

新手学

The Commercial Website
Structure and Layout Design
Entry to the Master

DIV+CSS

商业网站布局

从入门到精通

实战
案例版

基于 Dreamweaver 和 Photoshop 常用软件, 按照从入门到精通的写作思路, 结合由简单到复杂的多个典型网站实例, 使读者不但可以学会应用多种工具制作精美网页, 还能掌握现在最流行的DIV+CSS网页布局与美化技能。

朱印宏 编著

- 重点强调技术灵活应用
海量网页设计制作经验、网页布局 and 美化实用技巧全放送
- 精选最新最热实战案例
精美博客、企业、生活、社区和购物类网站现场指导制作
- 轻松掌握混合应用技术
详细讲解CSS与JavaScript、Ajax和XML的混合应用技巧
- 针对网页设计师真实需求
杜绝伪代码和伪案例、强调工作团队各个环节的技术衔接

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

新手学 DIV+CSS 商业网站布局 从入门到精通

实战
案例版

朱印宏 编著

内 容 简 介

本书系统地讲解了 CSS 的基础知识和实际运用技术,通过大量实例对 CSS 进行深入浅出的分析,主要包括 CSS 的基本语法和概念,系统讲解 CSS 布局模型、CSS 安全性、CSS 重用性、CSS 合作开发、CSS 商业开发规范。着重讲解如何用 CSS+DIV 进行网页布局,注重实际操作,使读者在学习 CSS 应用技术的同时,掌握 CSS+DIV 的精髓。本书还详细讲解了 5 个网站综合布局案例,指导读者制作符合 Web 标准的网页,帮助读者进一步巩固所学到的知识,提高综合应用的能力。在综合实战过程中,每章都会穿插讲解一项 CSS 应用技术,如文字、图片、背景、表格、表单和菜单等网页元素的样式设计技法,

本书内容翔实、结构清晰,并注意各个章节与实例之间的呼应和对照,既可作为 CSS 初学者的入门参考书,也适合作为中高级用户进一步学习的实用教材。

图书在版编目(CIP)数据

新手学 DIV+CSS 商业网站布局从入门到精通:实战案例版 / 朱印宏编著. — 北京:中国铁道出版社, 2014. 1

ISBN 978-7-113-17354-8

I. ①新… II. ①朱… III. ①网页制作工具 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2013)第 222100 号

书 名: 新手学 DIV+CSS 商业网站布局从入门到精通(实战案例版)
作 者: 朱印宏 编著

策 划: 武文斌
责任编辑: 张 丹
编辑助理: 吴伟丽
责任印制: 赵星辰

读者热线电话: 010-63560056
封面设计: 多宝格

出版发行: 中国铁道出版社(北京市西城区右安门西街 8 号 邮政编码: 100054)

印 刷: 北京鑫正大印刷有限公司

版 次: 2014 年 1 月第 1 版 2014 年 1 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 36.25 字数: 858 千

书 号: ISBN 978-7-113-17354-8

定 价: 69.00 元

版权所有 侵权必究

凡购买铁道版图书,如有印制质量问题,请与本社读者服务部联系调换。电话:(010) 51873174

打击盗版举报电话:(010) 51873659

前言

CSS 已经成为现代网页设计中必不可少的关键技术。它看似简单，但真正精通 CSS 绝非易事。在使用 CSS 开发网站时，会遇到形形色色棘手的问题，而解决方案又五花八门，往往让使用者感觉千头万绪，不知从何着手。

本书将最有用的 CSS 技术汇总在一起，还总结了 CSS 设计中的各种设计技巧和最佳实践，讨论了解决各种实际问题的技术，填补了一直以来 CSS 图书的空洞枯燥以及直白乏味。正因如此，本书可以作为网页设计人员必读的经典书目。

本书涵盖了 CSS 全部技术，详细介绍了各个 CSS 属性以及属性之间的相互作用，并指导你如何避免一些常见的错误。无论你是一位经验丰富的 Web 创作人员，还是知之甚少的新手，都可以把这本内容翔实的 CSS 参考资料放在手边。

本书特色

基础知识系统缜密

本书系统讲解 CSS 技术，细致入微地剖析 CSS 核心技术和主要内容，讲解循序渐进，配合大量实例，帮助读者奠定坚实的基础知识，做到知其所以然。

知识与实战相结合

本书在讲解每一个知识点时，都会提供实战案例，重点讲解设计过程，以及如何灵活应用技术。案例讲解结合作者长期的网页设计制作经验和教学经验，使读者真正做到学以致用。

深度剖析布局要领

本书在每个网站案例中都重点介绍了用 CSS+DIV 进行网页布局的方法和技巧，配合经典的图文解说，帮助读者掌握 CSS 最核心的应用技术。



寓教于乐、生动有趣

CSS 有些方面比较晦涩,如定位模型和特殊性,即使是有经验的 CSS 开发人员也会遇到一些问题。这是因为大多数 CSS 开发人员都是靠自学的,他们从各种文章和别人的代码中学习经验,而没有全面理解 CSS 规范。这也不奇怪,因为 CSS 规范本身十分复杂,它的目标读者是浏览器厂商而不是网页开发人员。因此对于初学者来说,初学 CSS 会存在一定的障碍,本书语言通俗易懂,知识和技术与实例相融合,更容易让读者阅读和学习。

本书内容

本书前 4 章讨论基本的 CSS 概念和知识,帮助你轻松入门。你将学习如何建立代码结构和添加注释,了解 CSS 语言的细节以及与 HTML 完美结合的方法。你也许已经掌握了其中的许多内容,但是可能会发现自己有遗漏或理解不充分的地方。因此,前 4 章是不错的 CSS 入门材料,它可以帮助你重温已经了解的知识。

介绍了基本知识之后,后面 5 章讨论 CSS 核心技术,比如网页布局、CSS 安全性、CSS 重用性、CSS 合作开发、CSS 商业开发规范,以及进行纯 CSS 布局。每一章都由浅入深,最后讨论比较复杂的示例。在这几章中,你将学习如何创建更安全、优化、富有扩展性的 CSS 代码。一般情况下,会先展示传统技术,然后说明如何用 CSS 制作出同样的效果。

浏览器 BUG 是许多 CSS 开发人员最头疼的问题之一,所以本书中的所有示例都着眼于创建跨浏览器的技术。此外,本书还用一章内容讨论 BUG 和 BUG 修复。在这一章中,你将全面学习 BUG 捕捉技术,学会在 BUG 出现之前就发现并解决它。

最后 7 章是真正的“大餐”。在这部分将本书讨论的各种技术组合成七个精彩的实例来研究。因此,你不但会学习这些技术的工作原理,而且会看到如何将它们用在实际项目中。每章先以 CSS 一个技术点开始,帮助读者从一个技术预热开始,然后通过对这些网站的策划、草图设计、效果图设计、切图、结构重构到样式重现,详细讲解整个制作过程,由浅入深地介绍了如何将最初的想法变成精美的网页。即使是初学者也可以通过这种过程训练,轻松掌握 CSS+DIV 布局方式,制作出精美的网页并搭建出功能强大的网站。

本书读者

本书适合系统学习 HTML 和 CSS 基础知识的读者阅读。无论你是刚刚接触 CSS 设计,还是已经开发纯 CSS 站点好几年了,书中都有适合的内容。如果你已经使用 CSS 一段时间了,但还没有达到专家级水平,那么你将能够从本书获得最大的收益。本书还提供了各种实用的建议和示例,可以帮助你精通现代 CSS 设计。

为了给读者提供更多的学习资源,本书提供了很多参考链接,许多书中无法详细介绍的问题都可以通过这些链接找到答案。因为链接地址会因时间而有所变动或调整,所以在此说明,这些链接地址仅供参考,本书无法保证所有地址是长期有效的。

本书约定

- “HTML”是指 HTML 和 XHTML 这两种语言。
- 除非特别声明，本书中提到的“CSS”是指 CSS 2.1 规范。
- IE 早期版本浏览器是指 Windows 的 IE 6 和 IE 7 等。
- 标准（现代、主流）浏览器是指最新版的 Firefox、Safari、Opera、IE 8 以及以上版本。
- 本书中的所有 HTML 示例都应该嵌套在一个有效文档的<body>标签中，同时，CSS 包含在内部样式表中。偶尔为了尽量简短，HTML 和 CSS 放在了同一个代码示例中。但是在真实的文档中，这些代码需要放在各自的位置上才能正常工作。
- 最后，对于包含重复数据的 HTML 示例，我们不会列出每一行，而是适时地使用省略号表示部分代码。
- 本书中的图例可能会与读者实际环境中的操作界面有所差别，这可能是由于操作系统平台、浏览器版本等不同而引起的，在此特别说明，读者应该以实际情况为准。

关于我们

本书由朱印宏编写，参与资料整理及编写的还有王佳佳、常才英、袁祚寿、袁衍明、张敏、袁江、田明学、唐荣华、毛荣辉、卢敬孝、刘玉凤、李坤伟、旷晓军、陈万林、陈锐、钱佩林、苏敬波、冉东林、杨龙贵、张炜、王慧明、涂怀清、卢国才、苏恢定、司成向、胡体清、陈宗亮、徐清银、周秀成、颜昌学、王幼平、冉原洲、李经键、胡厚成等，在此对大家的辛勤工作表示衷心的感谢。

由于编者水平有限，书中难免会有疏漏之处，恳请广大读者提出宝贵意见，也希望大家能够把个人的意见、建议或问题发送到 wwb_beijing@163.com，以便我们与您进行沟通和交流。

编者

2013 年 10 月

目录

第 1 章 初学者必知的常识

1.1 回眸 Web 标准发展之路	2
1.1.1 第一个吃螃蟹的是谁?	2
1.1.2 设计师的表格情结	2
1.1.3 归去来兮	3
1.1.4 传统布局与标准布局比较	5
1.1.5 网页设计前瞻	8
1.2 Web 标准: 开发者必知技术规范	9
1.2.1 网页结构	9
1.2.2 网页布局	10
1.2.3 网页行为	10
1.3 CSS 布局的基本思路	10
1.3.1 语义含义与实例解释	11
1.3.2 给网页设计师推荐的网站: CSS 禅意花园	11
1.3.3 CSS 布局的基本思路及实例	12
1.4 CSS 其实不难	14
1.5 制作第一个 XHTML+CSS 页面	15

第 2 章 CSS 语言基础 1

2.1 认识 CSS	22
2.1.1 为什么学习 CSS	22
2.1.2 编辑简单的 CSS 样式	23
2.1.3 编辑简单 CSS 样式具体操作步骤	23
2.1.4 CSS 与浏览器	25
2.1.5 CSS 样式表文件	26
2.1.6 CSS 注释	27
2.2 CSS 类型	28
2.2.1 行内样式解释及实例展示	28
2.2.2 内嵌式解释及实例展示	30
2.2.3 链接式解释及实例展示	33
2.2.4 导入样式解释及实例展示	34
2.2.5 各种方式的优先级实例展示	35



2.3 CSS 样式结构	36
2.4 CSS 基本选择器	37
2.4.1 标记选择器解释与实例展示	37
2.4.2 类别选择器解释与实例展示	39
2.4.3 ID 选择器解释与实例展示	43

第3章 CSS 语言基础 2

3.1 复合选择器	46
3.1.1 包含选择器解释及实例展示	46
3.1.2 子选择器解释及实例展示	48
3.1.3 相邻选择器解释及实例展示	50
3.1.4 属性选择器解释及实例展示	51
3.1.5 伪选择器解释及实例展示	55
3.1.6 集体声明解释及实例展示	57
3.2 层叠与继承	59
3.2.1 CSS 的层叠性应用实例解析	59
3.2.2 层叠性实例测试具体操作步骤	62
3.2.3 CSS 的继承性图解及实例	63
3.2.4 CSS 继承与层叠的运用实例	64

第4章 XHTML 重构基础

4.1 认识网页结构	69
4.1.1 XHTML 文档	69
4.1.2 定义文档类型	70
4.1.3 DTD 文档类型	70
4.1.4 声明命名空间	70
4.1.5 XHTML 语法	71
4.1.6 XHTML 元素	72
4.1.7 XHTML 常用结构元素: div	73
4.1.8 XHTML 常用结构元素: span	74
4.1.9 XHTML 常用结构元素: h1、 h2、h3、h4、h5 和 h6	74
4.1.10 XHTML 常用结构元素: p	75
4.1.11 XHTML 常用结构元素: ul、 ol、li、dl、dt 和 dd	75
4.1.12 XHTML 常用结构元素: table、 tr 和 td	76
4.1.13 把握 XHTML 结构的语义性	76

4.1.14 构建语义化页面操作	77
4.1.15 案例实战: 构建一个符合语义 的页面结构	78
4.2 DIV 和 Span 应用	80
4.2.1 比较 DIV 和 Span	80
4.2.2 案例实战: 使用 DIV	83
4.2.3 案例实战: DIV 嵌套	86
4.2.4 案例实战: 使用 Span	89
4.3 ID 和 Class 应用	91
4.3.1 使用 ID	91
4.3.2 比较 ID 和 Name	93
4.3.3 使用 Class	95
4.3.4 比较 ID 和 Class	97

第5章 CSS 基本模型

5.1 认识 CSS 盒模型	100
5.1.1 为什么要提出盒模型	100
5.1.2 盒模型结构模拟	101
5.1.3 盒模型的宽和高	102
5.1.4 盒模型并列间距	105
5.1.5 盒模型垂直间距	108
5.1.6 空盒模型	111
5.2 盒模型的边框	112
5.2.1 定义边框宽度	113
5.2.2 定义边框颜色	113
5.2.3 定义边框样式	114
5.2.4 分析内联元素的边框	117
5.3 盒模型的外部空隙	119
5.3.1 恰当选择边界值	119
5.3.2 警惕元素边界重叠	125
5.3.3 内联元素的边界	131
5.4 盒模型的内部空隙	132
5.5 IE 解析与盒模型的误解	135
5.5.1 IE6 元素浮动时边界错位	135
5.5.2 让父元素自适应子元素高度	136
5.5.3 IE6 子元素底边界不被解析	137
5.5.4 IE6 的子元素溢出	138

第 6 章 CSS 布局基础和实践

6.1 进一步理解盒模型	142
6.1.1 盒模型的显示类型	142
6.1.2 盒模型的特殊性: 包含块	145
6.1.3 构建 XHTML 代码模块	145
6.1.4 包含块的嵌套和实例	147
6.1.5 IE 浏览器在解析多层包含时 出现的问题	148
6.2 CSS 布局模型概述	149
6.2.1 流动模型	150
6.2.2 层模型	150
6.2.3 浮动模型	150
6.3 流动模型详解及其应用	151
6.3.1 流动布局基本特征	151
6.3.2 相对定位流动	153
6.4 浮动模型详解及其应用	156
6.4.1 浮动布局 5 个基本特征 与实例解析	156
6.4.2 清除浮动	161
6.4.3 浮动嵌套	165
6.4.4 混合嵌套	166
6.4.5 实战混合布局: 调整左右栏之间 的空隙	168
6.4.6 实战混合布局之调整上下栏之间 的空隙	172
6.4.7 深入混合布局	177
6.5 层模型详解及其应用	182
6.5.1 认识网页定位	182
6.5.2 定位参照坐标	183
6.5.3 定位的边距	187
6.5.4 相对定位	189
6.5.5 案例实战: 混合定位	190
6.5.6 定位元素层叠	191
6.6 布局实战: 高度自适应	195
6.7 布局实战: 伪列布局	198
6.7.1 设计思路	198
6.7.2 设计方法	199

第 7 章 设计更安全的 CSS

7.1 关于网页浏览器	203
7.1.1 Mozilla 系列浏览器	203
7.1.2 IE 内核浏览器	204
7.1.3 Safari 内核浏览器	204
7.1.4 Opera 内核浏览器	204
7.1.5 Chrome 内核浏览器	204
7.1.6 国内浏览器市场行情	204
7.1.7 多版本 IE 测试实现	205
7.2 认识 CSS 安全中的几个概念	205
7.2.1 什么是 CSS Hack	205
7.2.2 设计 CSS Hack 和 Filter 方法	206
7.2.3 CSS Filters 技术汇总站点	206
7.2.4 显示模式	208
7.3 过滤样式表	210
7.3.1 IE 条件语句	211
7.3.2 @import 规则	212
7.4 过滤样式	214
7.4.1 !important	214
7.4.2 下划线属性名	215
7.4.3 转义属性名	215
7.4.4 *html 选择符	216
7.4.5 属性选择符	216
7.4.6 子对象选择符	217
7.4.7 相邻选择符	217
7.4.8 转义选择符	217
7.4.9 注释反斜杠	218
7.5 隐藏样式	218
7.5.1 隐藏单个声明	218
7.5.2 隐藏多个声明	220
7.5.3 隐藏规则	220
7.5.4 IE7+ 浏览器不支持的过滤器	221
7.5.5 推荐使用的过滤器	222
7.6 网页不兼容怎么办	223
7.6.1 W3C CSS 验证服务	223
7.6.2 Web Developer 工具插件	225
7.6.3 代码隔离与验证	226

7.7 常见网页设计问题与修复	227
7.7.1 盒模型错误	227
7.7.2 双倍显示错误	228
7.7.3 多出 3 像素错误	228
7.7.4 高度不适应问题	231
7.7.5 多余字符错误	233
7.7.6 定位错误	234
7.7.7 捉迷藏问题	235
7.7.8 部分区域丢失	237
7.7.9 百分比计算错误	239
7.7.10 项目符号丢失	241
7.7.11 盒模型溢出问题	242
7.8 HTML 默认样式与浏览器解析差异	244
7.8.1 HTML 4 默认样式	244
7.8.2 浏览器默认样式	247

第 8 章 设计可重用的 CSS 与实例

8.1 设计可重用的结构	251
8.1.1 可重用的结构设计思路：我的网页结构该怎么规划	251
8.1.2 可重用的结构设计思路：初学者思维定式	252
8.1.3 可重用的结构设计思路：搭架基本框架	252
8.1.4 让结构更符合 SEO	255
8.2 设计固定宽度的单栏页面	256
8.2.1 单列版式的设计方法	256
8.2.2 实现单列设计	257
8.2.3 案例实战：设计固定宽度的单列页面	260
8.2.4 案例实战：单列固定版式布局	261
8.2.5 案例实战：重现江南水乡意象画之结构	265
8.2.6 案例实战：重现江南水乡意象画之布局	267
8.3 设计弹性宽度的单栏页面	271
8.3.1 案例实战：设计页面基本结构	271
8.3.2 案例实战：设计 CSS 布局	275
8.4 设计双栏页面	278

8.4.1 两列版式的基本结构	279
8.4.2 设计固定与自适应宽度兼容版式	281
8.4.3 设计宽度自适应版式	283
8.4.4 设计双浮动兼容版式	286
8.4.5 设计两栏固定版式	287
8.4.6 案例实战：黑色风暴	289
8.5 设计三栏页面	295
8.5.1 三列版式的基本设计方法	296
8.5.2 绝对定位布局的解决方法	297
8.5.3 设计液态版式	299
8.5.4 设计固定版式	302
8.5.5 设计一列液态两列固定版式	304
8.5.6 设计二列液态一列固定版式	306
8.5.7 案例实战：山鹰飞翔	308

第 9 章 CSS 团队合作与开发

9.1 统筹 CSS 文档	316
9.1.1 根据页面类型统筹	316
9.1.2 根据功能模块统筹	317
9.1.3 根据标签类型统筹	317
9.1.4 根据设备类型统筹	317
9.1.5 根据代码规模统筹	317
9.2 合理安排样式顺序	318
9.3 科学定义属性	321
9.3.1 按字母顺序定义	321
9.3.2 按主次关系定义	322
9.3.3 优先定义	322
9.4 严谨命名 CSS 样式	323
9.4.1 计算机编程命名三法	323
9.4.2 约定 CSS 样式的名称	324
9.4.3 如何实现 CSS 样式的严谨命名	324
9.5 养成 CSS 代码缩写习惯	327
9.5.1 CSS 常用属性简写	327
9.5.2 列表和背景样式缩写	329
9.5.3 颜色值缩写	330
9.5.4 字体属性缩写	330
9.6 设计 CSS 代码格式	331
9.6.1 CSS 代码格式一	331

9.6.2	CSS 代码格式二.....	332
9.6.3	CSS 代码格式三.....	333
9.6.4	快速格式化 CSS 代码.....	334
9.7	写好 CSS 注释.....	335
9.7.1	CSS 注释的方法.....	335
9.7.2	防止注释对源代码的破坏.....	337
9.7.3	清除 CSS 注释.....	339

第 10 章 CSS 规范化商业开发

10.1	栅格化设计.....	341
10.1.1	感性认识 CSS 栅格设计.....	341
10.1.2	商业网站的栅格设计.....	343
10.2	抽象化设计.....	346
10.2.1	应用 CSS 类样式.....	346
10.2.2	CSS 的 11 种通用类样式.....	347
10.2.3	默认样式.....	351
10.2.4	特殊类样式.....	352
10.3	CSS 代码的可维护性.....	353
10.3.1	类名和 ID 名可读性.....	353
10.3.2	CSS 代码版式规则.....	354
10.3.3	样式编码顺序.....	355
10.4	CSS 样式的可扩展性.....	356
10.4.1	继承 CSS 样式.....	356
10.4.2	CSS 样式的包含关系.....	358
10.4.3	样式表的分类、组织和管理.....	360
10.4.4	兼容主流浏览器.....	362
10.5	规范化网页布局实战.....	363
10.5.1	案例实战: Elements CSS.....	363
10.5.2	案例实战: Blueprint CSS.....	367

第 11 章 博客类网站布局与设计

——字体和文本样式

11.1	博客类网站功能预览.....	376
11.2	字体样式与设计.....	376
11.2.1	设置字体: 字体介绍.....	376
11.2.2	设置字体: 通用字分类及实例.....	377
11.2.3	设置字体时注意事项.....	379
11.2.4	设置大小 CSS 语法.....	379

11.2.5	设置大小: 使用绝对关键字.....	380
11.2.6	设置大小: 使用相对关键字.....	380
11.2.7	设置大小: 使用相对单位.....	381
11.2.8	设置大小: 使用百分比.....	382
11.2.9	设置大小: 使用磅.....	383
11.2.10	关于默认值.....	383
11.2.11	案例实战: 设计灵活、可读 的网页字体.....	383
11.2.12	设置字体颜色.....	384
11.2.13	设置字体样式.....	385
11.3	文本样式与设计.....	387
11.3.1	设置文本缩进.....	388
11.3.2	设置文本对齐.....	390
11.3.3	设置文本行距.....	393
11.3.4	设计分栏文本.....	394
11.3.5	设置文本字距.....	396
11.3.6	控制文本行的显示方式.....	397
11.3.7	设置文本空白显示效果.....	401
11.3.8	控制文本显示方式和流动方向.....	402
11.3.9	转换文本大小写.....	406
11.3.10	设计修饰性文本.....	407
11.4	案例实战.....	409
11.4.1	产品策划.....	409
11.4.2	画板和设计.....	410
11.4.3	切图和输出.....	413
11.4.4	网站重构.....	414
11.4.5	网站布局实现.....	417

第 12 章 企业类型网站布局与设计

——列表结构与样式

12.1	企业网站预览.....	424
12.2	列表结构和样式.....	424
12.2.1	列表的基本结构.....	424
12.2.2	设计列表样式.....	425
12.2.3	案例实战: 设计列表项流动显示.....	428
12.2.4	自定义列表符号.....	430
12.3	列表布局综合实践.....	433
12.3.1	设计灯箱图片新闻.....	433
12.3.2	设计选项卡效果.....	436

12.3.3	设计网页皮肤	438
12.3.4	设计网页皮肤: 构建列表结构并准备 素材	439
12.3.5	设计网页皮肤: 设计皮肤	440
12.3.6	设计网页皮肤: 动态控制皮肤	441
12.3.7	设计网页皮肤: 记忆皮肤	443
12.3.8	设计网页皮肤: 恢复皮肤	444
12.3.9	设计滑动门效果	445
12.4	案例实战	448
12.4.1	产品策划	448
12.4.2	画板和设计	449
12.4.3	切图和输出	451
12.4.4	网站重构	453
12.4.5	网站布局实现步骤	454

第 13 章 生活分类信息网站布局

与设计——超链接与

导航菜单样式

13.1	生活分类信息网站预览	461
13.2	超链接结构和样式	461
13.2.1	认识 CSS 伪类选择器	461
13.2.2	定义超链接	462
13.2.3	设计超链接样式	463
13.2.4	设计不同的超链接样式	464
13.3	设计导航菜单样式	466
13.3.1	设计水平菜单: 普通式	466
13.3.2	设计水平菜单: 图像式	467
13.3.3	设计垂直菜单	469
13.3.4	设计多级菜单: 平行式多级菜单	471
13.3.5	设计多级菜单: 垂直式多级 菜单	474
13.3.6	获取更多的导航菜单	478
13.4	案例实战	479
13.4.1	产品策划	479
13.4.2	画板和设计	480
13.4.3	切图和输出	482
13.4.4	网站重构	484
13.4.5	网站布局实现	485

第 14 章 购物类型网站布局与设计

——表单结构与样式

14.1	设计表单结构	490
14.1.1	表单基本框架	490
14.1.2	定义表单域	491
14.1.3	定义输入域	492
14.1.4	定义选择域	494
14.1.5	定义文本区域	495
14.1.6	定义辅助表单域	496
14.1.7	改善表单的视觉效果	501
14.2	设计表单样式	505
14.2.1	设置表单字体样式	505
14.2.2	设置表单边框样式	507
14.2.3	设置表单背景样式	510
14.3	案例实战	514
14.3.1	产品策划	514
14.3.2	画板和设计	516
14.3.3	切图和输出	518
14.3.4	网站重构	520
14.3.5	网站布局实现	522

第 15 章 社区类型网站布局与设

计——表格结构与样式

15.1	设计表格结构	533
15.1.1	设计基本结构	533
15.1.2	对列进行分组	534
15.1.3	对行进行分组	537
15.1.4	合并单元格	538
15.1.5	对齐单元格	539
15.2	设计表格样式	540
15.2.1	定义表格边框样式	540
15.2.2	设计方便浏览的表格样式	544
15.3	案例实战	553
15.3.1	产品策划	553
15.3.2	画板和设计	554
15.3.3	切图和输出	556
15.3.4	网站重构	558
15.3.5	网站布局实现	560

Chapter 1

初学者必知的常识



本章导读

Web 标准是什么？这是一个简单的问题，同时也是一个复杂的话题。如果在网上搜索一下，相信读者会看到不同的介绍。

笔者自从开始工作，一直研究 Web 标准，到现在已经很长时间，这期间越来越真切地感受到 W3C 组织所推荐的 Web 设计标准对于网页设计师的重要性和影响力，所以一直守规守矩推崇 W3C 所提倡的网页标准，同时本书也遵循着这样的规范进行讲解。

评价一种 Web 开发技术优劣的标准只有一个，那就是看这种技术能否在最恰当的时间和最恰当的地点，以最恰当的方式，为最需要信息的人提供最恰当的信息服务。

在讨论 Web 标准布局之前，纵览今日我国的 Web 技术发展，虽然还存在不同的声音，但普及和使用 Web 标准及 CSS 技术已是一个不争的事实。

回眸 Web 标准发展之路

国内信息技术的发展似乎总慢于西方半个节拍，其他不说，单就 CSS 技术也是一样。CSS 第一个正式规范早于 1996 年就已经发布，也就是说 1996 年 W3C 组织（World Wide Web Consortium，万维网联盟）正式推出 CSS 1.0 版本技术标准，到 1998 年 CSS 2.0 版本又面世。

而我们的 Web 设计师们在新世纪钟声之后，才慢慢感觉到 CSS 技术的先进性，并开始在网站中小心翼翼地进行尝试。

W3C 组织是一个专门负责网络标准制定的非赢利组织，如 HTML、XHTML、CSS、XML 等标准就是由 W3C 制定的。根据 W3C 官方网站的介绍，W3C 会员包括生产技术产品及服务的厂商、内容供应商、团体用户、研究实验室、标准制定机构和政府部门，W3C 会员一起协同工作，致力在万维网发展方向上达成共识。

关于 W3C 组织的更多信息，请读者访问 W3C 官方英文网站（<http://www.w3c.org>），或者通过万维网联盟中国香港办事处（成立于 1998 年 11 月，地址在香港科技大学），以及万维网联盟中国办事处（<http://www.chinaw3c.org/>，成立于 2006 年 4 月，地址在北京航空航天大学）获取有关 W3C 的中文介绍。

1.1.1 第一个吃螃蟹的是谁？

CSS 的中国实践与普及当推《CSS 权威指南》一书的引进，相信很多早期的 Web 设计师都看过这本书。时间大致应该锁定在 2001 年 5 月，此后，前卫的设计师们开始在网页中局部尝试 CSS 技术。

2004 年 10 月闪客帝国（现已关闭）开始用 Web 标准对自己的网站进行重构，当时在业界产生了不小的轰动，因为这是第一个“吃螃蟹”的人。对于一个商业网站来说，使用未经普遍实践的技术是存在很大风险的，所幸他们很顺利地走过来。

用当时网易学院采访边城浪子的话说：“网站重构是今后很多网站要面临的问题，标准化永远不会是个错误，我们知道我们早晚会走这一步。长痛不如短痛，所以，我们就下定决心。”

闪客帝国网站于 2005 年完成了全部 CSS 布局，作为国内第一个大型应用案例，虽然在设计思想和技巧使用上不是最佳案例，但其在 CSS 布局应用上带来了 many 宝贵经验。

1.1.2 设计师的表格情结

表格在网页布局中应用已经有很多年，由于多年的技术发展和经验积累，Web 设计工具功能不断增强，使表格布局在网页应用中达到登峰造极的地步。

一个最典型的例子就是网页设计师可以直接使用图像编辑器画图、切图，最后再由图像编辑器全自动生成表格布局的页面，像这样提供完整而又完善的一条龙服务的网页设计工具实在很多，如设计师比较熟悉的 Adobe Photoshop、ImageReady 和 Fireworks 等。

这些工具的使用都很简单,学习周期短、制作速度快,而且设计的页面效果都很漂亮。如果你是一位富有艺术感的读者,相信设计出来的页面必定很吸引眼球。

传统表格布局的快速与便捷加速了网页设计师对于页面创意的激情,而忽视代码的理性分析,甚至刚学几天的初学者都敢说出“我可以设计出最完美的网页”,表格布局就这样被神话。

表格为谁而生?

提出这个疑问决非偶然,一直以来我就在不断探寻和思考表格的渊源。假设我们从网页设计这个相对独立的领域跨入另一个陌生的财会世界,你会大悟表格的真正使命。

“表格在手,一清二楚”,这是财会人员的口头禅,记得以前看过一本书,书中介绍一企业高级白领,她的最大才干就是擅长设计表格,针对不同的工作、不同的岗位,设计出适合不同工作和岗位的表格,让不同工种和岗位的员工能将各工作更有序更高效地完成,并将管理成本降得更低。何为高级,原来是能够设计出各种表格的白领。

也许你会自以为擅长表格设计的人当属网页设计师,其实这种想法是错误的,随手翻阅高级管理人员案头的形形色色表格,你才明白什么是真正的表格,表格的真正价值所在。如果当你进入公司管理阶层运作时,才发现若能为公司根据需求不断设计出行之有效的各类表格,那才是很重要的工作。

表格能够使繁复的数据分门别类,让人一目了然地清楚数据内在的整体情况,它比许多文字说明都要简明、有用得多。

例如,如何将客户的不同模具的未付款帐单理清,要知道每一次签订制作的模具是分期付款的,同一客户的不同模具,因签订合同的时间不同,加上其他诸多因素等,其应付款和未付款的期数是各不相同的,只有用表格的形式,才能很清楚地表达不同模具应付款和未付款及所有模具总未付款情况。

走出传统的纸制表格再来看看与网页毗邻的电子表格技术。提起电子表格,你一定会想到微软的 Excel,但你不会想到 VisiCalc。

1979 年面世的 VisiCalc 是历史上第一个电子表格软件,是它直接奠定了电子表格软件的基础,VisiCalc 的发明人 Dan Bricklin 绝对是一位天才(个人主页: <http://www.bricklin.com/>),仅用 16KB 的代码就改变了整个计算机产业,直接推进了个人电脑的发展。

从界面上看,VisiCalc 完全左右现代电子表格软件的发展。最早的 VisiCalc 是在 Apple II 系统上运行,因此给苹果公司带来了巨额利润。后来 Mitch Kapor 创办 Lotus 公司,推出 Lotus 1-2-3,电子表格得到更完善的发展。

如今 Lotus 1-2-3 电子表格软件也消失在用户的视野,提起电子表格大家都只知道 Excel,或者国内金山公司的电子表格。但不管电子表格软件如何发展,它们最终都是为了数据管理服务的,谁也不会想到用表格来进行设计,或用表格来排版布局。

1.1.3 归去来兮

实际上,表格的功能不是用来进行布局的。最初的网页没有今天这么复杂,仅显示文本或几个简单的图像,网页文档从上而下自然流动分布,不需要考虑版式设计问题。

后来随着网页内容的丰富,图像、声音、动画等多媒体不断充斥网页,网页内容不断膨胀,同时用户对于网页视觉提出了更高要求,于是如何把传统印刷中的版式技术转移到网页设计中来就成为了一个比较紧迫的需求。

由于表格不仅可以控制单元格的宽度和高度，而且还可以嵌套，多列表格还可以把文本分栏显示，于是就有人试着在表格中放置其他网页内容，如图像、动画等，以打破比较固定的网页版式。

而网页表格对无边框表格的支持为表格布局奠定了基础，用表格实现页面布局慢慢就成为一种设计习惯。现在网上的大多数页面都是使用表格来完成布局的，即使在 CSS 不断普及的情况下，这种状况也很难在短时间内得以改观。

表格布局是不标准的，用 W3C 制定的规范来说，表格的目的是用来显示数据的，而不是来完成布局，错把表格当布局缘于当时 Web 技术的缺乏和对标准需求的乏力。

为什么时至今日才会出现这么多人大声疾呼表格布局的非标准问题呢，为什么有这么多设计师迫切地使用 XHTML+CSS 来重构自己的网页呢？

其实在表格布局之初，就有专家提出异议，而且那时 CSS 也已经被制定并公布于众，一套较完善的网页标准技术也已经被推向市场，但应者甚寡。

现在 Web 技术从 1.0 跨入 2.0 时代，表格布局已经不能再适应新技术发展趋势了，同时用户对于互联网的需求上升到一个新的层次。如今的 Web 世界，用户体验是上帝，在民意需求的强势之下，标准占据了上风，连微软这样的巨无霸也不得不臣服于 W3C 所提倡的标准之下。

什么是 Web 1.0，什么是 Web 2.0 呢？

Web 2.0 在过去的一年里也许还是一个新的名词，曾几何时它像网上核弹一样爆发了，并以不可阻挡之势燃烧了整个互联网，其热度不亚于当年的超女，又曾几何时它悄悄地走进了我们的生活，从陌生走向了熟悉，从概念走向了应用，今天，Web 2.0 构成了我们网络生活不可缺少的一部分，大家都在讨论、研究和实践 Web 2.0。

目前对什么是 Web 2.0 并没有很严格的定义，一般来说 Web 2.0（也有人称之为互联网 2.0）是相对 Web 1.0 的新的一类互联网应用的统称。Web 1.0 的主要特点在于用户通过浏览器获取信息，Web 2.0 则更注重用户的交互作用，用户既是网站内容的消费者（浏览者），也是网站内容的制造者。

Web 1.0 到 Web 2.0 的转变，具体地说，从模式上是单纯的“读”向“写”、“共同建设”发展。所以互联网下一步，是要让所有的人都忙起来，用全民力量共同织出贴近生活的网。

到目前为止，对于 Web 2.0 概念的说明，通常采用 Web 2.0 典型应用案例介绍，加上对部分 Web 2.0 相关技术的解释，例如，博客（BLOG）、播客、威客、掘客、RSS、百科全书（Wiki，也叫维客）、网摘、社会网络（SNS）、P2P、即时信息（IM）等。

Web 2.0 对于网页结构和布局提出了更高要求，这不仅仅是为节省几行代码、减少带宽的问题，实际上今天的互联网对于带宽已经不是主要发展障碍。要实现 Web 的交互与共建，就必须允许用户可以操纵网页，而这些都必须建立在完全标准的网页结构之上，以及更加灵活的布局方式。

显然，Web 2.0 的这种要求与表格布局是格格不入的，如果你透过精美的表格页面，会吃惊地发现，用切图生成的网页代码如同天书，后期的维护和更新竟成了一场噩梦。

例如，某网千龙伊氏女性频道（<http://yesee.qianlong.com/>）的一个栏目，这是一个典型的用切图生成的页面。如果查看它的源代码，你会发现里面存在大量的冗余代码，网页结构与表现混杂在一起，要想在源代码中查找信息是非常困难的，更别说让用户参与网页共建之中了。

1.1.4 传统布局与标准布局比较

为帮助读者更好地理解表格布局与标准布局的孰优孰劣，我们不妨结合一个案例进行详细分析。也许使用网页编辑工具切图可能产生很多代码冗余，这时读者会想到：我用手写代码实现表格布局还不行吗，难道还会出现代码冗余？

问题不是这么简单，读者可以看看一个简单的空白布局模板，如图 1.1 所示，它是一个三行三列的典型网页布局。

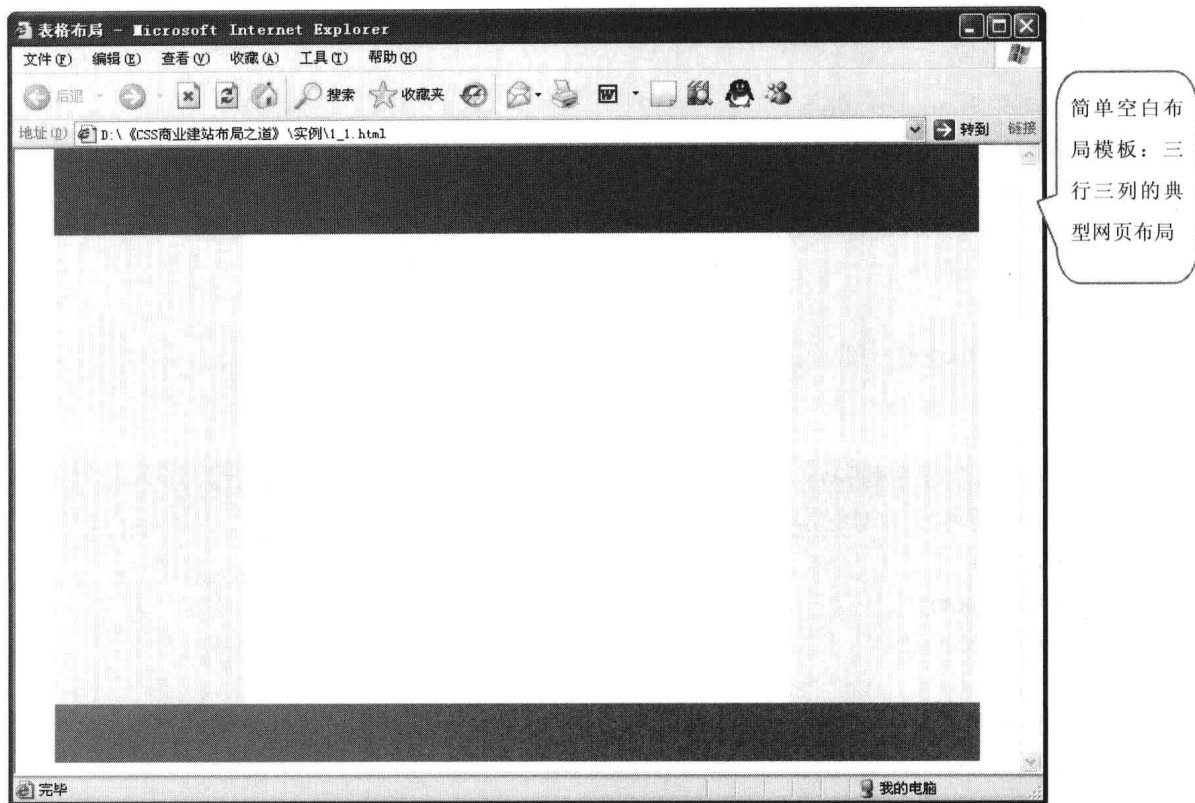


图 1.1 一个简单的布局模板

下面尝试用表格布局和 XHTML+CSS 标准布局来实现它，亲身体验二者的异同。

1. 实现上图布局效果使用表格布局代码

<!--[三行三列表格布局]-->

```
<table width="778" height="514" border="0" align="center" cellpadding="0" cellspacing="0">
  <tr>
    <td height="74" colspan="3" valign="top" bgcolor="#A94E38"></td>
  </tr>
  <tr>
    <td width="160" height="400" valign="top" bgcolor="#EEEEEE"></td>
    <td valign="top"></td>
    <td width="160" valign="top" bgcolor="#EEEEEE"></td>
```