

别具光芒 CSS

网页布局案例剖析

前沿科技 温谦 王觐程 编著

- ❖ 全类型原创案例
- ❖ 全剖析核心技术
- ❖ 全过程拆分讲解
- ❖ 全配套在线支持



人民邮电出版社
POSTS & TELECOM PRESS

别具光芒 CSS

网页布局案例剖析

前沿科技 温谦 王解程 编著



人民邮电出版社

北京

图书在版编目 (C I P) 数据

别具光芒 : CSS网页布局案例剖析 / 温谦, 王觐程
编著. — 北京 : 人民邮电出版社, 2010.5
ISBN 978-7-115-21870-4

I. ①别… II. ①温… ②王… III. ①主页制作—软件工具, CSS IV. ①TP393.092

中国版本图书馆CIP数据核字(2010)第033210号

内 容 提 要

本书以七大完整案例为线索, 紧密围绕在使用 CSS + DIV 布局制作网页过程中遇到的实际需要和应该掌握的技术, 全面介绍了 CSS 布局网页的各方面内容和技巧。

书中摒弃了案例书常见的一味罗列步骤, 轻视技术原理的做法, 把案例作为学习手段, 不但侧重讲解 CSS 盒子模型、标准流、浮动、定位等四大核心技术原理, 更侧重讲解如何在实践中应用它们。读者通过案例, 既可以掌握 CSS 的核心技术要点, 也可以了解相关技术和工作流程, 使读者不但知其然, 还知其所以然。

此外, 本书还结合了流行的 CMS 内容管理系统, 讲解了使用 CSS 进行模板设计的方法, 使读者可以举一反三地掌握为各种内容管理系统设计模板的方法。

本书适合具备一定 HTML 和网页设计制作基础的需要使用 CSS 的爱好者、Web 设计人员和开发人员阅读。

别具光芒——CSS 网页布局案例剖析

- ◆ 编 著 前沿科技 温 谦 王觐程
责任编辑 杨 璐
- ◆ 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>
北京鑫丰华彩印有限公司印刷
- ◆ 开本: 787×1092 1/16
印张: 16.25
字数: 413 千字 2010 年 5 月第 1 版
印数: 1—4 000 册 2010 年 5 月北京第 1 次印刷

ISBN 978-7-115-21870-4

定价: 49.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223
反盗版热线: (010)67171154

前言

CSS 和 HTML 一样，是一个非常基础性的规范，是任何一个 Web 站点设计过程中都离不开的“基础技术”，因此无论是作为设计师，还是作为开发人员，都应该很好地掌握 CSS 这个不可或缺的“武器”。本书就是希望给读者一个掌握 CSS 的阶梯，帮助读者更快捷地掌握 CSS。

本书的由来

编写出版这本书，很大程度上可以说是读者给我们的“命题作文”。在 2006 和 2008 年，我们分别出版了《精通 CSS+DIV 网页样式与布局》和《CSS 设计彻底研究》这两本书，受到很多读者的厚爱，令我们非常欣喜和感谢。在我们的网站上，收到了近万条读者留言，这也令我们十分感动。

同时，有很多读者在我们的网站上留言，希望能够有更多的实际案例的讲解，使读者可以跟随着这样的讲解，“完整地”、“全过程地”、“非常实际地”、“详细地”看一看一个 CSS 布局的页面是如何从无到有制作出来的。

本书正是按照这个思路来设计的，每章各自独立地介绍一个完整的案例，每个案例都是从零开始，直至制作出最后的效果来。希望这本书能给予读者一些实际的帮助。

本书的特点

本书主要突出了两个特点。

1. 知其然，更知其所以然

以“案例”为线索的书是很受读者欢迎的，因此市面上也有很多讲解“案例”的书，而此类书最容易出现的问题是把书变成了操作过程的简单记录，这样的结果是读者往往做过之后不明白为什么，遇到实际问题时，还是不知道该如何入手，这是因为没有把道理真正讲清楚。

“案例”只是一个讲解的手段，而更重要的目的是使广大读者能够真正理解为什么要这么做，只有这样，才能使读者真正掌握。

因此，在安排本书的内容时，我们特别考虑了以下重点问题：

- (1) CSS 的核心技术原理；
- (2) 如何在实践中使用某个原理；

(3) 在实际制作时, 需要注意的相关技术和工作流程。

在各章中, 我们都把上面 3 方面重点有机地融合在一个案例中。例如在第 2 章, 以一个摄影师个人网站为案例, 制作一个网站首页。通过这个案例的学习, 将讲解“盒子模型”、“标准流”、“绝对定位”和“相对定位”这几个 CSS 中非常重要的核心原理。与此同时, 还介绍了设计的完整过程和步骤。这样读者通过这一个案例, 既可以掌握技术要点, 也可以了解通用的工作流程。

2. 讲解了流行的 CMS 内容管理系统的模板设计方法

当前建立网站通常有两种方式, 一种是完全根据需要定制开发, 另一种是使用现成的内容管理系统, 简称为 CMS(Content Management System)。而当前也出现了很多非常流行的内容管理系统, 很多网站正是使用 CMS 系统建立起来的, 而要使用好 CMS 系统, 就需要对 CMS 的模板机制和利用 CSS 来进行定制模板的方法熟练掌握。本书的后两章, 就是专门针对使用 CMS 进行模板定制来讲解的。

配套内容

本书所有案例相关的素材和案例文件, 请到我们的支持网站下载, 网址是 <http://learning.artech.cn/download/css-case-book.rar>。

目 录

Chapter

1

第 1 章 从基础开始

1

1.1 (X)HTML 与 CSS	2
1.1.1 DOCTYPE (文档类型) 的含义与选择	2
1.1.2 XHTML 与 HTML 的重要区别	3
1.1.3 CSS 的基本思想	3
1.1.4 CSS 的引入	5
1.1.5 浏览器与 CSS	5
1.2 基本 CSS 选择器	6
1.2.1 理解 CSS 选择器的思想	6
1.2.2 标记选择器	7
1.2.3 类别选择器	8
1.2.4 ID 选择器	9
1.3 在 HTML 中引入 CSS 的方法	10
1.3.1 行内样式	10
1.3.2 内嵌式	11
1.3.3 外部样式表	11
1.3.4 链接式	12
1.3.5 各种方式的优先级问题	12
1.4 动手体验 CSS	14
1.4.1 从零开始	15
1.4.2 使用 CSS 设置标题	15
1.4.3 控制图片	16
1.4.4 设置正文	16

1.4.5	设置整体页面	17
1.4.6	对段落分别进行设置	17
1.4.7	兼容性检查	18
1.4.8	CSS 的注释	19
1.5	网页使用的编辑软件	20
1.5.1	可视化网页制作软件的优点	20
1.5.2	可视化软件的局限性	21
1.5.3	善于使用代码视图的功能	22
1.6	CSS 的复合选择器	24
1.6.1	“交集”选择器	24
1.6.2	“并集”选择器	25
1.6.3	“后代”选择器	27
1.7	CSS 的继承特性	29
1.7.1	继承关系	29
1.7.2	CSS 继承的运用	31
1.8	CSS 的层叠特性	33
1.9	本章小结	35

Chapter

2

第 2 章

摄影师个人网站布局

36

2.1	案例描述	37
2.2	内容分析	38
2.3	HTML 结构设计	40
2.4	原型设计	41
2.5	页面方案设计	43
2.6	布局设计	43
2.7	CSS 技术准备——盒子模型	44
2.8	设置页面的整体背景	46
2.9	制作照片展示区域	48
2.10	设置网页标题的图像替换	50
2.11	CSS 技术准备——定位	52
2.11.1	理解标准文档流	52
2.11.2	认识定位属性	53
2.11.3	绝对定位	54
2.12	设置网页标题的位置	55
2.13	设置网页文本内容	56
2.14	本章小结	58

3.1 案例描述	60
3.2 内容分析	61
3.3 原型设计	62
3.4 页面方案设计	63
3.5 CSS 技术准备——盒子的浮动	63
3.5.1 设置浮动	64
3.5.2 浮动的方向	66
3.5.3 使用 clear 属性清除浮动的影响	67
3.5.4 扩展盒子的高度	69
3.6 布局设计	70
3.7 CSS 技术准备——在 CSS 中设置边框	70
3.7.1 对不同的边框设置不同的属性值	71
3.7.2 在一行中同时设置边框的宽度、颜色和样式	71
3.7.3 对一条边框设置与其他边框不同的属性	72
3.7.4 同时指定一条边框的一种属性	72
3.8 制作页头部分	72
3.9 制作主体部分	75
3.9.1 主体的左侧部分	76
3.9.2 主要内容区	79
3.10 CSS 技术扩展——扩充布局	82
3.11 本章小结	85

4.1 两列布局	87
4.2 案例描述	89
4.3 内容分析	90
4.4 原型设计	91
4.5 CSS 技术准备——在 CSS 中使用背景图像	92
4.5.1 设置平铺方式	92
4.5.2 设置背景图像的位置	94
4.5.3 背景的简写	96
4.5.4 图片的固定设置	97
4.6 制作标题图像	97
4.7 CS3 技术准备——实现圆角设计	99

4.8	制作页头部分	100
4.8.1	搭建页头部分的 HTML 结构	100
4.8.2	页面标题的图像替换	102
4.8.3	顶部菜单	102
4.8.4	主菜单	103
4.8.5	搜索框	104
4.8.6	页头部分的圆角框	105
4.9	制作主体部分	107
4.9.1	结构分析	107
4.9.2	“面包屑”导航	109
4.9.3	设置正文标题	109
4.9.4	设置页脚	110
4.9.5	添加页面内容	110
4.10	本章小结	112

Chapter



5

第 5 章

网上书店布局

.....113

5.1	案例描述	114
5.2	内容分析	116
5.3	HTML 结构设计	118
5.4	原型设计	121
5.5	页面方案设计	122
5.5.1	配色的技巧	123
5.5.2	切片的技巧	125
5.6	使用滑动门技术制作导航菜单	128
5.7	制作主体部分	131
5.7.1	整体样式设计	131
5.7.2	内容部分的结构分析	131
5.7.3	设置右侧的主要内容列	133
5.7.4	制作左边栏	136
5.8	总结 CSS 布局的优点	139
5.9	制作可以适应变化宽度的圆角框	140
5.10	CSS 技术扩展——从“网页”到“网站”	143
5.10.1	历史回顾	143
5.10.2	不完善的办法	144
5.10.3	服务器出场	144
5.10.4	CMS 出现	144
5.10.5	具体操作	144
5.10.6	CMS 的弊端	145
5.11	本章小结	145

6.1 案例描述	147
6.2 内容分析	148
6.3 HTML 结构设计	151
6.4 原型设计	153
6.5 页面方案设计与切图	154
6.6 页面布局	155
6.6.1 切片	156
6.6.2 CSS 技术准备——行内元素与块级元素	157
6.6.3 布局	160
6.6.4 制作顶部菜单	160
6.6.5 制作标题图像	162
6.6.6 制作主体部分	162
6.7 实现超链接特效	164
6.7.1 技术准备——设置超链接的 CSS 样式	165
6.7.2 超链接效果	168
6.8 兼容性检查	169
6.9 本章小结	170

7.1 构思设计	172
7.1.1 站点分析定位	172
7.1.2 学习考察同类站点	173
7.1.3 构思规划站点	176
7.2 切片制作和生成	178
7.2.1 切片的制作	178
7.2.2 切片的生成	180
7.3 页面制作	180
7.3.1 整体框架的构建	181
7.3.2 头部的制作	182
7.3.3 首页左侧信息栏的制作	185
7.3.4 首页中部内容栏的制作	187
7.3.5 在页面右侧添加百度搜索	188
7.3.6 在页面右侧添加广告位	192

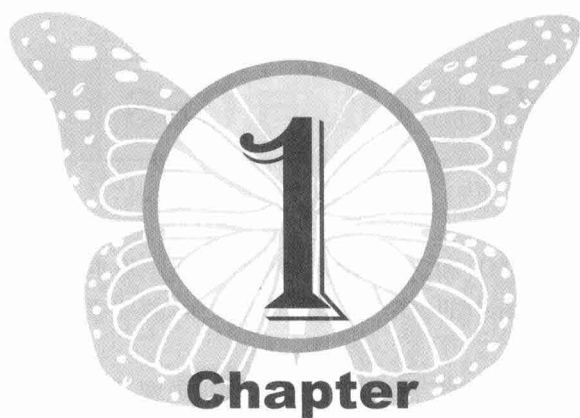
7.3.7 分类目录中导航的制作	192
7.3.8 文章浏览区域的制作	194
7.3.9 页脚的制作	197
7.3.10 用户面板的制作	199
7.4 本章小结	202

Chapter



第 8 章 橘汁仙剑游戏网站 (动态) 布局203

8.1 SupeSite 和 Discuz! 系统简介	204
8.2 系统安装	205
8.3 使用 SupeSite 系统	206
8.3.1 登录 SupeSite 后台设置	206
8.3.2 基本设置	208
8.3.3 资讯的发布和管理	210
8.3.4 资讯等级审核	212
8.3.5 资讯自定义字段	213
8.3.6 其他设置	215
8.4 制作 SupeSite 模板	215
8.4.1 SupeSite 模板系统	215
8.4.2 选择需要制作的模板	216
8.4.3 制作前的准备	218
8.4.4 首页头部信息的制作	220
8.4.5 首页头部导航的制作	221
8.4.6 体验 SupeSite 模块设置	223
8.4.7 在头部导航中加入资讯分类	226
8.4.8 首页主体内容的制作	228
8.4.9 深入探究 SupeSite 模块系统	230
8.4.10 首页页脚的制作	232
8.4.11 站点头部及页脚文件的制作	234
8.4.12 分类目录页的制作	236
8.4.13 文章浏览页面的制作	237
8.4.14 用户面板的制作	239
8.4.15 SupeSite 模板制作小结	241
8.5 模块系统的高级应用	242
8.5.1 SupeSite 模块系统的语法格式	242
8.5.2 条件判断语句	244
8.5.3 自定义广告显示函数	246
8.6 完成测试	247
8.7 Discuz! 模板系统简介	247
8.8 本章小结	250



第 1 章 从基础开始

制作网页的基础是使用 HTML 语言。实际上使用 HTML 非常简单，其核心思想就是需要设置什么样式，就使用相应的 HTML 标记或者属性。然而仅仅依靠 HTML 会遇到很多不可解决的问题，为此 HTML 逐步发展到了 XHTML，CSS 也应运而生。因此在本章中，将简单介绍 HTML、XHTML 和 CSS 三者之间的关系，以及 CSS 的基础，需要读者重点理解使用 CSS 的核心原理。



(X)HTML 与 CSS

相信读者都听说过 HTML 和 XHTML 这两个术语,基本上可以认为它们是一种语言的不同阶段,类似于汉语中的文言文和白话文之间的关系,因此他们也经常被称为 (X)HTML。

在 HTML 的初期,为了它能更广泛地被接受,大幅度放宽了其标准,例如标记可以不封闭,属性可以加引号也可以不加引号,等等。这导致出现了很多混乱和不规范的代码,不符合标准化的发展趋势,影响了互联网的进一步发展。随着网络技术日新月异的发展,HTML 也经历不断的改进,从而产生了 XHTML(可扩展 HTML),因此可以认为 XHTML 是 HTML 的“严谨版”。

(X)HTML 是由 W3C 组织(World Wide Web Consortium,即全球万维网联盟)负责制定的。W3C 的主要职责是研究 Web 规范和指导方针,推动 Web 发展,确定未来万维网的发展方向,并且制定相关的建议。它负责制定了 CSS、XML、XHTML 和 MathML 等多种网络技术规范。

从 HTML 到 XHTML,经历了若干版本。目前 HTML 最高到 4.01 版,XHTML 到 1.1 版。

1.1.1 DOCTYPE (文档类型) 的含义与选择

由于同时存在不同的规范和版本,因此为了使浏览器能够兼容多种规范,规范中规定可以使用 DOCTYPE 指令来声明使用哪种规范解释该文档。目前,常用 HTML 或者 XHTML 作为文档类型,而规范又规定,HTML 和 XHTML 又各自有不同的子类型,例如包括“严格类型”和“过渡类型”的区别。

其中,“过渡类型”是兼容以前版本定义、而在新版本已经废弃的标记和属性,“严格版本”则不兼容已经废弃的标记和属性。

目前,建议读者使用 XHTML 1.0 transitional 类型(XHTML 1.0 过渡类型),这样设计师可以按照 XHTML 的标准书写符合 Web 标准的网页代码,同时在一些特殊情况下还可以使用传统的做法。具体声明方法如下面的这段代码所示。

```
<!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">
  <head>
    <title> 无标题文档 </title>
  </head>
  <body>
  </body>
</html>
```

可以看到最上面有两行关于“DOCTYPE”(文档类型)的声明,它就是告诉浏览器,使用 XHTML 1.0 的过渡规范来解释这个文档中的代码。在第 3 行中,<html> 标记带有

一个 xmlns 属性，它称为“XML 命名空间”，其具体含义不用深究、不用修改，只要照抄即可。

读者如果觉得这些代码难以记忆，可以使用 Dreamweaver 或者 Expression Web 等网页制作辅助软件，在新建文档的时候选择使用哪种文档类型，这些文档类型代码就会自动生成，不需要用户记住具体代码。

例如，在 Dreamweaver 的新建文档对话框中，在右下方有一个【文档类型】下拉框，如图 1.1 所示。

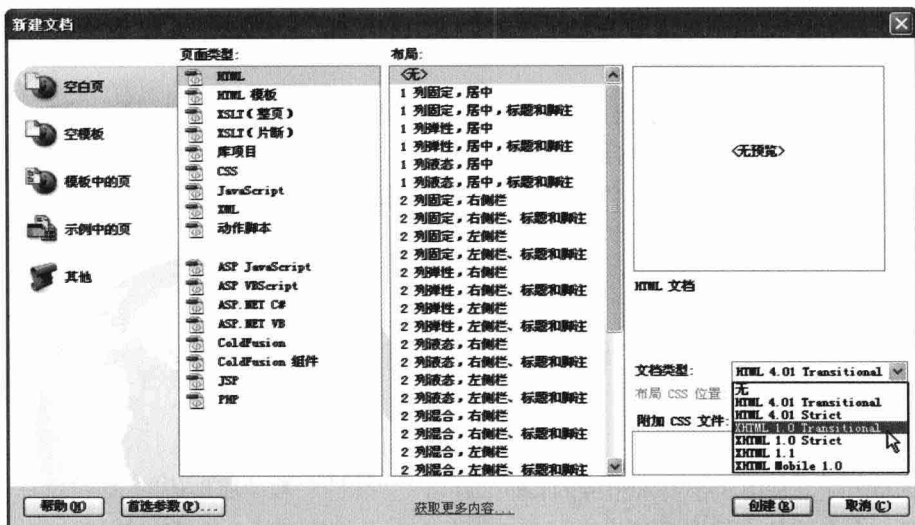


图 1.1 在 Dreamweaver 中选择文档类型

1.1.2 XHTML 与 HTML 的重要区别

XHTML 和 HTML 最重要的区别是许多 HTML 中定义的标记和属性已经被废止了，此外还需要注意以下注意事项：

- (1) 在 XHTML 中标记名称必须小写；
- (2) 在 XHTML 中属性名称必须小写；
- (3) 在 XHTML 中标记必须严格嵌套；
- (4) 在 XHTML 中标记必须封闭；
- (5) 在 XHTML 中空元素的标记也必须封闭；
- (6) 在 XHTML 中属性值用双引号括起来；
- (7) 在 XHTML 中属性值必须使用完整形式。

1.1.3 CSS 的基本思想

CSS 出现的根本原因在于，HTML 中的内容与表现代码混杂在一起，导致出现了以下几点仅靠 HTML 难以解决的问题。

- (1) 维护困难。为了修改某个特殊标记（例如上例中的 <h2> 标记）的格式，需要花费

很多的时间,尤其对于整个网站而言,后期修改和维护的成本很高。

(2) 标记不足。HTML 本身的标记十分的少,很多标记都是为网页内容服务的,而关于美工样式的标记(如文字间距、段落缩进等)在 HTML 中很难找到。

(3) 网页过“胖”。由于没有对各种风格样式进行统一控制,因此 HTML 的页面往往体积过大,占用掉了很多宝贵的带宽。

(4) 定位困难。在整体布局页面时,HTML 对于各个模块的位置调整显得捉襟见肘,过多的其他标记同样也导致页面的复杂和后期维护的困难。

为此,CSS 应运而生了。CSS 的全称是 Cascading Style Sheet,中文译为层叠样式表,它是用于控制网页样式并允许将样式信息与网页内容分离的一种标记性语言。CSS 最早是 1996 年由 W3C 审核通过并推荐使用的。

(X)HTML 与 CSS 的关系就是“内容结构”与“表现形式”的关系,由 (X)HTML 确定网页的结构内容,而通过 CSS 来决定页面的表现形式。

为了理解 CSS 的用法,在具体使用 CSS 之前,请读者先思考一个生活中的问题:通常我们是如何描述一个人的?我们可以为一个人列一张表:

```
李逵 {  
    身高: 185cm;  
    体重: 105kg;  
    性别: 男;  
    性格: 莽撞;  
}
```

这个表实际上是由 3 个要素组成的,即“姓名”、“属性”和“属性值”。通过这样一张表,就可以把一个人的基本情况描述出来了。表中每一行分别描述了一个人的某一种属性以及该属性的属性值。

CSS 的作用就是设置网页的各个组成部分的表现形式,因此,如果把上面的表格换成描述网页上一个标题的属性表,可以设想应该大致是这个样子:

```
2 级标题 {  
    字体: 宋体;  
    大小: 15 像素;  
    颜色: 红色;  
}
```

再进一步,如果我们把上面的表格用英文写出来,如下:

```
h2 {  
    font-family: 宋体;  
    font-size: 15px;  
    color: red;  
}
```

这就是完全正确的 CSS 代码了。由此可见,CSS 的原理实际上非常简单,对于使用英语的人来说,写 CSS 代码几乎就像使用自然语言一样简单。而对于我们中国人,只要理解了这些属性的含义,也并不复杂,相信每一位读者都可以掌握它。

CSS 的思想就是首先指定对什么“对象”进行设置,然后指定对该对象的哪个方面的“属性”进行设置,最后给出该设置的“值”。因此,概括来说,CSS 就是由“对象”、“属性”和“值”3 个基本部分组成的。

1.1.4 CSS 的引入

下面看一个具体的页面，代码如下。

```
<html>
<head>
  <title> 演示 </title>
  <meta http-equiv="Content-Type" content="text/html; charset=gb2312">
  <style>
  p{
    color:blue;
  }
</style>
</head>
<body>
  <h2> 这是标题文本 </h2>
  <p> 这里是正文内容 </p>
  <p> 这里是正文内容 </p>
</body>
</html>
```

可以看到，这个页面由 1 个标题和 2 个文本段落构成，在 HTML 中没有设置任何 font 属性，而在页面的 head 部分，使用了 <style> 标记以及其中对 <p> 标记的定义，即：

```
p{
  color:blue;
}
```

可以看到，页面中的两个文本段落都是用蓝色显示的，这就是 CSS 产生的作用，如图 1.2 所示。

从这个很简单的例子中可以明显看出，CSS 对于网页的整体控制较单纯的 HTML 语言有了突破性的进展，并且它后期修改和维护都十分方便。不仅如此，CSS 还提供各种丰富的格式控制方法，使得网页设计者能够轻松地应对各种页面效果，这些都将在后面的章节中逐一讲解。

最核心的变化就是，原来由 HTML 同时承担的“内容”和“表现”双重任务，现在分离开了，内容仍然由 HTML 负责，而表现形式则是通过 <style> 标记中的 CSS 代码负责的。当然，由于还没有介绍具体 CSS 的用法，因此以上代码的具体内容读者可能还无法清晰地理解，但是读者只要明白其中的原理即可。

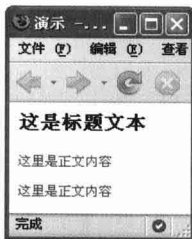


图 1.2 设置 CSS 样式后的效果

1.1.5 浏览器与 CSS

网上的浏览器各式各样，绝大多数浏览器对 CSS 都有很好的支持，因此设计者往往不用担心其设计的 CSS 文件不能显示。但目前主要的问题在于，各个浏览器对 CSS 很多细节的处理上存在差异，设计者在一种浏览器上设计的 CSS 效果，在其他浏览器上的显示效果很可能不一样。就目前主流的两大浏览器 IE (Internet Explorer) 与 Firefox 而言，在某些细节

的处理上就不尽相同。IE 6 与 IE 7 对相同页面的浏览效果也存在一些差异。图 1.3 显示的分别是 IE 和 Firefox 的标志。



图 1.3 IE 浏览器和 Firefox 浏览器的标志

就目前而言,使用最多的 3 种浏览器是 IE 6、IE 7 和 Firefox,制作网页后应该进行调整,使页面在 Internet Explorer 6.0、Internet Explorer 7.0 和 Firefox 这 3 个浏览器中都显示正确,这样可以保证 99% 以上的访问者正确浏览该网页。

但比较幸运的是,出现各个浏览器效果上的差异,主要是因为各个浏览器对 CSS 样式默认值的设置不同,因此可以通过对 CSS 文件各个细节的严格编写,使得各个浏览器达到基本相同的效果。这点在后续的章节中都会提到。

经验

使用 CSS 制作网页时,一个基本的要求就是网页在主流的浏览器中的显示效果要基本一致。通常的做法是一边编写 HTML 和 CSS 代码,一边在两个不同的浏览器上进行预览,及时地调整各个细节,这对深入掌握 CSS 也是很有好处的。

另外 Dreamweaver 的“视图”模式只能作为设计时的参考来使用,绝对不能作为最终显示效果的依据,只有浏览器中的效果才是大家所看到的。

1.2

基本 CSS 选择器

在 CSS 的 3 个组成部分中,“对象”是很重要的,它指定了对哪些网页元素进行设置,因此,它有一个专门名称——选择器 (selector)。

选择器是 CSS 中很重要的概念,所有 HTML 语言中的标记样式都是通过不同的 CSS 选择器进行控制的。用户只需要通过选择器对不同的 HTML 标签进行选择,并赋予各种样式声明,即可实现各种效果。

1.2.1 理解 CSS 选择器的思想

为了理解选择器的概念,可以用“地图”作为类比。在地图上都可以看到一些“图