

HTML & Web

设计技术与技巧

*HTML & Web Design
Tips & Techniques*

Kris Jamsa
(美) Konrad King 著
Andy Anderson

吴悦 吴冲华 等译



机械工业出版社
China Machine Press



Education

Internet新技术丛书

HTML & Web 设计技术与技巧

Kris Jamsa

(美) Konrad King 著

Andy Anderson

吴悦 吴冲华 等译



A1008600



机械工业出版社
China Machine Press

本书介绍提高站点性能、丰富站点内容、使站点易于维护的方法与技巧, 提供了快速掌握HTML、多媒体、流视频、Web应用程序及电子商务等内容的途径。同时还包括Java applet、ActiveX 对象、JavaScript、ASP、PHP4等技术内容。书中涵盖的数百个唾手可得的、实际可用的Web 设计解决方案能使你工作起来更加轻松自如。

Kris Jamsa, Konrad King, Andy Anderson: HTML & Web Design Tips & Techniques (ISBN:0-07-219394-8).

Copyright © 2002 by The McGraw-Hill Companies.

Original language published by The McGraw-Hill Companies, Inc. All rights reserved. No part of this publication may be reproduced or distributed in any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition jointly published by McGraw-Hill Education (Asia) Co. and China Machine Press.

本书中文简体字翻译版由机械工业出版社和美国麦格劳-希尔教育出版公司合作出版, 未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有McGraw-Hill公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2002-2184

图书在版编目 (CIP) 数据

HTML & Web设计技术与技巧/(美)詹姆萨(Jamsa, K.)等著;吴悦、吴冲华等译. -北京:机械工业出版社, 2002.9

(Internet新技术丛书)

书名原文: HTML & Web Design Tips & Techniques

ISBN 7-111-10616-4

I. H… II. ①詹… ②吴… ③吴… III. 超文本标记语言, HTML-应用-万维网-站点-设计 IV.TP393.429

中国版本图书馆CIP 数据核字 (2002) 第051851号

机械工业出版社 (北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 白红莉 李云静

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2002年9月第1版第1次印刷

787mm × 1092mm 1/16 · 31.75印张

印数: 0 001 - 4 000册

定价: 49.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前言

据分析家称，在因特网（Internet）上使用的域名已经超过一亿个（关于Internet的统计数据，可以访问<http://www.nua.com/surveys>）。随着高速DSL和电缆调制解调器连接的广泛应用，以及随着公司对Web主机所需支付费用的降低，当前有上千万的用户都在建立个人Web站点。另外，目前的在线用户已超过5亿（这个数字正以每月7%的速度上升），为了吸引这些网上“冲浪者”的注意力，Web开发者必须找到开发新技术的方法。

在本书的各章中，你将学到把Web技术立即应用到Web页面中的方法。每一章都集中介绍一项能够快速使用的Web技术，像动态内容、安全性、数据库访问以及客户端和服务端处理等。首先介绍该技术的基础原理和在Web页面中应用它的最好方法，之后可以通过简单的剪切和粘贴将现成的解决方案应用到Web页面中，最后介绍使Web站点能够释放全部“能量”的技巧以及后台设置。

如果你立志开发一个快速的、充满艺术魅力并且可以吸引“眼球”的站点，那么本书可以帮助你实现这个愿望。

本书的适用范围

每天都有上百万的用户和专业Web开发人员在使用Web页面。为了简化他们的工作，很多Web开发者利用各种各样的Web开发软件来帮助他们完成Web页面的设计工作。但是，要开发全新的Web技术并能使站点运行，开发者们就必须大干一场并需要深入学习HTML标记、属性、层叠样式单（Cascading Style Sheets, CSS）规则、JavaScript、动态服务器页面（Active Server Pages, ASP）、PHP以及更多内容。

不管组成WWW（World Wide Web, 万维网）的站点有多少，谁都希望自己的站点更快、更安全并更易于维护。另外，站点的访客不仅想要看到有趣的内容，还希望可以快速地下载它们。而且大多数访客会认为那些宣称安全的站点真的很安全。也就是说，访客们除了对你放到站点上的内容感兴趣之外，他们还有质量、速度及安全性方面的要求。

本书验证了数百种能令Web开发者提高站点性能和安全性、丰富站点内容及易于站点维护的方法。实现本书提供的操作，不要求是有经验的编程人员或者网络管理人员。逐个步骤介绍的每个技巧都很容易完成，并且书中提供的解决方案可以很方便地粘贴进Web页面中。

在本书中，你将学习以下内容：

- 用来建立你的第一个Web站点的HTML基础知识，以及发布Web站点必须执行的步骤。
- 利用基于HTML的表格组织页面信息的方法，以及如何用表格布局文本和图片。
- 如何利用层叠样式单实现对Web页面中文本格式的完全控制以及简化以后的更新。
- XHTML与HTML的区别，你在页面中可以使用的XHTML的新性能，以及将HTML页面转

换成XHTML的方法。

- 更好地开发带图像页面的技巧，尽量减少下载图片时用户等待时间的方法。
- 如何使用PHP和ASP建立动态页面内容。
- 使用HTTP和CGI设置完成后台操作，用以自动识别用户浏览器类型、连接速度等。
- 确保你的系统免受黑客攻击，以及识别系统中可能会被恶意访问者利用的、潜在安全漏洞的方法。
- 如何在你的站点中将信用卡机制与电子商务结合起来。
- 其他更丰富的内容。

使用本书所提供技巧的要求

本书提供了大量的可以提高所开发站点的性能、安全性以及丰富站点内容和使站点更易于维护的方法。在每一章中，都能找到可以立刻在你的站点中应用的操作。

要使用这些技巧，并不需要高端的软件或者其他的图形图像制作工具。你只需要花费5到10分钟按照提示的说明一步步去做就行了。如果需要，提示中会告诉你到哪些站点（免费的）下载所需的工具。

对于高级Web开发人员和编程高手来说，本书介绍使用脚本完成枯燥操作的方法。书中所使用的所有编程语言或者是内置在浏览器、服务器中，或者可以从Web站点直接下载。如果你以前从来没有使用过这些语言也不用担心，在每一章的开始部分，对每种开发语言都提供了所需基础知识的介绍。

本书包含的内容

本书共有12章。每章介绍一种特定的Web技术、编程语言或者设计技术。

第1章“HTML基础” 作为全书的开始，为了更好地学习本书的后续内容，该章为Web开发新手打基础。设计者将要学习如何建立HTML页，如何将位于他们自己的PC中的文件以页面的形式显示出来，以及如何利用Web服务器使页面能够被更多访客通过Web访问。

第2章“HTML表格” 在Web页面中，表格用来组织大量的、需要在屏幕显示的数据。很多Web设计者还用表格控制页面中文本和图形的排列。例如，用表格可以使文本围绕在图片的周围，或者在页面中提供缩略图的表格，当用户点击缩略图时，完整图像（通常会很大）将被下载。

第3章“HTML表单” 在Web中，几乎所有的站点都使用表单来实现与访客的交互。表单可能会要求访问者提供各种各样的信息，像货运地址、个人数据或者用于支付费用的信用卡号码等等。另外，表单使用户可以对本站点及其他站点提供的货物、服务及内容进行查询。设计者使用HTML标记建立表单。当用户提交了信息后，Web服务器运行一个特殊的程序（开发者称为“脚本”）来对数据进行处理，像存储数据。在该章中，你将学习如何使用HTML建立表单，如何使用JavaScript检验用户输入数据的有效性，以及如何处理用户提交的数据。

第4章“层叠样式单” 在Web页面中，开发者使用各种各样的HTML标记来格式化文本、

图形和链接。只使用 HTML 标记，像粗体显示标记、斜体显示标记<i>等，Web 开发者的确可以按照其设想完成页面的格式化。但是，如果格式需要改动的话，开发者就不得不改动很多标记。层叠样式单（CSS）使 Web 站点更易于维护。使用 CSS，可以设置页面中文本的字体、颜色、对齐方式以及其他格式。如果需要对这些格式进行改动，只需要改变类型定义就可以了，所有定义了类型的文本格式立即随之改变。另外，希望寻找高级设计技巧的开发人员会发现，CSS 可以精确定位页面元素，甚至可以将文本和图形堆砌出无穷无尽的效果。

第5章 “XHTML和显示趋势” 为了保证你的 Web 页面在下一代的 Web 浏览器中也能运行良好，你必须遵循万维网联盟（W3C）的推荐标准。当按照推荐标准的规定时，XHTML 标准中的“规则”就禁止使用不正确的语法、缺失或者自定义的标记以及无效属性来建立页面。如果在代码中避免了上述情况，所开发页面就会有很好的外观并且能够在所有的标准浏览器上运行。另外，如果页面设计者们保证了页面描述的正确性（遵照 XHTML 的标准），浏览器开发者们就可以忽略掉用来“猜测”不规范的 HTML 意图的代码。另外，在非 PC 设备（像移动电话、掌上器件、车载便携式电脑等）上运行的浏览器不能够做得很大，因为这些设备通常都只有有限的存储能力。该章首先讲述如何运行一个应用程序（HTML Tidy），用来检测你的 Web 页面 HTML，矫正其中的错误并将 HTML 转换成 XHTML。你还将学习使用在线确认程序，该程序可以检测你新建的 Web 页面以确保使其遵守 XHTML 标准。然后讲述如何通过添加新的标记和属性来扩展标记语言属性的主题。最后还有关于如何检测用户浏览器类型，以确保你的页面可以在非 PC 设备上运行的浏览器中正确和快速地显示。

第6章 “图像” 图像在 Web 页面中是被广泛使用的。但是，很多提供大量图像的站点却因为下载这些图形图像必须花费很长的时间，而失去了很多没有耐心的访客。在该章中你将学习像高速缓存和预加载等技术以及一些设置技巧，使你能够大大减少图片的下载时间。另外，还要学习各种图形图像文件格式以及在什么场合使用它们。此外，该章还要介绍调色板以及如何确保站点内容是“Web 友好”的方法。最后，专业制图人员还将学习简化将图片放到 Web 上的方法。

第7章 “动画、声音与视频” 随着高速电缆调制解调器和 DSL 接入的出现，很多 Web 站点都使用了动画、背景音乐以及视频来吸引用户的注意。在该章中，你将学习如何综合使用各种动画，从简单的动态图形到高级的 Flash 动画。另外，你还要学习如何设置页面的背景音乐和如何按要求传递视频流。最后，将要学习的是如何从 Web 站点广播声音（就像一个基于 Web 的广播站）。

第8章 “JavaScript” JavaScript 是 Web 设计者应用于 Web 页面中的一种编程语言。开发人员将 JavaScript 语句放在定义了 Web 页面的 HTML 的标记中。例如，JavaScript 语句可能验证用户输入表单数据的有效性（该程序或脚本可能确保每一个字段非空或者出现的值正确并有意义）。JavaScript 语句也可能依据用户在页面中的操作执行具体的处理。例如，当用户点击了“帮助”链接后，可以使用 JavaScript 显示一个包含帮助信息的弹出式窗口。用户的浏览器既显示 HTML 页面也执行 JavaScript 语句。在该章中，你首先将学习如何建立简单的 JavaScript 程序，之后学习几个可以直接粘贴进 Web 页面中使用的 JavaScript 解决方案。

第9章 “Java applet 和 ActiveX对象” 多年来，程序员们都使用Java语言来建立可以运行于多种平台的applet。也就是说，同样的Java applet可以运行在Windows、Linux和 Mac操作系统下。Java applet使编程人员不必担心当用户下载这些applet时会遭遇计算机病毒。在Web世界中，有不计其数的你可能想要应用到你自己的页面中的“公用” Java applet。在该章中，你将学习如何将Java applet与HTML内容相结合。另外，该章还讲述如何对基于Windows的站点提供ActiveX支持。与Java applet不同，Java applet不会给下载它们的用户带来安全性的危险，而ActiveX对象不限制代码可以执行的任何操作。因此，只使用来自包含授权的可靠站点的ActiveX对象是很重要的。第9章详细介绍了你需要防范的内容。

第10章 “PHP4” PHP 是开发者用来建立动态（变化的）Web页面的编程语言。例如，使用PHP，你可以基于用户在表单中提供的信息，或者依据从你的站点预先存储在用户硬盘中的cookie中获取的信息，定制Web页面的内容。在Web世界，有超过1 000 000个站点使用PHP建立动态Web页面。在使用PHP之前，你必须下载并在系统中安装一个PHP解析器。该章将带领你完成整个安装过程。然后，你将学习如何建立简单的基于PHP的应用程序。正如上面提到过的，PHP是一种编程语言，所以该章的开始部分将向你介绍建立、测试和调试PHP程序的一些基础知识。接下来将提供几个可以在你的Web页面中快速应用的、基于PHP的解决方案。

第11章 “ASP” 与PHP一样，ASP为Web开发者提供了一种建立动态内容的方法。要建立动态服务器页面（即一个动态Web页面），你可以使用很多编程语言，包括VBScript、JScript（Microsoft 版的JavaScript）、PerlScript、Python、Rexx等等。该章是你开发自己的动态服务器页面的基础。另外，该章提供了大多数Web页面都需要的解决方案，像建立和处理cookie、处理表单、与数据库交互、使用用户名和密码控制站点的访问、显示和跟踪广告条等等。

第12章 “安全性与性能” 今年（2002年），黑客和计算机病毒花费了一般用户和企业超过120亿美元！在该章中，你将学习为保护站点，需要立即采取的步骤。如果当前你不是使用防火墙来保护站点（或者PC），就将学习如何下载、安装和配置站点。你还将学习到，选择不同的文件系统（操作系统用来存储磁盘上的文件和文件夹的软件）可以影响保护文件（使用授权和加密）的能力。另外，该章还介绍了建立安全传输（使用加密在后台交换数据包）所必须采取的步骤以及如何执行信用卡交易。最后，该章还介绍了一些可以用来提高你的站点性能的方法。

如何使用本书

虽然本书的内容安排是由浅入深的；但是，它们在结构上各自独立，你可以迅速跳到任何主题查找你需要的内容。

浏览全书，你会找到很多“实践”提示，这些内容可用来立即实现某种功能。

使用本书的参考站点

贯穿本书各章，我们提供了大量的HTML文件、动态服务器页面、JavaScript代码、PHP解

决方案等等。在大多数情况下，你都可以将这些方案剪贴进自己的Web站点。本书中所有的文件，都可以通过链接<http://www.osborne.com>站点得到。

本书原书名：HTML & Web Design Tips & Techniques

本书原书号：0-07-219394-8

参加本书翻译工作的除封面署名外还有：孙英霖、徐志勇、曾朝阳、高智勇、朱砂、梁开莉等。

目 录

前言	
第1章 HTML基础	1
1.1 下载和安装Personal Web Server	12
1.2 管理Personal Web Server和发布自己的Web页面	14
1.3 使用标题描述Web页面的内容	17
1.4 用文档类型定义标示Web文档	19
1.5 在Web文档中插入注释	20
1.6 指定Web页面文本的字体	21
1.7 用段落和换行标记控制文本的行文	22
1.8 使用标题级别标记和标记的size属性改变文本的字号	25
1.9 在HTML文档中改变文本的颜色	27
1.10 使用基本标记在Web页面上添加图片	28
1.11 改变文本和图片的对齐方式	30
1.12 向Web页面添加超文本链接	32
1.13 使用字符格式化标记改善单个字母和文字的显示效果	34
1.14 在HTML文档中使用符号和特殊字符	35
1.15 使用水平线组织Web内容	36
1.16 使用块引用标记控制文本的左右边界	37
1.17 创建有序列表和无序列表	39
1.18 创建嵌套列表	41
1.19 创建定义列表	42
1.20 使用预格式化文本标记控制Web内容显示	43
1.21 在Web页面框架内显示导航菜单	45
1.22 同时在屏幕上显示多个Web页面	46
第2章 HTML表格	49
2.1 创建含跨列或跨行单元格的表格	56
2.2 设置表格和单元格的边框宽度	58
2.3 设置表格和单元格的边框颜色	59
2.4 设置背景图像和颜色	61
2.5 确定颜色属性值	64
2.6 设置单元格补白和单元格间距	65
2.7 使用相对值或绝对值设置表格的大小	66
2.8 使用相对值或绝对值设置单元格的大小	67
2.9 水平或垂直对齐单元格的内容	68
2.10 在Web页面上对齐表格	70
2.11 通过插入一个透明GIF图像来控制单元格的宽度和高度	70
2.12 让文本对图像进行环绕	72
2.13 在表格中显示缩略图库	73
2.14 为表格和图片创建项目符号和列表	75
2.15 使用表格创建导航栏	76
2.16 在表格的单元格中添加图像和链接	77
2.17 在Web页面上利用嵌套表格控制边界	78
2.18 通过将一幅图像的组成小块放进表格来模拟图像映射图	79
2.19 将图像分割并放进表格单元格以创建一个快速下载的Web图像	80
2.20 缩短Web浏览器绘制表格的时间	81
2.21 用表格来模拟Web页面的框架	82
2.22 用单元格背景颜色吸引访问者的注意力	84
2.23 使用可视表格边框直观地对齐Web页面中的内容	85
2.24 控制Web页面文本的中缝尺寸和页边距宽度	86
第3章 HTML表单	89
3.1 在表单上创建单行输入区域	93
3.2 在表单上创建多行输入区域	95
3.3 在提交表单前验证文本元素的数据	96
3.4 在表单上添加复选框	98
3.5 在表单上添加单选钮	99
3.6 在提交表单结果前验证单选钮组的选择	100
3.7 在表单上放置下拉列表	102

3.8	在提交表单前核实访问者已经对选择列表作出了选择	104
3.9	根据访问者提供的信息改变选择列表中的有效条目	106
3.10	用隐藏区域将所有选择列表值发送给Web服务器	107
3.11	在表单上添加重置按钮	108
3.12	防止访问者意外清除表单元素	109
3.13	在表单上添加提交按钮	111
3.14	用其他图像替换表单中的标准提交按钮和重置按钮	112
3.15	使用不带CGI脚本的电子邮件发送表单结果	113
3.16	用HTML表格控制表单元素和文本的布局	114
3.17	用<label>标记创建表单导航快捷键	116
3.18	用onClick属性指示Web浏览器执行表单验证函数	118
3.19	通过隐藏区域将值传递给Web服务器	119
3.20	用密码元素隐藏访问者的输入	120
3.21	在表单上添加一个普通按钮对象	121
3.22	根据情况控制表单元素的使用和禁用	121
第4章	层叠样式单	124
4.1	一个选择器使用多个CSS规则	133
4.2	用font-family属性定义字体	135
4.3	用font-size属性定义字体大小	136
4.4	创建覆盖文本	138
4.5	对齐Web页面文本	139
4.6	设置边空与行高	141
4.7	把文本放入列中	142
4.8	使用边框	144
4.9	段落缩进和文字之间距离的控制	146
4.10	段落首字母的设置	147
4.11	自定义超链接的外观	148
4.12	创建阴影效果	149
4.13	使用边框图像	150
4.14	设置背景图像和水印	152

4.15	浮动图像和文本	154
4.16	自定义列表的外观	155
4.17	用过滤器创建文本和图像效果	157
4.18	对齐表单的标签并为表单增加颜色	159
4.19	显示带说明文字的缩微图	161
4.20	控制鼠标指针形状	162
4.21	布局页面元素	164
4.22	检验样式单	165
第5章	XHTML和显示趋势	168
5.1	使用HTML Tidy将HTML转换成XHTML	178
5.2	为网页选择正确的DOCTYPE	181
5.3	使用XHTML有效性验证器来检验网页的有效性	183
5.4	在XHTML网页中用关键字设置字体大小	186
5.5	用<optgroup>标记创建分组的XHTML选择列表	187
5.6	使用层叠样式单规则为XHTML表格添加颜色	189
5.7	在XHTML网页中使用CSS规则嵌入字体	191
5.8	在XHTML网页中使用内联帧嵌入另一个XHTML网页	193
5.9	一次更新多个XHTML页面内联帧	195
5.10	根据媒体类型来改变XHTML页面的外观	198
5.11	控制Web浏览器打印XHTML网页的方式	200
第6章	图像	203
6.1	带图像页面的跨平台问题	208
6.2	在标记中指定图像尺寸	211
6.3	使用alt属性显示替代图像的文字	213
6.4	用标记的title属性创建工具提示	214
6.5	把照片压缩为JPEG格式	216
6.6	用GIF格式文件保存剪贴画和文字	218
6.7	使图像下载的速度显得更快	219
6.8	PNG-8和PNG-24文件格式	220
6.9	用图像编辑器转换图像文件格式	221

6.10 用扫描仪制作用于Web页面上的图像	222	8.10 调用一个用户定义的JavaScript函数	292
6.11 创建彩色横线	223	8.11 在事件处理器中调用JavaScript函数	294
6.12 用lowsrc属性快速下载图像	224	8.12 深入了解JavaScript事件处理器	295
6.13 使用Web安全调色板来精确显示颜色	225	8.13 利用翻转效果创建交互式的 导航条	297
6.14 创建图像超链接	226	8.14 利用Scripting对象的原型数组	299
6.15 用图像创建平铺背景	227	8.15 通过名字替代位置码来查询Web页面 对象	300
6.16 创建GIF格式的透明图像	229	8.16 支持文档对象的目录	301
6.17 点击缩略图像得到完整尺寸的图像	231	8.17 利用JavaScript图像数组	304
6.18 预下载并缓存图像	233	8.18 使用JavaScript链接数组	305
6.19 用抖动技术扩展Web安全调色板	234	8.19 使用JavaScript改变Web页面颜色	307
6.20 使用抗锯齿功能的问题	235	8.20 在访问者的计算机硬盘上存储cookie 文件	309
第7章 动画、声音与视频	238	8.21 用JavaScript格式化cookie数据	310
7.1 从头开始创建GIF动画	253	8.22 从cookie文件恢复cookie值	312
7.2 通过内部设置控制GIF动画	254	8.23 从cookie文件中删除一个cookie	313
7.3 用GIF动画创建广告条	256	8.24 利用外部脚本节约时间和减少编程	314
7.4 使用中间帧动画在GIF动画中产生平滑 过渡的效果	258	8.25 利用onLoad事件创建动画	316
7.5 在Web站点上插入一个完整的Flash 闪屏	260	8.26 利用JavaScript显示自替换广告条	317
7.6 从头创建一段Flash电影	262	8.27 指示超链接到新文件	318
7.7 使用FlaX创建基于文本的动画	264	8.28 预缓存图片以缩短图像显示的时间	320
7.8 广播流式音频和视频	264	8.29 使用JavaScript创建一个滚动文本框	321
7.9 创建自己的流式媒体	267	第9章 Java applet和ActiveX对象	323
7.10 创建使用Web摄像机的页面	268	9.1 使用applet创建弹出式导航菜单	338
7.11 用SMIL将视频和音频集成到Web 站点	269	9.2 使用applet以动画形式显示文本字符串	340
第8章 JavaScript	273	9.3 使用applet显示和打印任意一年的日历	341
8.1 处理不支持脚本的旧版浏览器	282	9.4 使用applet随机显示文本文件中的一个 段落	342
8.2 通过使用JavaScript数组在一个变量中 存储多个数值	283	9.5 使用applet创建一个在自己的窗口中运行 的导航菜单	344
8.3 让脚本做出判断然后进行处理	284	9.6 使用applet在屏幕上的矩形框中垂直滚动 文件内容	347
8.4 基于两个或多个条件做出判断	286	9.7 使用applet沿正弦曲线以动画形式显示 图像	349
8.5 条件非真时执行的代码	286	9.8 通过编辑Java源代码来创建定制 的 applet	350
8.6 指定次数的重复语句	287		
8.7 条件为真时的循环语句	288		
8.8 响应JavaScript事件	288		
8.9 在Web页面主体中执行JavaScript语句	290		

9.9 使用Microsoft Chat ActiveX控件向Web页面添加Internet Chat	355	11.3 使Web浏览器重定向到另一个网页	426
9.10 在Internet Explorer中只使用ActiveX对象	357	11.4 使用cookie在HTTP请求之间保留变量	428
第10章 PHP4	360	11.5 从ASP Form集合中获取表单结果	431
10.1 从HTML表单向PHP脚本发送数据	376	11.6 从ASP QueryString集合中获取表单结果	433
10.2 使用PHP解析、提取表单数据	379	11.7 从服务器变量集合中获取信息	436
10.3 使用PHP发送e-mail消息	382	11.8 通过MyODBC驱动程序连接到MySQL数据库管理系统	438
10.4 判断访问者的Web浏览器是否接受cookies	384	11.9 设置对网站的用户名/密码访问	440
10.5 使用PHP和硬盘文件来设置访问网站的用户名/密码	386	11.10 开始一个会话并与会话变量一起工作	443
10.6 防止访问者直接链接到网站中的页面	388	11.11 执行SQL查询并在HTML表中显示查询结果集	446
10.7 使用PHP会话在网站的访问者和Web服务器之间建立持久的连接	389	11.12 用Microsoft Banner Ad Rotator显示广告条	450
10.8 创建MySQL数据库和数据表	391	11.13 跟踪Microsoft Banner Ad Rotator印记和点击通道	453
10.9 在Web页面上用HTML表格列出SQL查询结果	392	11.14 处理“Status: 404 Not Found”错误	457
10.10 使用PHP生成随机的密码	394	第12章 安全性与性能	461
10.11 使用PHP和MySQL来定位访问者在网站上所在的位置	396	12.1 下载和安装公钥、数字签名以及服务器ID	470
10.12 防止访问者修改URL参数中的变量值	398	12.2 在IIS下创建安全网页	471
10.13 使用PHP和MySQL来跟踪访问者在网站上所在的位置	399	12.3 安装基于软件的防火墙	472
10.14 从通过代理服务器发送的Web页面请求中获取访问者的IP地址	401	12.4 细调防火墙的端口分配	474
10.15 防止访问者猜测其他访问者的PHP会话标识	402	12.5 减少你的站点对病毒的漏洞	475
10.16 使用PHP函数创建Web页面模板	404	12.6 通过禁用打印和文件共享提升性能和安全性	478
10.17 使用PHP在Web页面上加入文件上载功能	406	12.7 使用客户端证书限制用户访问	478
第11章 ASP	409	12.8 审核系统事件以检测入侵者	480
11.1 控制从Web服务器到浏览器通过HTML输出流传送的数据内容	421	12.9 使用NTFS文件系统	482
11.2 防止Web浏览器显示过期的ASP	425	12.10 禁用远程服务	484
		12.11 分析系统的弱点	486
		12.12 处理信用卡数据	487
		12.13 分析Web站点的性能链	488
		12.14 创建网络“农场”	492
		12.15 监测服务器性能	492

第1章 HTML基础

超文本标记语言 (Hypertext Markup Language, HTML)是万维网(World Wide Web)的核心计算机语言。创建Web站点时,需要使用HTML向组成站点的各个Web页面放置文本、图片、动画,可能还有视频与音频信息等。此外,HTML还允许你插入超文本链接和交互操作按钮,用来将一个Web页面与站点内的其他Web页面或与全球范围内其他站点上的Web页面连接起来。因此Web设计是一个创造过程,简单地说HTML就是一种用来设计Web页面的工具(页面描述语言)。

HTML是一种文本标记语言,而非编程语言。在原理上,使用HTML语言创建的Web页面对于任何拥有一台计算机、一个Web浏览器,并且对能访问Internet的人来讲都应该是可见的。但是实际上,浏览Web网页上所有内容的能力与各人所用Web浏览器的具体性能有关。Web浏览器是一些用来转换Web页面文档中的HTML,并在访问者的计算机屏幕上显示文本、图片和动画的程序。浏览器本身(或借助于安装的其他程序)还可以播放使用HTML插入到Web页面的任何视频和声音文件。Microsoft公司的Internet Explorer和Netscape公司的Netscape Navigator是当前最为流行的两种Web浏览器,它们的最新版本几乎可以显示你能够使用HTML放置到Web页面上的任何信息。

用户要想访问你在Web站点上的页面,就必须首先连接到Internet上并启动一个Web浏览器软件。Web服务器将一个网页发送到用户的计算机上之后,Web浏览器将HTML转换为Web页面文件,然后再把文件的内容作为文本和图形显示在浏览器的应用窗口中。

Web服务器和Web浏览器使用超文本传输协议(Hyper Text Transport Protocol, HTTP)进行通信。HTTP协议的主要内容之一是规定了Web浏览器和服务器发送消息的方法以及消息本身的结构。本书将不详细讨论各种HTTP消息的类型和结构。不过,即使要设计和创建一个比较复杂的Web站点,你也只需要对图1-1中所示的(Web浏览器和Web服务器之间的)HTTP请求和响应过程有一个基本的了解就可以了,其过程如下:

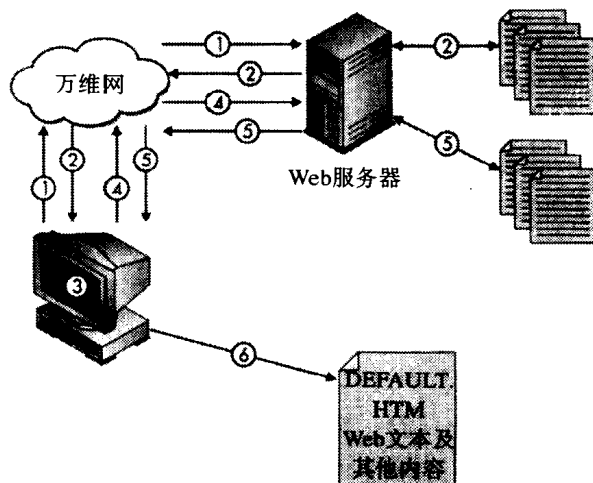


图1-1 Web服务器处理一个Web内容请求的过程

1) 用户每次向浏览器的地址栏中输入一个Web地址(例如http://www.NVBizNet.com)并按下回车键后, Web浏览器便向Web服务器发送一个HTTP请求, 要求通过Internet传送一个文件。

2) Web服务器收到这个要求传送文件的HTTP请求后, 找到所请求的文件并将该Web页面文件发送给Web浏览器。

3) Web浏览器分析Web页面文件, 确定其中是否有需要从Web服务器上获得的嵌入文件(例如图形、动画、声音等)。

4) Web浏览器向Web服务器发送多个HTTP请求(与浏览器所需要的嵌入文件一一对应)。

5) Web服务器收到需要传送文件的请求后, 找到各个文件(每个HTTP请求对应一个文件)并发送给Web浏览器。

6) Web浏览器将根据原来的Web页面文件, 并且按照文件中的HTML标记所给出的指示, 将Web页面和嵌入文件组装成一个Web文档, 显示在计算机屏幕上。

理解Web浏览器和Web页面的角色

为了适应市场的不断变化, 并让所有的人都能访问Internet, 微软公司和Netscape公司针对所有商业计算机平台和操作系统推出了各种不同版本的Web浏览器软件。由于Web浏览器必须转换HTML代码, 以便在计算机屏幕上显示Web页面, 因此用户访问Web站点时所使用的Web浏览器的品牌和版本就决定了HTML中的哪些特性将对它们起作用。因此, 在你的Web页面设计中使用最新最先进的特性和网页格式化能力, 并不能保证浏览页面的每个访问者都能看到用HTML放到页面上去的所有信息。

在设计Web页面的过程中, 需要牢记的是, 并非每一个访问你站点的用户使用的都是最新版本的Internet Explorer或Netscape Navigator。所以在设计Web页面的大部分内容时, 应当尽量使用将在本章中学习到的HTML基础知识。然后再使用HTML的一些较为先进的特性来添加有趣的花样, 以便访问站点的用户下次还想再到这儿来看看。因此通过在同一页面的设计中恰当地结合使用HTML的基础特性和先进特性, 就可以保证既能让每一位访问者都能访问到你想要发布的重要信息, 又能让那些使用最新浏览器的访问者拥有一次真正回味无穷的访问经历。

Web页面是由一系列HTML指令组成的, 因此可以使用任何文本编辑器来将HTML指令输入到文件中。正如前面曾经提到过的, Netscape Navigator和Internet Explorer这些Web浏览器能够根据你所创建的文本文档中的指令来将Web页面显示到计算机屏幕上。(Web页面内容是指你使用HTML放在Web页面上的文本、图片及其他信息, 例如视频与声音信息。)

如果你认为创建Web页面很容易, 那么你算说对了。实际上, 在刚开始创建网站时, 一点儿也不麻烦。最初的HTML标准描述的是一种毫不复杂、易于学习的语言。你可以使用它来创建纯文本文档, 它对于任何能够访问Internet的人都是可见的。尽管HTML的标准委员会——万维网联盟(World Wide Web Consortium, 简称W3C)已经向HTML语言增加了许多新的指令(称为标记和属性), 但你仍然可以通过在使用某种文本编辑器(如Windows Notepad)建立的文档中输入简单的HTML命令来创建极富特色的Web页面。通过“http://www.w3.org/”网址访问万维网联盟, 可以获得有关HTTP、HTML、Extensible Hypertext Markup Language (XHTML)、Extensible Markup Language (XML)、Cascading Style Sheets (CSS)等各种Internet技术的完整描述。

既然你已经知道了Web页面是一种什么东西以及(在一般意义上)如何创建,那么现在让我们来快速浏览一下Web页面HTML。如果与Internet没有固定的连接,你可以通过ISP来建立一个拨号连接。下一步是启动Web浏览器,在地址栏中输入URL(统一资源定位器, Uniform Resource Locator, 即Web页面的地址),显示自己喜欢的Web页面。比如,你可以在Web浏览器的地址栏中输入<http://www.Osborne.com>,然后按下回车键。

随后,Web浏览器将向Web服务器发出一个请求,要求传送你输入地址所对应的Web页面。当Web服务器作出响应,将该Web页面发送给你的Web浏览器之后,浏览器将在屏幕上显示这个页面,并向Web服务器请求页面上嵌入的所有对象(如图形图像)。当浏览器取得并显示了所有的页面内容之后,就可以选择View | Source 以显示用来描述页面的HTML(如果使用的是Netscape Navigator,则选择View | Page Source)。相应地,Internet Explorer将使用Windows Notepad来显示Web页面的HTML,而Netscape Navigator则会在一个新的窗口屏幕中显示HTML语句。不必担心看不懂各条HTML语句的含义,在学习完这本书之后,你就会明白一切。至于现在,重要的就是记住你可以看到任何Web页面的源代码。

HTML的容器标记

从本质上讲,Web页面是一个文本文件,其中包含HTML代码(称作标记, tag)和属性(attribute)形式的指令。标记是用来指示Web浏览器格式化你想要放置在Web页面上的文本和插入的图形图像的命令。有一部分(但不是全部)HTML命令要求有开始标记和结束标记,这些标记被称为容器标记(container tag),因为起始标记中的指令将被应用于Web页面中包含的、位于起始标记和结束标记之间的所有内容。

每个HTML命令(即每个HTML标记)都以一个小于号(<)开始,后面紧跟着标记的名称以及各种属性,并以一个大于号(>)结束。要为一个开始标记创建一个结束标记,只需在标记的名称前面插入一个(/)符号即可。因此,开始标记的形式为<标记名[属性]>,结束标记的形式为</标记名>。标记名告诉Web浏览器标记的用途,标记名之后的属性(如果有的话)为Web浏览器提供执行标记命令所需的附加信息。

例如,下面的代码显示了段落开始标记(<p>)和段落结束标记(</p>)是如何包含一段文本的。在这个例子中,<p>标记指示Web浏览器使用默认的格式化规则来显示直到</p>标记之前的文本。

```
<p> This is an example of paragraph text. </p>
```

如果你希望浏览器使用某种特定的字体和颜色来显示段落文本,可向位于<p>标记之后、需要指定显示样式的文本之前的标记中增加指定的颜色、字体、字号等其他特征的属性。(阅读完后面两部分有关标记的更多内容之后,我们将开始学习有关属性的内容。)

HTML的部分标记

为了对描述一个Web页面的各部分HTML进行统一组织,就需要使用一组部分类型的标记。在Web页面定义的各个部分中,这种类型的HTML标记有着特定的用途,以下列出了部分标记:

- <html></html> 用在HTML文档的起始和结束位置。起始和结束HTML标记

(`<html></html>`)包含了用来描述Web页面的所有其他HTML标记。

- `<head></head>` 起始和结束头部标记紧跟在起始HTML标记(`<html>`)之后, 指示一个Web页面的头部。可以使用Web页面头部中的标记来包含作者的姓名、作者创建该页面的日期等信息。另外, 还可加入带有描述页面信息的标记。这样Web搜索引擎就可以向其搜索索引中加入指向你的页面的引用。在放入页面头部的HTML标记和信息中, 访问者的浏览器只显示Web页面的标题。Web页面的标题应当插入到位于起始和结束title标记(`<title></title>`)之间的头部部分。具体设置请参见后面的代码范例。

- `<body></body>` 起始和结束主体标记紧跟在Web页面的头部之后, 指示一个Web页面的主体。Web页面的主体部分包含有指示Web浏览器应该在屏幕上显示什么以及如何显示的标记。

下列这段代码给出了HTML的部分标记的正确使用方法:

```
<html>
<head>
  <title> The Web Page's Title </title>
</head>
<body>
  Text content and body tags inserted here
</body>
</html>
```

需要注意的是, 在书写HTML标记时, 可以使用大写字母、小写字母, 或者大小写字母混合书写, 因为目前Web浏览器对大小写并不敏感。不过, 尽管Web浏览器在一段时间之内还将继续支持大写字母标记和小写字母标记, 但XHTML和XML这些标准规范就要求必须使用小写字母标记。因此, 最好的做法是将标记和属性全部使用小写字母书写。这样, 一旦将来的Web浏览器强制Web设计人员遵守更新的标准时, 它仍然可以正确加载你今天所创建的Web页面。

HTML的空标记

除了容器标记之外, HTML使用的另外一种标记叫做空(empty)标记。容器标记将页面内容(例如一行文本)包含在一个起始标记和一个结束标记之内, 而空标记则不需要结束标记。我们可以把空标记设想为像“来这儿”或“做这个”这样的单个命令或语句。例如, 下面的代码使用换行符标记(`<br>`)来指示Web浏览器在显示页面上的下一个内容(本例中是下一行文本)之前, 先转移到下一行:

```
<html>
<head>
  <title> Example of the line break tag </title>
</head>
<body>
  <p> This text is displayed on line one <br>
    This text is displayed on line two </p>
</body>
</html>
```


创建Web页面

HTML文本文档的创建其实很容易。要创建一个HTML文档,你只需要打开一个文本编辑程序并输入HTML代码。HTML代码输入完毕之后,应将该文档保存为一个文本文件,使用.htm或.html作为扩展名。举个例子,启动你喜欢使用的文本编辑器(如Windows Notepad)并输入下列HTML代码:

```
<html>
<head>
  <title>Example of a Simple HTML Document</title>
</head>
<body>
  <p>HELLO WORLD! Here I am.</p>
</body>
</html>
```

现在,将你的工作作为一个Web页面文件(即使用.htm或.html作为扩展名的文件)保存到硬盘上。考虑到你很可能希望从Osborne站点下载这些代码或者自己动手将它们输入到文本编辑器中,而且你还可能想保存在阅读本书的过程中将会遇到的许多Web页面,因此你可以建立一个文件夹(如C:\HTMLExamples),用来保存Web页面。然后,将你在上面这个例子中输入的代码以“TestPage.htm”的名称保存在新建的文件夹中。

为了测试一下你的第一个Web页面,可以启动Web浏览器,在地址栏中输入“File://”,后面输入驱动器名和Web页面的路径。例如,如果你将本例中的Web页面以“TestPage.htm”的名称保存在“C:\HTMLExamples”文件夹中,则应当在浏览器的地址栏中输入“File:///c:/HTMLExamples/TestPage.htm”。在按下回车键后,Web浏览器将会显示图1-2中所示的网页。这就是你所需要做的全部工作!现在,你也成了一个真正的Web页面设计者了。

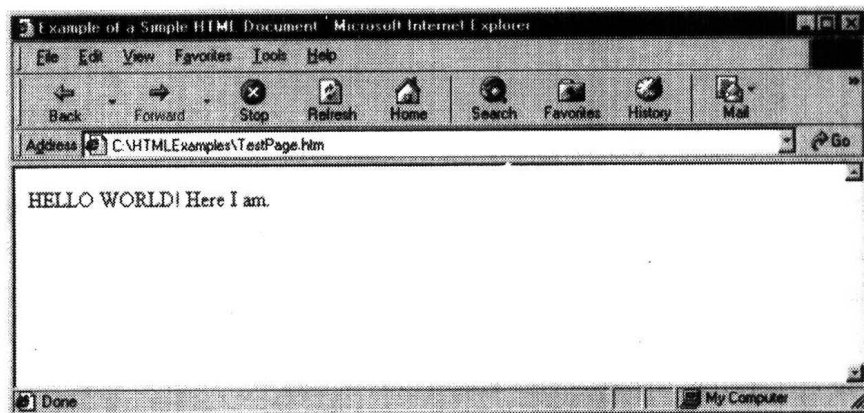


图1-2 Internet Explorer显示的简单Web页面

控制文本的格式

在标准的文字处理中,使用段落来定义一组句子。通常情况下,一个空白行或首字缩进就