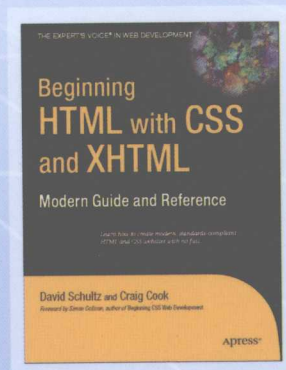


Beginning HTML with CSS and XHTML

# 深入浅出HTML

[美] David Schultz 著  
Craig Cook 译  
谢廷晟 译

- 简明实用的Web开发入门教程
- 强调现代Web标准
- 紧密结合开发实践



人民邮电出版社  
POSTS & TELECOM PRESS

**TURING**

图灵程序设计丛书 Web开发系列

第1版 (2001年) 第1次印刷

Beginning HTML with CSS and XHTML

# 深入浅出HTML

[美] David Schultz 著  
Craig Cook  
谢廷晟 译

人民邮电出版社  
北京

## 图书在版编目 (CIP) 数据

深入浅出 HTML / (美) 舒尔茨 (Schultz, D.), (美) 库克 (Cook, C.) 著; 谢廷晟译. —北京: 人民邮电出版社, 2008.5

(图灵程序设计丛书)

书名原文: Beginning HTML with CSS and XHTML

ISBN 978-7-115-17736-0

I. 深… II. ①舒…②库…③谢… III. 超文本标记语言, HTML—主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2008) 第 027423 号

## 内 容 提 要

本书是一部讲述现代 Web 标准的优秀教程, 彻底摒弃了过时的内容, 始终贯彻三层分离的思想。书中结合实例讲述如何使用 HTML、CSS 设计符合现代 Web 标准的网页, 并讲解了如何使用 JavaScript 添加动态行为。书中的主体部分让读者接触 HTML、XHTML 的各种主题, 逐渐熟悉各种元素。最后一章将书中讲述的各种技术付诸应用, 带领读者一步步地创建一个实用的网站。

本书适合 Web 设计和开发初学者阅读。

图灵程序设计丛书

## 深入浅出 HTML

- 
- ◆ 著 [美] David Schultz Craig Cook
  - 译 谢廷晟
  - 责任编辑 杨 爽
  - ◆ 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>
  - 北京顺义振华印刷厂印刷
  - 新华书店总店北京发行所经销
  - ◆ 开本: 800×1000 1/16
  - 印张: 20.5
  - 字数: 485 千字
  - 印数: 1—5 000 册
  - 2008 年 5 月第 1 版
  - 2008 年 5 月北京第 1 次印刷

著作权合同登记号 图字: 01-2007-4254 号

ISBN 978-7-115-17736-0/TP

定价: 39.00 元

读者服务热线: (010) 88593802 印装质量热线: (010) 67129223

反盗版热线: (010) 67171154

# 版 权 声 明

Original English language edition, entitled *Beginning HTML with CSS and XHTML: Modern Guide and Reference* by David Schultz, Craig Cook, published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright © 2007 by David Schultz, Craig Cook. Simplified Chinese-language edition copyright © 2008 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Apress L.P.授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

## 译者序

有这样一种说法：FrontPage的问世，圆了众多业余爱好者的网页设计梦，但也使因特网上涌现了无数丑陋的网页。其实名声更响亮的网页设计工具DreamWeaver又何尝不是如此？这类“所见即所得”的网页设计工具可以大大提高网页设计的工作效率，但是如果对它所创作出来的文档的语法不熟悉的话，你不可能真正驾驭这些工具，也摘不掉“丑陋网页制造者”这顶羞于示人的帽子。这种丑陋不仅仅体现在美术意义上，还体现在源代码意义上。美术意义上的丑陋可以通过提高绘画和平面设计水平加以解决，但再高超的绘画技艺也无法帮你解决源代码意义上的丑陋问题。

各种所见即所得的网页设计工具，无论它叫什么名字、由什么公司制作、有什么样的操作界面、是否免费，用其创作出来的网页，都是HTML文档，即用HTML语言编写的文档。这种文档中可能还夹杂着一些CSS代码和JavaScript代码。在设计网页的过程中，也可能会产生一些独立的专门包含CSS代码或JavaScript代码的文件。显然，如果不熟悉HTML、CSS和JavaScript，就不可能真正理解网页的工作机制，也不可能设计出高质量的网页。我个人认为，学习网页设计最好从学习这些本质性的技术入手。一个精通这些技术的人，仅仅使用记事本这样简陋的文本编辑器也能制作出精美而且简练的网页（实际上有些专业人员的确更喜欢这样手工编写网页文档，而不是使用DreamWeaver这类工具）。有了这样的技术功底，再来学习使用DreamWeaver这类网页设计工具就非常简单了，而且你也能深刻地认清其种种利弊，并根据实际情况合理地使用这些工具。反之，一个从所见即所得的网页设计工具入手学习网页设计的人，也许很快就能学会在设计视图中用鼠标拖拖画画地做出一些令初学者兴奋不已的花哨效果，而且，如果他是个出色的平面设计师的话，这样做出来的网页在浏览器中看起来甚至还可能非常漂亮，但他可能很快也会发现有许多事仅仅使用设计视图难以办到、需要在代码视图中直接编辑源代码，而这这就要求他必须熟悉HTML、CSS和JavaScript——到头来，他还是必须学习这些技术，否则他就只能止步于一个比较肤浅、业余的层次。由对HTML和CSS一无所知的人用这些工具制作出来的网页，从源代码的角度来看往往比较臃肿、丑陋。

为什么这里一再提到源代码是否丑陋的问题？只要网页在浏览器中看起来不丑陋，它的源代码丑不丑陋有什么要紧吗？这样纠缠于源代码是否丑陋的问题是否有洁癖之嫌？我在这里所说的丑陋的源代码，是指那种充斥着不必要的或语义上不恰当的HTML标签的代码，是那种把文档的结构性因素、表现性因素和行为性因素混杂在一起的代码。我曾经被要求为一个由美工（他正是那种对HTML和CSS不太熟悉的人）初步设计好的网页添加后台逻辑。这需要对源代码进行修改。由于原来的网页有一些缺陷，所以我决定先改正一下这些缺陷再说。一打开源文档，只见大

堆大堆的表格标签像俄罗斯套娃一样嵌套了无数层，看得人眼花缭乱，而且代码中还夹杂着一些语法错误。我在这堆乱麻中挣扎了半天，最后终于失去了耐心，索性自己动手从头设计这个页面，结果反而很快就搞定。在这个事例中，我所面对的这种由无数层表格定义网页布局的代码，正是把结构因素和表现因素混杂在一起的典型。源代码丑陋的网页对非可视化用户代理（如屏幕阅读器）很不友好，而且很难维护，修改起来工作量比较大，也不便于用脚本进行操纵，而且由于充斥着大量不必要的东西，自然其文件也就更大，需要更多的时间下载。因此，源代码的质量是网页设计工作中必须加以重视的一个问题。

如今的网页设计方法与几年前相比已经有了很大变化。这主要体现在结构层、表现层和行为层分离这一思想上。表现性的HTML标签逐渐退出舞台，曾经风靡一时的表格布局技术也日渐式微，取而代之的是CSS的风行和对HTML标签语义正确性的强调。对于初学者来说，一本彻底摒弃过去种种陋习的HTML和CSS教材尤为重要。这可以保证他们不会背上一大堆过时的标签、属性和技术的包袱，而且不会沾染上种种不良习气。本书正是这样一本入门级别的教材。作者用简练的笔法讲述了HTML和CSS的基本知识，也附带简单介绍了一下JavaScript。贯穿整本书的，是三层分离的思想和对Web标准的强调。这本书可以帮助初学者迅速掌握HTML和CSS的基础知识。

当然，这仅仅只是一本入门教材，不可能详细讲述HTML和CSS的方方面面。如果你需要进一步深入学习HTML和CSS的话，我建议先看看这两本书：《HTML & XHTML权威指南》（*HTML & XHTML: The Definitive Guide*, Bill Kennedy和Chuck Musciano著）和《CSS权威指南》（*CSS: The Definitive Guide*, Eric A. Meyer著）。这两本书都由O'Reilly出版社出版，目前国内都已经引进了最新版本（分别是第6版和第3版）。如果你还要承担客户端脚本编程工作的话，还应该专门学习一下JavaScript语言。这方面的书我建议先看看这两本：《JavaScript权威指南》（*JavaScript: The Definitive Guide*, David Flanagan著，O'Reilly出版）和《JavaScript高级程序设计》（*Professional JavaScript for Web Developers*, Nicholas C. Zakas著，Wrox出版，中文版由人民邮电出版社出版）。这两本书的最新版本分别为第5版和第1版，国内都已经引进。

本书作者在前言中有这样一句话：“如果你是一个更有经验的Web设计人员，正在回头温习基础知识，以便弄清‘语义XHTML和CSS’这类晦涩用语究竟是什么意思，那么这正是你所需要的书。”对此我并不赞同，因为本书对于他们来说可能实在浅了点。对于这类读者，《Web设计技术手册》（*Web Design in a Nutshell*, Jennifer Niederst Robbins著）这本书应该更适合一些。这本书也是由O'Reilly出版，国内已经引进了最新的第3版。

译者

2007年10月

# 序

1999年,我买了一本关于Web的书。这个相对新颖的世界引起了我的兴趣。那本袖珍的 *Rough Guide to the Internet* 中用了大约4页讲述基本的HTML。大约3小时后,我创建了一个网页,并把它链接到另一个网页。第一个网页很难看,但我很兴奋。那天稍晚些,我设法将它上传到一个域,并且意识到我创建了一个网站——一个实实在在的网站。

那时我幼稚地以为,为了让其他Web用户看到我的网页,我需要让家里的计算机一直开着!第二天,当我在办公室当着老板的面成功地调出我那个小小的网站时,我大为惊讶。

于是我决定再买一本书,书名叫 *Learn HTML in a Weekend*。那是一个相当长的周末。这本书和其他一些初级读物教了我许多知识,但其中大部分都是糟粕。我的代码中乱哄哄地充斥着font标签、框架和用于布局的表格。那时CSS还未成气候。

当初我们用HTML来完成所有艰难的工作,是因为我们不知道更好的办法。HTML是一门内容丰富而又宽容的语言,其早期的规范给对它的滥用留下了太多余地,那种繁琐、受限、笨拙的构建网站的方法正是根源于此。我能接受这个事实,但我不知道为什么现在许多新书还在继续向读者灌输各种拙劣、过时的技术,需知遵循Web标准本可做得更好、更轻松。

我一向关注人们如何学习构建网站,并且我也知道对于初学者来说这有点难以掌握。同样,我担心许多专业人员仍然在用粗制滥造的作品坑蒙客户。这正是我非常乐于推荐这本书的原因。David Schultz 和 Craig Cook 深知构建网站是一门技艺。凭借本书,他们把多年的实践经验浓缩为一份饶有趣味、认真编排过的参考资料呈现于读者面前,它讲述了可靠、强大的HTML、CSS、XHTML,甚至还有JavaScript,是一本理想的入门级综合读本。

你会看到有大量实用的例子可供借鉴。书中所有示例都能通过针对HTML Strict(你很快就会有更多了解)的验证,David和Craig已确保了所有方法都可以跨浏览器平台工作,并能应付以后因特网可能会发生的各种变化。作为一个极其注重代码质量的人,我对此印象尤为深刻。

你即将开始一次重大的冒险活动,但你手中已经攥着最好的地图,还有两名老练的向导牵着你的手。很快你就能到达自己的目的地,跟任何想听你谈谈自己对Web标准的心得的人滔滔不绝地聊个没完,并且纳闷那些老伙计为什么还在钻木取火。探险悍将们,这本书会把所有需要知道的东西都告诉你们。

Simon Collison

*Beginning CSS Web Development*<sup>①</sup> (Apress 2006) 的作者

---

① 本书中文版《CSS基础教程》已由人民邮电出版社出版。——编者注

# 前言

万维网只用了一段相对较短的时间就取得了长足进步。自20世纪90年代早期粉墨登场以来，Web已经从小圈子内的学术论文集合迅速发展成为一种成熟的、无处不在的媒体，与出版、广播和电视鼎足而立。Web是一个巨大的信息仓库，涵盖了从如天体物理学、古代哲学到寄居蟹的饲养等各种各样你能想象得到的主题。它已经成为许多人日常生活不可或缺的组成部分，并且成为现代商贸很多方面的平台。但就其本质而言，Web依旧不过是一种共享文档的手段。

本书将向你演示如何创建自己的文档并将其在Web上共享。你将逐渐熟悉HTML（HyperText Markup Language，超文本标记语言）的种种规则和构成。HTML是一种计算机语言，Web就构建在其基础之上。它很简单，其基本规则易学易用。作为一种工具，一旦你懂得如何使用它，能限制你的就只有你的想象力了。

就在不久以前，HTML的某些部分还在经常被误用，其规则很大程度上也总被人们忽视，这是因为我们没有别的选择或者仅仅是不知道还有更好的选择。但是在最近几年中，Web已经成熟了许多，我们也开始认识到有时遵从规则才是最佳途径。令人遗憾的是，产生于Web桀骜不驯的青春期的许多陈旧方法如今依旧广为使用。本书将帮助你避免过去人们犯过的一些错误，并为未来构建一个更好的Web。你将学习如何有效、可靠地使用HTML，使自己的文档更纯净、更富于意义，并能被尽可能多的人和设备访问。

如果你曾接触过Web，可能会听说过CSS（Cascading Style Sheet，层叠样式表），而且也许很想知道这是什么东西。简单地说，CSS是一种描述Web文档在视觉上应该如何被呈现的语言。它非常强大、灵活，而且酷毙了。但是，CSS本身就是一门丰富多彩的复杂语言，我们不可能在本书中论及其各方面。不过你很快就会知道，CSS与HTML直接相关，要想熟练运用CSS，你先得懂得标记代码。本书将让你初步了解CSS，并提供一些演示其用法的实用例子。我们会帮助你打下HTML的坚实基础，以此为起点，你才能更深入地钻研用CSS设计网页的技艺。

## 本书目标读者

本书面向任何想学习如何运用最新的最佳实践从头开始构建网页的读者。我们假定你熟悉因特网和万维网，因为如果你甚至未曾听说过“HTML”一词的话，你是不太可能拿起一本书名中就包含这个词的书的。除此之外，我们不再假定你已经具有Web设计或计算机程序设计方面的任何知识。在你学习本书的过程中，越往后接触到的内容会越有深度。但是不要害怕，这是一本写给初学者看的书，我们会陪你走过那些困难的地段。

即使你不是初学者，本书也许还是值得一读。如今构建网页的常用方法与仅仅几年前的已经大相径庭。最近一段时期内有太多东西发生了变化，因此如果你是一个更有经验的Web设计人员，正在回头温习基础知识，以便弄清“语义XHTML和CSS”这类晦涩用语究竟是什么意思，那么这正是你所需要的书。

## 本书组织方式

在此，我们展示一张路线简图，以便了解本书将带领你前往什么地方。本书开头两章介绍了开始创建自己的Web文档之前你必须了解的起码的基本知识。第3章至第10章是本书的主体部分，从中你将接触到HTML和XHTML的各种主题，并逐渐熟悉可供你使用的各种元素。在此过程中，你也将看到一些CSS技术的示例，可以把它们用于自己的网页的可视化设计。本书结束于第11章，那里有一个案例研究，它综合运用了许多你在前面几章中学过的技术，用XHTML和CSS从头构建一个实用的网站。

- 第1章从一个高层视角概述Web的运作机制，并说明为了创建XHTML文档，你需要一些什么东西。
- 第2章介绍了在创建Web文档和样式表时需要遵循的基本语法规则，为本书后面章节打下基础。
- 第3章介绍文档的head元素，解释了它为什么如此重要，并展示了可以放在其中的几种元素。
- 第4章研究如何在文档中添加内容，并使文本具有稳定、富有意义的结构。
- 第5章说明如何向网页中添加图像。它们可能传达有含义的信息，也可能仅仅是装饰。
- 第6章关注如何在文档中加入指向你的网站或因特网上其他地方的文档的链接。
- 第7章展示如何通过用表格组织复杂数据，把相关的信息组织到关联的行和列的集合之中。
- 第8章展示如何创建表单，以便访问者能够输入信息并与网站互动。
- 第9章更深入地研究CSS的使用，包括一些较高级的主题。在为自己的网页设计视觉样式时，你需要理解这些内容。
- 第10章概述JavaScript语言的基础知识。JavaScript可用于使网页变得更富动态性和互动性。
- 第11章把书中讲述的各种技术付诸应用，带领你一步步地创建一个实用的网站。

本书最后有4个附录供你参考。它们涉及的内容依次是XHTML 1.0 Strict、颜色名称和值、特殊字符和CSS浏览器支持状况。

## 本书排版约定

本书有大量CSS和XHTML代码示例。多数示例都出现在编了号的代码清单中，与正文分开。它们形如：

**代码清单1 代码清单示例**

```
<!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" lang="en">
  <head>
    <title>Just an Example</title>
  </head>
  <body>
    <p>Hello, world!</p>
  </body>
</html>
```

有时一行过长的代码会超出印刷页面的宽度限制，我们不得不把它折为两行。在此情况下，我们会使用符号➡来让你知道换行的目的仅在于适应页的版面，实际代码还是出现在单独的一行上。

我们会不时地添加一些与你刚读过的小节相关的注解、提示和警告。它们与正文有着截然不同的外观。如下所示：

---

**提示** 不要忽视这些额外的小花边。它们与当前主题有关，并且值得特别关注。

---

有时我们也许会稍稍偏离当前主题，谈点别的话题，尽管它与当前主题无关，但却提供了你需要知道的重要信息。为了不影响正文的流畅性，我们以如下方式给出这种补充信息：

**补充信息**

为了避免干扰主题，有些额外的信息被放在此处，以便更深入地探讨相关话题。

**下载代码**

本书中所有HTML和CSS代码都可以从Apress网站 (<http://www.apress.com>) 的源代码/下载区下载<sup>①</sup>。在下载并解压文件后，每一章的源代码都存放在各自的文件夹中，你可以单独将其抽出来以供闲时查阅。你也可在本书的附属网站<http://www.beginninghtmlbook.com>获取源代码。

**联系作者**

你可以通过电子邮箱[david.schultz@apress.com](mailto:david.schultz@apress.com)与David Schultz联系。

你可以通过Craig Cook的个人网站<http://www.focalculture.com>与他联系。

---

① 本书源代码也可从图灵网站 ([www.turingbook.com](http://www.turingbook.com)) 的本书配套网页下载。——编者注

## 致谢

撰写一本书不是一件小事。虽然封面上印的是作者的姓名，荣誉也主要归他们所有，但是这本书的问世离不开幕后的Apress的众多工作人员。我特别要感谢Chris Mills，是他与我接洽，并使我在项目的整个生命期中都保持着一份热情。非常感谢技术评审Gez Lemon，是他让我与标准保持一致并提供了宝贵的反馈意见。Elizabeth Seymour和Richard Dal Porto做了大量工作，使我的工作处于正轨，并帮我顺利通过了这一冗长沉闷的过程。感谢Apress的制作部门所做的最后修改以及把一部手稿变成印刷品的神奇工作。我也要感谢我的合著者Craig Cook为这本书带来的创意和做出的贡献。

最后，如果没有家人——我的妻子Kim和孩子Justin、Jessica、Crystal——的支持，我不可能完成这一长途旅程。

David Schultz

首先，如果说我还算有点见识的话，那是因为我站在了许多作家、艺术家、设计师、程序员、博客作者、布道者和专家的肩上，我要感谢所有这些人。下面这些人的作品和教导令我受益匪浅：John Allsopp、Douglas Bowman、Andy Budd、Dan Cederholm、Tantek Çelik、Joe Clark、Andy Clarke、Simon Collison、Derek Featherstone、Aaron Gustafson、Christian Heilmann、Jon Hicks、Molly Holzschlag、Shaun Inman、Roger Johansson、Jeremy Keith、Ian Lloyd、Scott McCloud、Eric Meyer、Cameron Moll、Keith Robinson、Richard Rutter、Dave Shea、Jeffrey Zeldman。此外还有其他许多人，他们的名字同样值得我在此引以为荣，但我必须将这份致谢限制在一页的篇幅之内。

我也要感谢所有那些为创建一个更好的Web而努力的普通工作者，他们的努力不是出于对钱财和名望的追求，而仅仅是因为他们热爱自己做的事并且在乎如何正确地做这些事。感谢每一个拥抱Web标准和可访问性、用自己的奇思妙想推进这一事业并开垦暗无天日的浏览器大战留下的焦土的人，是你们让Web变成一个更加宜居的地方。鞠躬致敬。

感谢每一个为把这本书变成现实而做出贡献的人。是Chris Mills与我共享烤肉并邀请我参加这个项目。是Gez Lemon以一种温和而残忍的坦诚指出我的错误和疏忽。是我的合著者David Schultz，承担了如此之多的艰苦工作。是Elizabeth Seymour、Richard Dal Porto、Grace Wong、Nicole Abramowitz、Kim Wimpsett、Laura Esterman，还有所有其他Apress的工作人员，对我在整个过程中常显拖沓的进度保持着一份耐心。

特别要感谢我的朋友Jolene、Jannycy和Bill。在写作过程中，每当我努力寻找恰当的词语时，我都以他们为假想的读者来进行考量。希望我能达到他们的要求。

对我的父母R.L.和Beverly我有一份道不尽的感激之情，是他们培养了我学习的渴望、创造的激情和传授的冲动。

Craig Cook

# 目 录

## 第1章 启动..... 1

### 1.1 因特网和万维网简介..... 1

### 1.2 什么是 HTML..... 2

#### 1.2.1 HTML 的发展..... 2

#### 1.2.2 一种语言, 多个版本..... 3

#### 1.2.3 一个版本, 三种风味..... 3

#### 1.2.4 验证文档..... 4

### 1.3 将内容与表现分离..... 5

### 1.4 处理 XHTML 和 CSS 文档..... 5

#### 1.4.1 选择 HTML 编辑器..... 6

#### 1.4.2 选择 Web 浏览器..... 6

#### 1.4.3 网站托管..... 7

### 1.5 URL 简介..... 7

#### 1.5.1 URL 的构成..... 7

#### 1.5.2 绝对和相对 URL..... 9

### 1.6 小结..... 9

## 第2章 XHTML 和 CSS 基础..... 10

### 2.1 标记代码的部件: 标签、元素和属性..... 10

#### 2.1.1 块级元素和行内元素..... 12

#### 2.1.2 嵌套元素..... 12

#### 2.1.3 空白符..... 13

#### 2.1.4 标准属性..... 13

#### 2.1.5 添加注释..... 14

### 2.2 XHTML 文档..... 15

#### 2.2.1 Doctype..... 16

#### 2.2.2 <html>元素..... 17

#### 2.2.3 其他部分..... 19

#### 2.2.4 文档树..... 19

### 2.3 CSS 基础..... 20

#### 2.3.1 CSS 规则剖析..... 20

#### 2.3.2 CSS 选择符..... 21

#### 2.3.3 具体性与层叠..... 24

#### 2.3.4 将样式表附加到文档..... 25

#### 2.3.5 层叠顺序..... 27

#### 2.3.6 !important..... 28

#### 2.3.7 编排 CSS 的格式..... 28

#### 2.3.8 CSS 注释..... 30

### 2.4 小结..... 30

## 第3章 前进..... 32

### 3.1 head 节..... 32

### 3.2 辅助性元素..... 33

#### 3.2.1 <base>..... 33

#### 3.2.2 <link>..... 34

#### 3.2.3 <meta>..... 35

#### 3.2.4 <script>..... 36

#### 3.2.5 <style>..... 37

#### 3.2.6 <title>..... 38

### 3.3 小结..... 39

## 第4章 添加内容..... 40

### 4.1 内容和结构..... 40

### 4.2 美妙的<body>..... 41

### 4.3 富有意义的部分..... 42

#### 4.3.1 <p>..... 42

#### 4.3.2 标题: <h1>、<h2>、<h3>、<h4>、 <h5>和<h6>..... 44

#### 4.3.3 <blockquote>..... 45

#### 4.3.4 <address>..... 47

#### 4.3.5 <pre>..... 48

### 4.4 列表..... 49

#### 4.4.1 <ul>..... 49

#### 4.4.2 <ol>..... 50

#### 4.4.3 <li>..... 51

|                                         |    |                     |     |
|-----------------------------------------|----|---------------------|-----|
| 4.5 定义列表                                | 52 | 5.4 背景图像            | 94  |
| 4.5.1 <dl>                              | 53 | 5.5 小结              | 100 |
| 4.5.2 <dt>                              | 53 | 第6章 链接到 Web         | 101 |
| 4.5.3 <dd>                              | 54 | 6.1 锚标签             | 101 |
| 4.6 短语元素                                | 55 | 6.2 使用<a>标签         | 103 |
| 4.6.1 <em>                              | 55 | 6.2.1 链接到其他文档       | 103 |
| 4.6.2 <strong>                          | 55 | 6.2.2 链接到非 XHTML 文档 | 105 |
| 4.6.3 <cite>                            | 56 | 6.2.3 链接到电子邮件地址     | 106 |
| 4.6.4 <q>                               | 57 | 6.3 把图像用作链接         | 107 |
| 4.6.5 <dfn>                             | 58 | 6.4 使用 ID           | 108 |
| 4.6.6 缩写: <abbr>和<acronym>              | 59 | 6.5 为锚标签添加 CSS      | 109 |
| 4.6.7 修改文档: <del>和<ins>                 | 60 | 6.6 创建图像映射          | 113 |
| 4.6.8 <bdo>                             | 61 | 6.6.1 <map>         | 114 |
| 4.6.9 编程: <code>、<kbd>、<samp><br>和<var> | 62 | 6.6.2 <area>        | 114 |
| 4.6.10 <br>                             | 63 | 6.7 小结              | 118 |
| 4.6.11 <hr>                             | 64 | 第7章 使用表格            | 119 |
| 4.7 多用途元素                               | 66 | 7.1 表格的基础知识         | 119 |
| 4.7.1 <div>                             | 66 | 7.1.1 <table>       | 120 |
| 4.7.2 <span>                            | 67 | 7.1.2 <tr>          | 121 |
| 4.8 嵌入外部内容                              | 68 | 7.1.3 <td>          | 122 |
| 4.8.1 <object>                          | 68 | 7.2 创建简单表格          | 123 |
| 4.8.2 <param>                           | 69 | 7.2.1 <caption>     | 125 |
| 4.9 表现性元素                               | 70 | 7.2.2 <th>          | 127 |
| 4.9.1 <i>和<b>                           | 70 | 7.3 表格的高级用法         | 129 |
| 4.9.2 <big>和<small>                     | 70 | 7.3.1 <tbody>       | 131 |
| 4.9.3 <tt>                              | 70 | 7.3.2 <thead>       | 132 |
| 4.9.4 <sup>和<sub>                       | 71 | 7.3.3 <tfoot>       | 133 |
| 4.10 特殊字符                               | 72 | 7.3.4 <colgroup>    | 136 |
| 4.11 用 CSS 控制内容的样式                      | 73 | 7.3.5 <col>         | 137 |
| 4.11.1 声明基础字体样式                         | 73 | 7.4 用 CSS 为表格添加样式   | 138 |
| 4.11.2 控制列表的样式                          | 78 | 7.4.1 添加边框          | 139 |
| 4.12 小结                                 | 80 | 7.4.2 对齐表格中的文本      | 140 |
| 第5章 使用图像                                | 81 | 7.4.3 为单元格添加衬距      | 142 |
| 5.1 数字图像的工作原理                           | 81 | 7.4.4 为表格添加背景       | 144 |
| 5.2 在内容中加入图像                            | 86 | 7.5 小结              | 146 |
| 5.2.1 <img>                             | 86 | 第8章 创建表单            | 147 |
| 5.2.2 上下文中的图像                           | 92 | 8.1 表单的工作原理         | 147 |
| 5.3 让文字环绕图像                             | 93 | 8.2 表单的组成部分         | 148 |

|        |                  |     |        |                            |     |
|--------|------------------|-----|--------|----------------------------|-----|
| 8.2.1  | <form>           | 148 | 10.4   | 高级主题                       | 221 |
| 8.2.2  | <input>          | 151 | 10.4.1 | 处理事件                       | 221 |
| 8.2.3  | <button>         | 160 | 10.4.2 | DOM                        | 225 |
| 8.2.4  | <select>         | 161 | 10.4.3 | 表单验证                       | 225 |
| 8.2.5  | <option>         | 163 | 10.5   | 小结                         | 228 |
| 8.2.6  | <optgroup>       | 165 | 第 11 章 | 综合运用                       | 229 |
| 8.2.7  | <textarea>       | 166 | 11.1   | 案例说明: Spaghetti & Cruft    | 229 |
| 8.3    | 组织表单             | 168 | 11.2   | 设计流程                       | 230 |
| 8.3.1  | <fieldset>       | 169 | 11.2.1 | 第 1 步: 明确目标                | 230 |
| 8.3.2  | <legend>         | 170 | 11.2.2 | 第 2 步: 设计信息架构              | 230 |
| 8.3.3  | <label>          | 172 | 11.2.3 | 第 3 步: 筹备模板                | 231 |
| 8.4    | 用 CSS 控制表单的样式    | 174 | 11.2.4 | 第 4 步: 构思设计方案              | 232 |
| 8.4.1  | 去掉 fieldset 的边框  | 176 | 11.2.5 | 第 5 步: 组装网站                | 234 |
| 8.4.2  | 对齐签条             | 177 | 11.2.6 | 第 6 步: 测试                  | 234 |
| 8.4.3  | 改变表单控件的字体        | 178 | 11.3   | 构建 Spaghetti & Cruft       | 235 |
| 8.5    | 小结               | 180 | 11.3.1 | 创建文档                       | 235 |
| 第 9 章  | 为文档添加样式: CSS     | 181 | 11.3.2 | 编写刊头部分的标记代码                | 236 |
| 9.1    | 使用外部样式表          | 181 | 11.3.3 | 编写主体内容区域的标记<br>代码          | 237 |
| 9.2    | 度量单位             | 183 | 11.3.4 | 编写导航栏的标记代码                 | 238 |
| 9.3    | 布局               | 183 | 11.3.5 | 编写口号和页脚的标记代码               | 238 |
| 9.3.1  | 容器               | 184 | 11.3.6 | 完成后的模板                     | 239 |
| 9.3.2  | 容器的大小和流动         | 187 | 11.4   | 用 CSS 设计 Spaghetti & Cruft | 240 |
| 9.3.3  | 定位容器             | 188 | 11.4.1 | 定义页面主体的样式                  | 241 |
| 9.4    | 背景               | 197 | 11.4.2 | 定义刊头的样式                    | 247 |
| 9.5    | 设置文本的样式          | 198 | 11.4.3 | 页面布局                       | 251 |
| 9.6    | 媒体类型             | 200 | 11.4.4 | 定义导航栏的样式                   | 254 |
| 9.7    | 兼容性              | 201 | 11.4.5 | 定义页脚的样式                    | 256 |
| 9.8    | 小结               | 201 | 11.5   | 组装页面                       | 256 |
| 第 10 章 | 客户端脚本编程基础        | 203 | 11.5.1 | 菜单页面                       | 257 |
| 10.1   | 什么是脚本编程          | 203 | 11.5.2 | 评论页面                       | 261 |
| 10.2   | 放置 JavaScript 代码 | 203 | 11.5.3 | 联系页面                       | 264 |
| 10.3   | JavaScript 语言    | 205 | 11.6   | 小结                         | 268 |
| 10.3.1 | JavaScript 语法规则  | 205 | 附录 A   | XHTML 1.0 Strict 参考        | 269 |
| 10.3.2 | 运算符和表达式          | 207 | 附录 B   | 颜色名称和值                     | 296 |
| 10.3.3 | 语句               | 209 | 附录 C   | 特殊字符                       | 303 |
| 10.3.4 | 循环               | 215 | 附录 D   | CSS 浏览器支持状况                | 307 |
| 10.3.5 | 函数               | 220 |        |                            |     |
| 10.3.6 | 数组               | 220 |        |                            |     |

**我**们将用一些关于因特网和万维网的常识来引导你步入本书。在你开始使用HTML和CSS来创建自己的网页时，这里打下的基础会很有用。本章并不是一个全面的概述，但它会让你掌握一些术语和概念，本书所有剩余部分都需要你熟悉它们。如果你用过网站或从事网站方面的工作已经有一段时间，觉得自己很了解Web，那么你可以跳到第2章去着手实践。

## 1.1 因特网和万维网简介

“因特网 (the Internet)”只不过是一个笼统的说法，指的是那个庞大的全球范围的计算机网络，那些计算机彼此联接在一起，并且能够发送和接收数据，以接近光速的速度在世界各地来回传送信息。自从一些非常聪明的人搞清如何让一部计算机与另一部计算机对话以来，它就已经以某些形式存在了将近半个世纪。因特网此后变得无处不在、无孔不入，对现代生活产生了极其广泛的影响，以至于很难想象如果没有它世界会是一个什么样子。

万维网 (the World Wide Web) 只是因特网的一个方面，就像大城市里一个熙熙攘攘的街区。它由保存在因特网上不同的计算机上的无数文件和文档组成，这些交叉引用、彼此连接着的文件和文档织成了一张世界范围的信息大网，万维网由此得名。在其相对较短的历史中，Web的发展已经远远超越了当初那种简单的文本文档的形式，同样的渠道如今承载着图像、视频和音频等多种信息，提供着引人入胜的互动体验。但就其本质而言，Web基本上还是一种基于文本的媒体，它使用的文本通常编码为HTML（稍后详述）。

可以访问Web的设备有很多：台式计算机、笔记本电脑、个人数字助理 (personal digital assistant, PDA)、移动电话、游戏机，甚至包括一些家用电器。无论是哪种设备，它都要转而借助于为解释HTML而设计的软件。这些程序在技术上被称为用户代理 (user-agent)，但更为人们熟知的名称是Web浏览器 (Web browser)。特别要指出的是，Web浏览器是用于对Web文档进行视觉化表达 (render) 的程序，而有些用户代理虽然会解释HTML，但并不显示它。

我们在本书中通常使用浏览器一词来指任何能够处理和表达HTML文档的用户代理。当我们使用图形化浏览器 (graphical browser) 一词时，所指的是用一种在视觉上有所强化的格式来呈现文档的浏览器，它支持所有颜色、具有样式的文本和图像。进行这种区分很重要，因为有些

Web浏览器并非图形化的，它们只用单纯、无样式的文本来表达文档，不使用图像。

浏览器或用户代理也被称为客户（client），因为它是请求和接受服务的一方。向客户提供数据的计算机，毫不奇怪，被称为服务器（server）。因特网上遍布着服务器，它们存储、处理数据，并应客户的请求将数据发送给它们。客户和服务器是链的两端，连接它们的是因特网。

## 1.2 什么是 HTML

要想把互相连接的数据文本片段编织成Web，就必须有某种用于建立这种连接的技术。这就是超文本（hypertext）的基础。使用超文本，一个文档中的一串文字可以直接链接到Web上某个地方的另一个文档。超文本标记语言（HyperText Markup Language, HTML）是一种计算机语言，用于将普通文本转化为活性（active）文本以供显示和在Web上使用，并为普通的无结构文本提供结构。人们需要这种结构帮助他们阅读。如果没有结构，单纯的文本将会汇合在一起，以至于无法把一串文字与另一串区分开来。

HTML由一些称为标签（tag）的经过编码的标记符组成，标签包围着文本片段、将其与其他部分区分开来，并且表明了所“标记”的文本的功能和用途。标签被直接嵌入普通文本文档，并在计算机软件处理该文档时得以解释。它们被称为“标签”，是因为它们的确就是标签。正如价格标签表明商品的价格、尸牌表明尸体的身份一样，HTML标签表明内容的某一组成部分的性质，并且提供关于它的重要信息。标签自身不会被显示，并且会被区别对待于它们所封装的实际内容。

HTML被刻意设计为一种简单、灵活的语言。它是一个免费、公开的标准，不属于、也不受控于任何公司或个人。不需要购买许可，也不需要使用特别的软件来创建HTML文档。任何人都可以自由创建和发布网页，Web能发展成为现在这样一种强大、影响深远的媒体正是得益于这种开放性。HTML的存在使我们都能自由、方便地共享信息。

但是，当你使用HTML创作文档时，你必须遵守特定的规则，因为HTML中的各种要素必须按特定方式组织起来才能正确发挥作用。这些规则由万维网协会（World Wide Web Consortium, 缩写为W3C）负责维护。W3C是一个非赢利性组织，它制定了用以构建Web的许多公开的技术标准，它们统称Web标准（Web standard）。对Web语言的标准化使得任何人，包括文档作者和设计解释这些语言的软件的人，都遵守同一套公认的规则。其意义类似于一套公认的语法和标点规则有助于你理解这个句子。

### 1.2.1 HTML 的发展

HTML最初出现在20世纪90年代早期，它基于已有的标准通用标记语言（Standard Generalized Markup Language, SGML），专为标记用于新生的万维网的文档而设计。自从诞生以来，HTML已经经历了多次修改和增强。新的特性被加入，同时也有一些特性因变得过时而被剔除出规范。标准规范中正式弃用一个特性的行动称为废弃（deprecation）。已废弃的特性应被逐步淘汰，新文档中要避免使用它们。

HTML的所有正式版本的技术规范都可以从W3C的网站（<http://www.w3.org>）免费获得。这

些规范往往不易阅读，因为它们本来就特别技术化，主要是为计算机科学家和生产Web用户代理的软件生产商而编写。但是这种标准对于Web的推广至关重要，它确保了网站在不同的浏览器和操作系统上都能一致地运作。Web的目标是“平台无关”和“设备无关”，只有遵守Web标准，才能实现这个目标。

在Web的早期，语言规范并不总能得到应有的严格遵守。彼此竞争的各种浏览器支持不同的特性，并且引入自己的一些非标准特性。这复杂化了当时的Web开发，Web作者们常常要针对不同浏览器为自己的网站创建多个版本。值得庆幸的是，现在已经没有这样做的必要了。如今的Web浏览器对标准的遵守比早先的浏览器一致得多，这促进了Web向其最终目标——真正大一统的媒体——进一步发展。

### 1.2.2 一种语言，多个版本

在HTML发展进化的过程中，新的语言版本不断发布以引入新的特性并废弃旧的特性。HTML的最初版本，1.0版，发布于1993年。1995年发布的2.0版以及紧接着在1996年发布的3.0版对它做了进一步改善和扩展。4.0版发布于1997年，在做了一些小（但是重要）的修改之后又作为HTML 4.01在1999年发布。这原本将是HTML语言的最终完整版本。2000年，一个名为eXtensible HTML或XHTML的新成员加入了这个团体，并被誉为未来的潮流。

XHTML是按照可扩展标记语言（eXtensible Markup Language, XML）更严格的规则对HTML进行的重新定义。XML是一种强大的语言，它允许Web作者创建自己的自定义标签。与XML不同，XHTML只提供了一套有限的预定义标签以供选用。XHTML与HTML 4.01相似，只是多了一些规定它应该如何撰写的规则，其当前版本为XHTML 1.0。XHTML 1.1和2.0已在开发之中，但在本书写作时还未被敲定为正式推荐标准。

在本书余下部分中，你将学习如何遵循XHTML 1.0创作自己的Web文档。尽管如此，你在本书学到的大多数东西同样适用于较早的——但依然充满活力的——HTML 4.01 语言。

---

**注解** 虽然HTML 4.01长期以来被当做HTML语言的“最终版本”，但是最近W3C启动了一个起草HTML 5规范的方案。本书写作时这个版本仍处于开发的早期阶段，还没有被公布。

---

### 1.2.3 一个版本，三种风味

好像HTML的各种不同版本还不够添乱一样，这些版本本身又有多个版本，每一种都有一些稍有不同的规则和特性。XHTML的3种“风味(flavor)”分别是严格的(Strict)、过渡的(Transitional)和框架集(Frameset)：

- ❑ XHTML 1.0 Strict的规则最严格。已废弃的特性被彻底禁用，只有严格地遵守规则的文档才是合式的。
- ❑ XHTML 1.0 Transitional比Strict略为宽松。它的合式文档允许存在一些过时的特性。这一变体用于实现从HTML的早期版本到XHTML 1.0的过渡。
- ❑ XHTML 1.0 Frameset仅应用于使用框架来控制网页布局的场合[框架(frame)是来自