

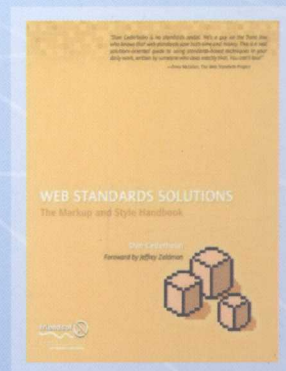
Web Standards Solutions

The Markup and Style Handbook

Web 标准实战

[美] Dan Cederholm 著
常可 译

- Amazon全五星图书
- 世界级Web大师力作
- 难得一见的实战手册，充满创意和技巧



人民邮电出版社
POSTS & TELECOM PRESS

The Turing logo, featuring the word "TURING" in a bold, sans-serif font with a small crown-like symbol above the "I".

TURING

图灵程序设计丛书

Web开发系列

Web Standards Solutions

The Markup and Style Handbook

Web 标准实战

[美] Dan Cederholm 著

常可 译

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

Web 标准实战 / (美) 锡德霍姆 (Cederholm, D.) 著;
常可译. —北京: 人民邮电出版社, 2008.5
(图灵程序设计丛书)
书名原文: Web Standards Solutions: The Markup and
Style Handbook
ISBN 978-7-115-17822-0

I. W… II. ①锡…②常… III. 网站—设计 IV.
TP393.092

中国版本图书馆CIP数据核字 (2008) 第033403号

内 容 提 要

本书是一部非常实用的讲述 Web 标准的著作。全书分为两部分: 第一部分讲述了与页面代码相关的话题, 第二部分则讲述了如何使用 CSS 来设定样式。书中的每一章都讲解了不同的 Web 标准概念的理论 and 实践, 并且对每一个问题都给出了多种解决方案。

本书适合中高级 Web 设计与开发人员阅读。

图灵程序设计丛书

Web 标准实战

-
- ◆ 著 [美] Dan Cederholm
 - 译 常 可
 - 责任编辑 傅志红 刘艳娟
 - ◆ 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>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
 - ◆ 开本: 800×1000 1/16
印张: 14.25
字数: 337 千字 2008 年 5 月第 1 版
印数: 1—5 000 册 2008 年 5 月河北第 1 次印刷
著作权合同登记号 图字: 01-2007-2128 号

ISBN 978-7-115-17822-0/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 *Web Standards Solutions: The Markup and Style Handbook* by Dan Cederholm, published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright © 2004 by Dan Cederholm. Simplified Chinese-language edition copyright © 2008 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。

对本书的赞誉

在使读者在阅读时始终兴趣盎然上，本书做得极其出色（技术类书籍往往难以做到这一点）。

——William Nichols, Slashdot (www.slashdot.org)

10分满分的话，本书至少得9分，在我看来，你应当毫不迟疑地把它放在你的书桌上。

——Dave Shea, mezzoblue (www.mezzoblue.com)

我经常一再重读本书中一些内容，甚至记住了它们所在的页码。本书一定能在你的工作书架中占据一席之地，你得仔细看管好，很容易不翼而飞的。

——Thomas Duff, Duffbert's Random Musings (www.twduff.com)

它是一本绝佳的参考手册和教程，在理论与实践之间架起了一座桥梁。

——D. Keith Robinson, Asterisk (www.7nights.com)

本书很合开发人员的口味，因为Dan Cederholm清晰地演示了如何嵌套、重用以及简化样式表和页面结构，以实现简洁的、易于维护的成品。

——Todd Dominey, Dominey Design (www.domineydesign.com) 的创始人
以及What Do I Know (www.whatdoiknow.org) 的站长

对于Web专业人员来说，Dan Cederholm的*Web Standards Solutions*必将是这个夏天甚至这一年度的最佳图书！

——Molly E. Holzschlag, molly.com (www.molly.com)

译者序

这是一本迟到很久的书。

它是*Bulletproof Web Design*的作者Dan Cederholm的成名作，面世于2004年夏天。如果没记错的话，那个时候国内的Web前端开发领域还处在IE6一统天下、Firefox初出茅庐、Web标准的概念刚刚引进的情况下。许多关于Web设计的书基本上就是Dreamweaver和FrontPage的教程。初夏的时候，一本叫做《网站重构》的书出版了。对于我本人而言，这算是第一次真正开始接触“Web标准”，我才了解到原来Web设计绝不是简单的、用可视化软件加上一点美术感觉就能胜任的工作，而是严谨的、注重细节的、艺术细胞和编码技能缺一不可的技术活。真正的Web设计师必须也是“代码高手”。于是我开始饥渴地寻找有关的学习资源，但当时国内的网站资源很少，而图书市场上除了《网站重构》之外可谓一片空白。《网站重构》一书着重介绍Web标准的理念、思想以及好处，偏向于理论宣讲，在实践方面有很多不足，只靠它没办法快速地上手，写出符合Web标准思想的页面。而那时国外已经有不少这方面的好书了，Dan的这本*Web Standards Solutions*就是其中之一。

本书的特点是非常容易上手，着眼于基本需求，一步一步的示例配以简明易懂的讲解，在某些需要特别留意的地方又不吝笔墨地反复强调。通过阅读本书，我学会了以符合Web标准思想的方式创建网页上的各种基本元素——标题、列表、表格、表单、超链接等的正确方法，还了解了CSS的不同引入途径、动态切换样式、精简页面代码、创建面向打印机的样式、用图片替换文本等高级技巧。虽然本书写于2004年，但其中的许多技术即使在现在也毫不过时，并且还有许多人不了解或不会纯熟地运用。

对于已经了解了Web标准理论的设计师，本书是极好的实战参考手册。

放眼今天的国内技术图书市场，关于Web标准的书籍已经有如雨后春笋，这要感谢Firefox等浏览器的进步和Web 2.0的风潮。“页面不符合Web标准”已经是技术和产品落后的一个表征了。

我在2004年时就想翻译出版本书，可惜未能如愿。3年后承蒙人民邮电出版社图灵公司的编

辑老师们的信任，我得以有机会实现曾经的愿望。

感谢我的家人和朋友们的支持；感谢作者Dan奉献给我们好书；感谢所有爱好并积极推动Web标准的朋友们！希望这本书能进入你们的阅读清单。

序

你手里拿着的是一本处方书。设计师Dan Cederholm通过清晰的示例和切题的文字，展示了如何运用Web标准来创建漂亮的、轻量级的并具有亲和力的用户界面。

Dan并不是要展示基于标准的Web设计的创意或者商业案例。早已经有人这么做了。坦率地说，如果你主动拿起本书并翻阅一下，就可能已经知道了基于标准的设计所能提供的亲和力、持久性以及商业优势。你不需要另一份概览或者Web标准如何如何重要之类的话，而需要实用的、能立刻上手的、模块化的（component）知识和观点，这些正是本书要讲述的。

使用易懂的、自然的语言——好的网站都应该有——Dan讲述了通用的网站成分，例如页面分区和导航。使用他在SimpleBits.com上所开创的教学方法，Dan展示了如何利用Web标准让这些通用页面成分更加容易创建；在面对老板或客户的紧急需求变更时更易于修改；而最重要的是易于使用。

下面举一个例子来说明本书的内容组织方式，以及它为什么值得你掏腰包并花时间阅读：

你所设计的站点的首页需要一个三栏的布局，而内容页面则又要求用两栏布局。以前的老办法是创建两个不相关的页面模板（还是表格布局的）。而由万维网协会（World Wide Web Consortium, W3C）所推荐的、基于标准的设计师们所实践的新办法是用最少的、富含语义的XHTML代码来创建页面结构，用层叠样式表（Cascading Style Sheet, CSS）来实现页面布局 and 外观。

作为有经验的Web设计师，你可能会自然地想到需要写两个不同的XHTML模板和两个不同的样式表以产生两栏和三栏的布局效果。但是本书中展示了：一个XHTML结构和一个样式表就足够了。在两种布局样式之间的切换很简单，只需给<body>标签加上一个类属性。

本书中充满了类似上例那样的创意和技巧——它们可以提高你的工作效率，简化任务难度，同时激发你的创意。其中一些技术由Dan自己发明，还有一些则来自蓬勃涌现的、由基于标准的Web设计先锋们开发的最佳实践。你需要了解这一切。而最好的了解方式就在你的手中。祝你阅读愉快。

Jeffrey Zeldman

*Designing With Web Standards*一书的作者

前言

本书的写作目的是向你提供一些“武器”，它们能够给你的项目提供基于Web标准的解决方案，让你在运用HTML和CSS的时候能做出更好的决策。贯穿每一章，我们都将比较各种常见的Web设计方法，试图弄明白为什么其中一种方法比另一种要更好。通过这些比较，我们将为项目找到最适合的工具。

但首先，我们要确认我们处于同一页上——书中含有缩写词、代码段以及可能比较陌生的概念。我们先来谈谈Web标准。

什么是 Web 标准

引用万维网协会（World Wide Web Consortium: www.w3.org/Consortium/）的话：

“万维网协会（W3C）创立于1994年10月，其宗旨是：通过发展通用的协议来带领万维网发挥出其最大潜力，推动它的进化并确保它的互操作性（interoperability）。W3C拥有大约400家来自世界各地的成员组织，其对Web发展所做的贡献已经为世人所肯定。”

W3C的创始人是Tim Berners-Lee。该协会起草了今天建立Web所依据的Web标准规范。我们的注意力重点放在其中两种标准上：可扩展超文本标记语言（XHTML），它具有HTML 4.01的语义和XML的语法；以及层叠样式表（CSS），它用来设计Web页面的样式。

为什么要用 Web 标准

昨天我买了一幅窗帘。我在去商店之前先量了一下窗户的尺寸。我从货架上选了一件23英寸的买回家，它非常合适。

去年，我和妻子一起买了一台新的洗碗机。我们把旧的洗碗机从厨柜里移出，并订购了一台新的。当新的到来后把它放进去，它也非常合适。

我这里想说的是：因为产品遵循某些（外形）标准（standard），使得居家布置变得非常容易。我可以走进商场，买一根下水道用软管，它一般不会接不上。我也可以买个新的门把手，十有八

九它不需要做任何调整就能装在门上用了。

因为预先的规定，标准的尺寸让人们建造和维修房屋更加轻松。当新的房主想要翻新或维修房子时，标准让这项工作变得更容易。

当然，事情并非总是这样。在20世纪以前，修建的房子并没有都遵循标准。并不是说不按标准修建的房子就不是好房子——只是说如果要翻新、修葺或者维修这些房子，需要花更多的工夫。

很多人都买旧房子，然后再装修。当繁重的装修工作完成时，房主便因标准的尺寸而受益，今后的维修也会简单很多。

本书并不是关于房子的。但前面的这些比喻同样可以用于Web世界——通过在页面中使用标准，维修工作将变得容易得多。其他设计师和开发者可以更容易地参与和理解页面的构造和样式。

从历史上看，设计师和开发者们一直都依赖于臃肿的页面代码来实现设计效果，今天也遍布这样的网站。嵌套了三层之深的表格以及用透明的GIF图片获得像素精度的布局是沿用了多年的模范做法。但是随着现代浏览器对标准支持的程度越来越好，将精简的结构化页面代码和CSS相结合的能力已经十分强大，兼容Web标准绝不意味着难看而乏味的设计效果。

潮流正在改变，那些意识到Web标准的优势的人与Web设计和开发社区中的其他人相比，会超前一大步。这是事物的未来走向。

通过理解和运用Web标准，可以获得下面这些好处：

- ❑ **更少的页面代码：**更少的代码意味着更快的页面加载速度，也意味着更强的服务器负载能力，这反过来也意味着更少的存储空间占用和带宽花销。
- ❑ **内容和表现的更大程度的分离：**通过用CSS来控制站点的外观设计，样式修改和重构变得更容易。修改一个样式表就能立刻改变整个网站。
- ❑ **更好的亲和力（accessibility）^①：**Web标准使得我们能支持最大可能数量的浏览器和设备。站点内容可以被任何浏览器、手机、PDA或者那些使用辅助性软件的设备所访问。
- ❑ **前向兼容性（forward compatibility）：**通过使用Web标准创建写页面，可以保证它们在未来仍然可读。

上述任何一个好处都可以是使用Web标准的充分理由，本书将会告诉你如何（how）丢掉那些坏习惯，教给你创建富有吸引力的、基于标准的设计方案的经验和技巧。

为什么要用 XHTML

本书中的页面代码示例都使用XHTML，即可扩展的超文本标记语言编写。从技术上讲，

^① 也译为可访问性。——编者注

XHTML是HTML 4以XML形式的重生。这是什么意思呢？嗯，你已经熟悉HTML了，对吧？XHTML也是一样的——只是多了一些规则。

再一次引用W3C的内容（www.w3.org/TR/xhtml1/#xhtml）：

“XHTML家族是因特网进化的下一步。通过今天转移到XHTML上，内容开发者们能够走进XML的世界，获得它现有的全部好处，同时还能确保他们的内容具备对过去和未来的兼容性”。

这是个很好的出发点——对未来的兼容性。现在用XHTML编写页面，我们就能进一步保证这些页面在未来的浏览器和设备中仍然可用。

XHTML还被设计为可由最大数量的浏览器、设备和软件所识别。不管用什么来读取，XHTML代码都比用其他语言编写的代码更有可能被正确地解析。

不过，XHTML有一些规则。

在XHTML的世界里，要写出合法的代码，需要遵循更加严格的规则。所有的标签和属性名都必须用小写字母，属性值必须用双引号引起来，所有标签最后都必须是闭合的。有一些规则继承自XML。但这些规则是有好处的。

除了对未来的兼容性，通过坚持严格的规则，书写正确合法的XHTML，设计师和开发者们可以更容易地调试代码（在多个作者维护同一页面时尤其有帮助），而且页面在支持标准的浏览器中也更有机会被正确渲染。整本书中我们的示例都将采用XHTML。

结构化代码

整本书中你可能会发觉我经常使用“语义”（semantic）这个词。我还会经常使用“结构化代码”（structured markup）。它们是可互换的。所谓语义，是指我们要尽可能使用那些表示“意义”（meaning）的标签，而不是表示“外观”的。你是在处理一个列表吗？那就用表示列表的代码。这是一个数据表格吗？那就用相符合的结构。

通过用语义化代码构建页面，我们将更好地实现内容和表现相分离，这样页面就能更好地理解——这是你所希望的——支持更多种类的浏览器和设备。

如同我前面提到过的，以前Web设计师们依赖于嵌套表格和占位图片来实现像素精度的布局。<table>标签用来标识页面上的每个成分，导致了大量的多余代码——还导致页面实际上无法用文本浏览器、小屏幕设备或者辅助性软件来浏览。这些臃肿的东西从过去到现在一直在阻塞Web。

贯穿本书的每一章，我都会说明语义化代码如何帮助页面变得更简洁、更富亲和力、更容易用CSS设定样式。

本书的起源

它开始得很单纯。我在我的个人网站上发布了一些简单的测验题。一个多项选择题，每个答案都实现相同或者相似的结果。为什么某种方法要比其他方法更好？这是测验问题的真正答案。

测验的目的是揭示每种方法的优点和缺点，要注意到，即使许多方法使用的是合法的代码，它们仍然不一定是最佳方案。读者们可以评论并留下自己的意见，通过这些讨论，我前面提到的“武器”就出现了。如果我们能够理解为什么（why）使用页面标题以及合适的列表是重要的，那么我们就能将这些东西应用到自己的日常项目中去。

还值得提及的是，我并非试图规定构建某些页面成分只能用一种方式——如同Web设计中的每样东西，要实现相同或相似的结果，有多种方式。使用最适合当前任务的——但通过理解不同方法的优点和缺点，你再遇到这些情况时就会做出最佳决策。

本书结构

本书分为两大部分，第一部分涵盖了和页面代码相关的话题，第二部分则着重于CSS。每一章都解答一个专门的问题，通常都会展示能实现同样结果的多种方法。我们会详细介绍每种方法，了解它们各自的优缺点。在许多章节的最后部分还有一节“实例秀”，进一步研究了与本章内容相关的更高级的页面代码和CSS话题。

我希望你喜欢本书——现在让我们开始吧。

致谢

我向下面这些在成书过程中给予我帮助的人致以诚挚的谢意：

Chris Miller 很早就参与到我的写作中，给予我很多指导并确保整体进展。

Drew McLellan 给了我极好的建议和指导，并付出了很大努力。

Apress/friends of ED 出版社的许多人都为本书做了许多事，特别是 Sofia Marchant, Ami Knox 和 Ellie Fountain。

Jeffrey Zeldman，没有他的话我不可能会写出本书，他为 Web 标准所做的贡献超过了任何人。

Douglas Bowman 给了我无懈可击的设计灵感，他还证明了 CSS 布局在大型商业网站上也能工作得非常漂亮。

Dave Shea 种植了那个美丽的花园（CSS Zen Garden），证明了 CSS 可以实现我们想要的任何效果。

Jason Kottke 提出了关于“spark”的问题。

SimpleBits 的访客们提供了很有价值的讨论，使我产生了撰写本书的想法。

Eric Meyer、Christopher Schmitt、Tantek Celik、Molly Holzschlag、Todd Dominey、Mike Davidson、Ryan Carver、Dan Rubin、D. Keith Robinson、Mark Pilgrim、Joe Clark、Craig Salla、Nick Finck、Owen Briggs、Simon Willison、Ian Lloyd、Dan Benjamin 和其他许多人，他们在 Web 标准社区里线上线下的贡献帮助了成百上千的人，我就是其中之一。

Web Standards Project 的成员们，他们让所有的 Web 设计师和开发者都一直受益良多。

我在 Fast Company and Inc. 时的同事们——特别是 Rob Roesler，他给我提供了很好的机会和支持；David Searson，我从他那里学到了太多太多；同样还有 Bob Joyal；Paul Maiorana 容忍了我的旅行强迫症；还有 Daigo Fujiwara、Paul Cabana、Nick Colletta、Heath Row、Irina Lodkin、Carole Matthews、Becca Rees、Alex Ashton、Peter Wilkinson——以及 Linda Tischer，是她将我带进了 FC 公司。

我的家人和朋友，以及最重要的——我的爱妻 Kerry——感谢她对我无比的支持。

当然，还有你们，我亲爱的读者们。感谢你们的阅读！

目 录

第一部分 重新审视 HTML

第1章 列表.....2

1.1 逛商店.....2

1.2 测验.....3

1.2.1 方法 A: 用
来分隔.....3

1.2.2 折行问题.....3

1.2.3 方法 B: 项目符号.....4

1.2.4 方法 C: 更加接近.....5

1.2.5 方法 D: 包装的喜悦.....6

1.3 内容回顾.....6

1.4 实例秀.....7

1.4.1 去掉项目符号.....7

1.4.2 创建更酷的项目符号.....8

1.4.3 可导航的列表.....9

1.4.4 迷你选项卡形状.....12

第2章 标题.....14

2.1 构造文档标题的最佳方式是什么.....14

2.1.1 方法 A: 富有语义.....14

2.1.2 方法 B: <p>与的组合.....15

2.1.3 方法 C: 样式和内容.....16

2.2 内容回顾.....18

2.3 实例秀.....18

2.3.1 简洁的样式.....18

2.3.2 加背景.....19

2.3.3 可变换的图标.....21

2.3.4 方便地更新.....21

2.3.5 变色龙效果.....22

2.4 小结.....23

第3章 表格是邪恶的吗.....24

3.1 完全的表格形态.....24

3.2 人人可有的表格.....25

3.3 添加摘要.....27

3.4 表头.....28

3.5 表头和数据的关系.....29

3.6 使用 abbr 属性.....31

3.7 <thead>、<tfoot>和<tbody>.....31

3.8 表格是邪恶的吗.....33

3.9 实例秀.....33

3.9.1 创建网格.....33

3.9.2 压缩空白.....34

3.9.3 除 IE/Mac 之外的版本.....35

3.9.4 腾出空间.....35

3.9.5 自定义表头外观.....36

3.9.6 带背景图片的表头.....37

3.9.7 给 ID 指定图标.....39

3.10 小结.....41

第4章 引用文本.....42

4.1 方法 A: 缺乏语义.....42

4.2 方法 B: 类的演出.....43

4.3 方法 C: <blockquote>是最好的.....43

4.4 内容回顾.....44

4.5 实例秀.....44

4.5.1 刨根问底的 cite.....45

4.5.2 行内引用.....45

4.5.3 为<blockquote>设定样式.....46

4.6 小结.....51

第5章 表单.....52

5.1 创建表单时有哪些选择.....52

5.1.1 方法 A: 使用一个表格.....52

5.1.2 方法 B: 没有表格, 却变得

拥挤.....53

5.1.3 方法 C: 简单并且更有亲和力	54	8.1.1 方法 A: 无序的顺序	92
5.2 内容回顾	58	8.1.2 方法 B: 有序列表	92
5.3 实例秀	59	8.2 构造一组词条和解释的最佳代码是什么	94
5.3.1 神奇的 tabindex	59	8.2.1 方法 A	95
5.3.2 为频繁访问的表单设定 accesskey	60	8.2.2 方法 B	95
5.3.3 给表单设定样式	61	8.3 内容回顾	98
5.4 小结	67	8.4 实例秀	98
第 6 章 、及其他短语元素	68	8.4.1 标识各个部分	98
6.1 表现化与结构化	69	8.4.2 自定义编号	99
6.1.1 为什么和比和<i>要好	69	8.4.3 将编号加到 CSS 里	99
6.1.2 那么呢	70	8.4.4 最终效果	100
6.1.3 只要粗体或者斜体就行了	70	8.5 小结	101
6.1.4 粗体加斜体	72	第 9 章 精简 HTML 代码	102
6.2 内容回顾	73	9.1 使用 Web 标准构建网站时如何精简 HTML 代码	102
6.3 实例秀	73	9.1.1 子代选择符	103
6.3.1 短语元素	73	9.1.2 方法 A: 愉快地使用类	103
6.3.2 <cite>设计	74	9.1.3 方法 B: 自然选择	104
6.3.3 <abbr>和<acronym>	76	9.1.4 不必要的<div>	106
6.3.4 <code>	78	9.1.5 方法 A: 愉快地使用<div>	107
6.3.5 <samp>	78	9.1.6 方法 B: 丢掉<div>	107
6.3.6 <var>	79	9.1.7 其他示例	107
6.3.7 <kbd>	79	9.1.8 内容回顾	108
6.4 最后的短语	79	9.2 实例秀	108
第 7 章 锚点	80	9.2.1 未加工的 HTML 代码	108
7.1 当指向页面特定部分时, 如何构建锚点才是最佳方案	80	9.2.2 添加样式	109
7.1.1 方法 A: 空的名字	80	9.2.3 自定义项目符号	110
7.1.2 方法 B: 都在名称中	81	9.2.4 添加边框	111
7.1.3 方法 C: 去掉名称	82	9.3 小结	113
7.1.4 方法 D: 全合一	84	第二部分 样式点滴	
7.2 内容回顾	84	第 10 章 应用 CSS	116
7.3 实例秀	85	10.1 怎样将 CSS 应用到文档上	116
7.3.1 title 属性	85	10.1.1 方法 A: <style>元素	117
7.3.2 给链接设定样式	86	10.1.2 方法 B: 外部样式表	118
7.4 收锚启航	90	10.1.3 方法 C: @import	118
第 8 章 更多列表	91	10.1.4 为多个样式表结合使用方法 B 和 C	120
8.1 构造编号列表的最佳方式	91		

10.1.5 两种级别的样式	121	13.1.9 段落缩进	169
10.1.6 拥抱层叠	122	13.2 内容回顾	170
10.1.7 方法 D: 行内样式	123	第 14 章 图片替换	171
10.2 内容回顾	124	14.1 怎样通过 CSS 用图片替换文本	171
10.3 实例秀	125	14.1.1 不存在完美的解决方案	171
10.4 小结	128	14.1.2 使用它, 但要小心	172
第 11 章 打印样式	129	14.2 方法 A: Fahrner 图片替换	172
11.1 如何对打印指定样式	129	14.2.1 HTML 代码	172
11.1.1 媒介类型	130	14.2.2 额外的标签	172
11.1.2 两种定向方式	130	14.2.3 CSS 代码	173
11.1.3 方法 A: media 属性	130	14.2.4 指定背景	173
11.1.4 方法 B: @media 或者 @import	131	14.2.5 优点	174
11.1.5 允许多个值	131	14.2.6 缺点	174
11.1.6 分隔屏幕和打印样式	132	14.2.7 权衡利弊	175
11.1.7 建立打印样式表	132	14.3 方法 B: Leahy/Langridge 图片替换 (LIR)	175
11.2 内容回顾	137	14.3.1 HTML 和 CSS 代码	175
第 12 章 CSS 布局	138	14.3.2 框模型的痛苦	176
12.1 如何用 CSS 构造两栏布局	138	14.3.3 缺点	176
12.1.1 方法 A: 浮动侧栏	139	14.4 方法 C: Phark 方法	176
12.1.2 方法 B: 双浮动	144	14.4.1 HTML 和 CSS 代码	176
12.1.3 方法 C: 浮动内容	145	14.4.2 还是不完美	177
12.1.4 方法 D: 绝对定位	148	14.5 内容回顾	177
12.2 内容回顾	154	14.6 实例秀	178
12.3 实例秀	155	14.6.1 标志切换	178
12.3.1 框模型问题	155	14.6.2 示例	178
12.3.2 伪栏	158	14.6.3 一对标志	179
12.4 小结	160	14.6.4 CSS 代码	179
第 13 章 设定文本样式	161	14.6.5 恢复链接	180
13.1 怎么把超文本变酷	161	14.6.6 结果	180
13.1.1 改变 Times 字体	162	14.7 具有亲和力的可翻转图片选项卡	181
13.1.2 改变行间距	162	14.7.1 问题	181
13.1.3 都是一家人	163	14.7.2 解决方案	182
13.1.4 字符间距调整	164	14.7.3 HTML 代码: 一个列表统领 一切	182
13.1.5 段首字母下沉	165	14.7.4 一张图片, 三种状态	182
13.1.6 文本对齐	167	14.7.5 CSS: 魔法产生的地方	183
13.1.7 字形变换	168	14.7.6 结果	184
13.1.8 小型大写字母	169	14.7.7 为什么要使用它	185

14.7.8 文本不能缩放	185	15.2.2 标明各个部分	192
14.8 兼容性	185	15.2.3 魔法 CSS	193
14.9 小结	186	15.3 内容回顾	194
第 15 章 为<body>设定样式	187	第 16 章 下一步	195
15.1 两栏, 有时又是三栏	187	16.1 下一步去往何处	195
15.1.1 页面和样式结构	189	16.1.1 组织和出版物	195
15.1.2 <body>有类	190	16.1.2 富有影响力和启发性的博客	197
15.1.3 不仅可用于栏	190	16.1.3 书籍	199
15.2 “你在这里”	191	16.2 结束语	200
15.2.1 导航列表	191	索引	201