

margin-right

用法	设置元素的 右边距
值	auto、length、%

margin-top

用法	设置元素的 上边距
值	auto、length、%

B.18 CSS 填充样式属性

填充让你能够在元素周围（边框内部）增加空间。

padding

用法	让你能够在 一个地方 设置所有 填充属性
值	padding-top、padding-right、padding-bottom、padding-left

padding-bottom

用法	设置元素的 下填充
值	length、%

padding-left

用法	设置元素的 左填充
值	length、%

padding-right

用法	设置元素的 右填充
值	length、%

padding-top

用法	设置元素的上填充
值	<i>length</i> 、%

B.19 CSS 布局和显示样式属性

在指定如何在页面上布局和排列元素方面，CSS 布局和显示样式属性扮演了至关重要的角色。

bottom

用法	设置元素下边缘同其父元素下边缘之间的偏移量
值	<i>auto</i> 、 <i>length</i> 、%

clear

用法	指定元素周围的哪些地方不能有其他元素
值	<i>left</i> 、 <i>right</i> 、 <i>both</i> 、 <i>none</i>

clip

用法	设置元素的形状
值	<i>auto</i> 、 <i>shape</i>
备注	显示时元素被裁剪为指定的形状

cursor

用法	指定要显示的鼠标形状
值	<i>url</i> 、 <i>auto</i> 、 <i>crosshair</i> 、 <i>default</i> 、 <i>pointer</i> 、 <i>move</i> 、 <i>e-resize</i> 、 <i>ne-resize</i> 、 <i>nw-resize</i> 、 <i>n-resize</i> 、 <i>se-resize</i> 、 <i>sw-resize</i> 、 <i>s-resize</i> 、 <i>w-resize</i> 、 <i>text</i> 、 <i>wait</i> 、 <i>help</i>

display

用法	设置是否以及如何显示元素
值	<i>none</i> 、 <i>inline</i> 、 <i>block</i> 、 <i>list-item</i> 、 <i>run-in</i> 、 <i>compact</i> 、 <i>marker</i> 、 <i>table</i> 、 <i>inline-table</i> 、 <i>table-row-group</i> 、 <i>table-header-group</i> 、 <i>table-footer-group</i> 、 <i>table-row</i> 、 <i>table-column-group</i> 、 <i>table-column</i> 、 <i>table-cell</i> 、 <i>table-caption</i>

float

用法	设置图像或文本相对于另一个元素的位置
值	<i>left</i> 、 <i>right</i> 、 <i>none</i>

left

用法	设置元素左边缘同其父元素左边缘之间的偏移量
值	<i>auto</i> 、 <i>length</i> 、%

overflow

用法	指定如果元素的内容容纳不下如何处理
----	-------------------

值	auto、visible、hidden、scroll
<b>position</b>	
用法	指定元素的布局为静态的、相对的、绝对的还是固定的
值	static、relative、absolute、fixed
<b>right</b>	
用法	设置元素右边缘同其父元素右边缘之间的偏移量
值	auto、 <i>length</i> 、%
<b>top</b>	
用法	设置元素上边缘同其父元素上边缘之间的偏移量
值	auto、 <i>length</i> 、%
<b>vertical-align</b>	
用法	设置元素的垂直对齐方式
值	baseline、sub、super、top、text-top、middle、bottom、text-bottom、 <i>length</i> 、%
<b>visibility</b>	
用法	指定应显示（可见）还是隐藏（不可见）元素
值	visible、hidden、collapse
<b>z-index</b>	
用法	设置元素的 z 顺序（堆叠顺序）
值	auto、 <i>number</i>

## B.20 CSS 列表和项目符号样式属性

你可能还没有意识到，通过 CSS 指定列表样式有多灵活。有几个 CSS 样式属性可用于列表和列表中的列表项目符号。

<b>list-style</b>	
用法	让你能够在一个地方设置所有的列表属性
值	<i>list-style-type</i> 、 <i>list-style-position</i> 、 <i>list-style-image</i>
<b>list-style-image</b>	
用法	设置用作列表项目符号的图像
值	<i>none</i> 、 <i>url</i>
<b>list-style-position</b>	
用法	指定项目符号在列表中的位置
值	inside、outside



list-style-type

用法	设置列表项目符号的类型
值	none、disc、circle、square、decimal、decimal-lead-ing-zero、lower-roman、upper-roman、lower-alpha、upper-alpha、lower-greek、lower-latin、upper-latin、hebrew、armenian、georgian、cjk-ideographic、hira-gana、katakana、hiragana-iroha、katakana-iroha

B.21 CSS 表格样式属性

有几个高级表格属性让你能够对表格的显示方式进行微调。

border-collapse

用法	设置表格边框的模式
值	collapse、separate

border-spacing

用法	设置相邻单元格边框之间的间距
值	<i>length length</i>

caption-side

用法	设置表题相对于表格的位置
值	top、bottom、left、right

empty-cells

用法	指定没有可见内容的单元格是否有边框
值	show、hide

table-layout

用法	指定表格布局
值	auto、fixed
备注	如果将其设置为 fixed，表格大小将是固定的，这样可以提高浏览器显示表格的速度



## 附录 C

# 从 HTML 迁移到 XHTML

虽然本书的多个地方都阐述了 HTML 和 XHTML 之间的关系，但我认为将这些信息集中在一个地方对读者会有所帮助。本附录可供读者快速获悉有关 HTML 4 和 XHTML 1.0 之间有何不同的信息，HTML 4 和 XHTML 1.0 是创建网页时涉及的两个主要的 Web 标准。

### C.1 XHTML 1.1 和 HTML 4 之间的差别

XHTML 1.1 是 HTML 4 的改进版，采用了更严格的 XML 规则。所幸的是，XHTML 和 HTML 之间的大部分差异都不会给 HTML 文档的整体结构带来巨大影响。将 HTML 文档迁移到 XHTML 与其说是转换到一种新语言，不如说是清理和减少代码。

下面列出了 XHTML 和 HTML 之间的主要差别，重点是 XHTML 要求的，而 HTML 不要求的。

- 元素和属性名必须小写（<a href="index.html">）。
- 非空元素必须有结束标签（<p>Howdy!</p>）。
- 空元素必须由一个开始/结束标签对或一个空元素组成（<br />）。
- 属性值必须用引号括起（href="index.html"）。
- 不能使用没有值的属性（ismap="ismap"）。
- XHTML 名称空间必须在 html 元素中声明。
- 不能省略元素 body 和 head。
- title 元素必须是 head 元素中的第一个元素。
- 文档必须使用 id 属性来唯一地标识页面中的元素；XHTML 不再将 name 属性用于这种目的。

基于读者的 XML 知识，不应对上述差别感到惊讶，尤其是你在本书中一直遵循了 XHTML 规则。幸运的是，这些差异在 HTML 文档中都很容易找到并修复，这使得从 HTML 迁移到 XHTML 相对比较简单。

## C.2 XHTML 和文档有效性

所有 XHTML 文档都必须指定文档类型定义 (DTD)，它用于将 XHTML 指定为网页使用的语言。在 XHTML 3.0 中，W3C 开发了三种不同的 DTD，它们都包含在 XHTML 规范中。这些 DTD 为 XHTML 提供了不同的详细程度，导致 XHTML 文档被分为三类。这里的理念是，如果不需要某些 XHTML 语言特性，可以使用较小的 XHTML DTD；如果需要更多的特性，可以使用较详细的 DTD。三种 XHTML 1.0 DTD 如下，它们的特性依次增多。

- **Strict**: 不能使用 HTML 格式元素 (font、table 等)，必须使用样式表来格式化文档。
- **Transitional**: 可以使用格式化文档的 HTML 元素。
- **Frameset**: 可以使用框架以及 HTML 格式化元素。

Strict DTD 是最小的 DTD，用于创建非常干净的 XHTML 文档，其中不包含任何格式化标签。要使用该 DTD 创建文档，必须使用样式表来格式化文本。XHTML 1.1 就是基于该 DTD 的，稍后将讨论该主题。Transitional DTD 在 Strict DTD 的基础上增加了对格式化标签的支持，当你不想花时间开发样式表时，该 DTD 对于快速转换 HTML 文档很有用。Frameset DTD 是这三个 DTD 中最大的，支持使用框架创建网页。

XHTML 1.1 没有采用三种 DTD，而只采用一种 DTD，该 DTD 与 XHTML 1.0 Strict DTD 极其相似。其思想是，将重点放在将 XHTML 1.0 中的内容和格式化分开，这意味说应使用样式表而不是老式的格式化标签。本书遵循 XHTML 1.1 及其严格方法来创建高度结构化的网页。

在所有 XHTML 文档的开头，必须在文档类型声明中声明其 DTD。在文档类型声明中，使用正式公开标识符 (Formal Public Identifier, FPI) 来引用标准的 XHTML 1.1 DTD。下面是一个如何在文档类型声明中声明 Strict DTD 的例子：

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

读者无需理解上述代码中 FPI 的细节，要点是它标识 XHTML 1.1 DTD，这对于所有的 XHTML 1.1 网页都是必须的。读者可能熟悉上述代码，因为它出现在本书每个示例网页的开头。请记住，文档类型声明的意义在于，必须将其放在每个 XHTML 网页的开头，这样它才被视为有效的 XHTML 文档。

## C.3 声明 XHTML 名称空间

除在文档类型声明中声明合适的 DTD 外，有效的 XHTML 文档还必须在 html 元素中声明一个 XHTML 名称空间。下面是与前面介绍的 DTD 相关联的 XHTML 1.1 名称空间：

```
http://www.w3.org/1999/xhtml
```

在 XHTML 1.1 文档的 html 元素中，使用名称空间声明属性 xmlns 来声明 XHTML 名称空间。下面是一个如何指定 XHTML 1.1 名称空间的例子：

```
<html xmlns="http://www.w3.org/1999/xhtml" ...
```

</html>

### C.4 将HTML文档转换为XHTML

虽然使用XHTML从空白开始创建新网页是个不错的主意，但现实情况是，有时候可能遇到需要将其转换为XHTML的HTML文档。所幸的是，对HTML 4文档进行转换，使其遵守XHTML 1.1规范并不太难。前面介绍了XHTML文档和HTML文档之间的差别，这些差别将成为将HTML文档转换为XHTML的指南。

执行从HTML到XHTML的转换时，可将下面的检查表作为指南。

- 1. 添加声明XHTML 1.1 DTD的文档类型声明。
- 2. 在html元素中声明与DTD相匹配的XHTML 1.1名称空间。
- 3. 将所有元素和属性名转换为小写。
- 4. 给每个开始标签加上结束标签。
- 5. 将每个空标签末尾的>替换为/>。
- 6. 将所有属性值用引号(")括起。

7. 确保所有元素和属性都是在XHTML 1.1 DTD中定义了；使用CSS样式表来代替已摒弃的格式化标签。

8. 将特殊字符转换为实体引用，例如，将"转换为&quot;（更详细的信息请参阅表C.1）。

表 C.1 在XHTML网页中应避免使用的字符

将其替换	替换为
&（和字符）	&amp;或&#38;
"（双引号/英寸符号）	&quot;或&#34;
<（左尖括号）	&lt;或&#60;
>（右尖括号）	&gt;或&#62;
[（左中括号）	&#91;
]（右中括号）	&#93;
'（撇号/单引号）	&#39;

**Did you know?**

**提示：**如果有很多HTML网页需要转换为与XHTML兼容的格式，强烈建议从<http://www.chami.com/html-kit/>下载免费的HTML-Kit软件。这个免费程序包含一个名为HTML-Tidy的模块，能够自动修改HTML文档，使其遵循前面提到的规则，并报告其他问题或不兼容性。HTML-Kit还是一个界面友好和设计良好的文本编辑器，包含很多用于编写XHTML网页的方便功能。需要指出的是，HTML-Tidy会修改代码中的换行和空格，如果你要整齐地缩进代码，可能需要手工调整格式。

将页面转换为XHTML后，可能想对其进行验证，以确保转换是成功的。W3C提供了免费的在线验证服务，你可以用来检查页面是否完全遵循了官方的XHTML标准。该验证服务位于<http://vallidator.w3.org/>，让你能够指定自己的计算机中以及已经发布到网上的要验证的XHTML文件。强烈建议你尝试使用该验证服务，因为它能够对XHTML代码的精确性做出权威判断。

有关XHTML文档转换的好消息是，本书已经教读者养成良好的编码习惯，对于读者新创建的网页，应该不存在转换的问题。遇到没有遵循XHTML的已有网页时，读者按照本附录列出的步骤应能够能够应付转换。如果执行上述每个步骤，应该能够得到合法的XHTML文档，它们遵循了XHTML 1.1标准的所有规则。

# 计算机精品学习资料大放送

软考官方指定教材及同步辅导书下载 | 软考历年真题解析与答案

软考视频 | 考试机构 | 考试时间安排

Java 一览无余: Java 视频教程 | Java SE | Java EE

.Net 技术精品资料下载汇总: ASP.NET 篇

.Net 技术精品资料下载汇总: C#语言篇

.Net 技术精品资料下载汇总: VB.NET 篇

撼世出击: C/C++编程语言学习资料尽收眼底 电子书+视频教程

Visual C++(VC/MFC)学习电子书及开发工具下载

Perl/CGI 脚本语言编程学习资源下载地址大全

Python 语言编程学习资料(电子书+视频教程)下载汇总

最新最全 Ruby、Ruby on Rails 精品电子书等学习资料下载

数据库精品学习资源汇总: MySQL 篇 | SQL Server 篇 | Oracle 篇

最强 HTML/xHTML、CSS 精品资料下载汇总

最新 JavaScript、Ajax 典藏级资料下载分类汇总

网络最强 PHP 开发工具+电子书+视频教程等资料下载汇总

UML 学习电子书下载汇总 软件设计与开发人员必备

经典 LinuxCBT 视频教程系列 Linux 快速学习视频教程一帖通

天罗地网: 精品 Linux 学习资料大收集(电子书+视频教程) Linux 参考资源大系

Linux 系统管理员必备参考资料下载汇总

Linux shell、内核及系统编程精品资料下载汇总

UNIX 操作系统精品学习资料<电子书+视频>分类总汇

FreeBSD/OpenBSD/NetBSD 精品学习资源索引 含书籍+视频

Solaris/OpenSolaris 电子书、视频等精华资料下载索引





# HTML与CSS

SAMS

入门经典

(第7版)

## 学习如何：

- 创建网页并将其放到网上；
- 在网页中加入自定义图像和图像映射；
- 使用级联样式表（CSS）创建卓越的网页布局；
- 使用 HTML 和 CSS 使 eBay 拍卖更有吸引力；
- 使用表单接受网站访问者的反馈；
- 使用 JavaScript 在网页中加入交互性；
- 了解如何在网站上通过 Google 广告赚钱；
- 创建和更新博客；
- 为网站做广告并让搜索引擎列出它。

## 24 章阶梯教学

学完本书全部内容，就能创建并发布自己的网站。本书采用直观、循序渐进的方法，每章建立在前一章的基础上，引导读者全面学习使用 HTML 和 CSS 创建和发布网页的关键知识。

**循序渐进的示例** 引导读者完成最常见的任务。  
**问与答、测验和练习** 帮助读者检验知识的掌握情况。  
**“注意”和“提示”** 指出捷径和解决方案。

封面设计 胡平利

分类建议：计算机 / 网页制作 / HTML  
人民邮电出版社网址：www.ptpress.com.cn

ISBN 978-7-115-15625-9



9 787115 156259 >

ISBN 978-7-115-15625-9/TP

定价：39.00 元