



高职高专**立体化教材**计算机系列

CSS+DIV网页布局 技术教程

黄玉春 主 编



赠送电子课件及
其他立体化资源



清华大学出版社

内 容 简 介

CSS+DIV 技术是目前最流行的网页制作技术。

本书主要介绍了超文本标记语言 HTML 和 XHTML、CSS 层叠样式表的基础理论和实际运用技术,通过实例分析讲解了 CSS 语法、文字段落设置、背景图片设置以及 CSS 滤镜的应用等,重点讲解了如何运用 CSS+DIV 进行网页布局,注重实际操作,使读者能更好地掌握 CSS 技术及 CSS+DIV 布局的精髓。本书最后给出了两个常见类型的完整网页的综合实例,让读者进一步巩固所学的知识,提高综合应用的能力。

本书适合作为大中专院校计算机应用技术和电子商务类专业的教材,也可作为网页设计技术培训教程,还可以作为 Web 网站开发人员、网页设计师、网页前端架构师、用户体验设计师的参考书。

本书有配套的实例源代码及电子教案(PPT)。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

CSS+DIV 网页布局技术教程/黄玉春主编.--北京:清华大学出版社,2012

(高职高专立体化教材 计算机系列)

ISBN 978-7-302-29531-0

I. ①C… II. ①黄… III. ①网页制作工具—高等职业教育—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2012)第 169960 号

责任编辑:桑任松

封面设计:刘孝琼

版式设计:杨玉兰

责任校对:周剑云

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:15.5

字 数:374 千字

版 次:2012 年 9 月第 1 版

印 次:2012 年 9 月第 1 次印刷

印 数:1~4000

定 价:28.00 元

产品编号:045864-01

《高职高专立体化教材计算机系列》丛书序

一、编写目的

关于立体化教材,国内外有多种说法,有的叫“立体化教材”,有的叫“一体化教材”,有的叫“多元化教材”,其目的是一样的,就是要为学校提供一种教学资源的整体解决方案,最大限度地满足教学需要,满足教育市场需求,促进教学改革。我们这里所讲的立体化教材,其内容、形式、服务都是建立在当前技术水平和条件基础上的。

立体化教材是“一揽子”式的(包括主教材、教师参考书、学习指导书、试题库在内的)完整体系。主教材讲究的是“精品”意识,既要具备指导性和示范性,也要具有一定的适用性,喜新不厌旧。那种内容越编越多,本子越编越厚的低水平重复建设在“立体化”的世界中将被扫地出门。与以往不同,“立体化教材”中的教师参考书可不是千人一面的,教师参考书不只是提供答案和注释,而是含有与主教材配套的大量参考资料,使得老师在教学中能做到“个性化教学”。学习指导书更像一本明晰的地图册,难点、重点、学习方法一目了然。试题库或习题集则要完成对教学效果进行测试与评价的任务。这些组成部分采用不同的编写方式,把教材的精华从各个角度呈现给师生,既有重复、强调,又有交叉和补充,相互配合,形成一个教学资源有机的整体。

除了内容上的扩充,立体化教材的最大突破还在于在表现形式上走出了“书本”这一平面媒介的局限,如果说音像制品让平面书本实现了第一次“突围”,那么电子和网络技术的大量运用就让躺在书桌上的教材真正“活”了起来。用 PowerPoint 开发的电子教案不仅大大减少了教师案头备课的时间,而且也让学生的课后复习更加有的放矢。电子图书通过数字化使得教材的内容得以无限扩张,使平面教材更能发挥其提纲挈领的作用。

CAI(计算机辅助教学)课件把动画、仿真等技术引入了课堂,让课程的难点和重点一目了然,通过生动的表达方式达到深入浅出的目的。在科学指标体系控制之下的试题库既可以轻而易举地制作标准化试卷,也能让学生进行模拟实战的在线测试,提高了教学质量评价的客观性和及时性。网络课程更厉害,它使教学突破了空间和时间的限制,彻底发挥了立体化教材本身的潜力,轻轻敲击几下键盘,你就能在任何时候得到有关课程的全部信息。

最后还有资料库,它把教学资料以知识点为单位,通过文字、图形、图像、音频、视频、动画等各种形式,按科学的存储策略组织起来,大大方便了教师在备课、开发电子教案和网络课程时的教学工作。如此一来,教材就“活”了。学生和书本之间的关系不再像领导与被领导那样呆板,而是真正有了互动。教材不再只为老师们规定什么重要什么不重要,而是成为教师实现其教学理念的最佳拍档。在建设观念上,从提供和出版单一纸质教材转向提供和出版较完整的教学解决方案;在建设目标上,以最大限度满足教学要求为根

本出发点;在建设方式上,不单纯以现有教材为核心,简单地配套电子音像出版物,而是以课程为核心,整合已有资源并聚拢新资源。

网络化、立体化教材的出版是我社下一阶段教材建设的重中之重,以计算机教材出版为龙头的清华大学出版社确立了“改变思想观念,调整工作模式,构建立体化教材体系,大幅度提高教材服务”的发展目标,并提出了首先以建设“高职高专计算机立体化教材”为重点的教材出版规划,希望通过邀请全国范围内的高职高专院校的优秀教师,共同策划、编写这一套高职高专立体化教材,利用网络等现代技术手段实现课程立体化教材的资源共享,解决国内教材建设工作中存在的教材内容更新滞后于学科发展的状况。把各种相互作用、相互联系的媒体和资源有机地整合起来,形成立体化教材,把教学资料以知识点为单位,通过文字、图形、图像、音频、视频、动画等各种形式,按科学的存储策略组织起来,为高职高专教学提供一整套解决方案。

二、教材特点

在编写思想上,以适应高职高专教学改革的需要为目标,以企业需求为导向,充分吸收国外经典教材及国内优秀教材的优点,结合中国高校计算机教育的教学现状,打造立体化精品教材。

在内容安排上,充分体现先进性、科学性和实用性,尽可能选取最新、最实用的技术,并依照学生接受知识的一般规律,通过设计详细的可实施的项目化案例(而不仅仅是功能性的小例子),帮助学生掌握要求的知识点。

在教材形式上,利用网络等现代技术手段实现立体化的资源共享,为教材创建专门的网站,并提供题库、素材、录像、CAI 课件、案例分析,实现教师和学生更大范围内的教与学互动,及时解决教学过程中遇到的问题。

本系列教材采用案例式的教学方法,以实际应用为主,理论够用为度。教程中每一个知识点的结构模式为“案例(任务)提出→案例关键点分析→具体操作步骤→相关知识(技术)介绍(理论总结、功能介绍、方法和技巧等)”。

该系列教材将提供全方位、立体化的服务。网上提供电子教案、文字或图片素材、源代码、在线题库、模拟试卷、习题答案、案例动画演示、专题拓展、教学指导方案等。

在为教学服务方面,主要是通过教学服务专用网站在网络上为教师和学生提供交流的场所以,每个学科、每门课程,甚至每本教材都建立网络上的交流环境。可以为广大教师信息交流、学术讨论、专家咨询提供服务,也可以让教师发表对教材建设的意见,甚至通过网络授课。对学生来说,则可以在教学支撑平台所提供的自主学习空间中进行学习、答疑、操作、讨论和测试,当然也可以对教材建设提出意见。这样,在编辑、作者、专家、教师、学生之间建立起一个以课本为依据、以网络为纽带、以数据库为基础、以网站为门户的立体化教材建设与实践的体系,用快捷的信息反馈机制和优质的教学服务促进教学改革。

本系列教材的专题网站是 <http://lth.wenyuan.com.cn>。

前 言

随着 Web 2.0 标准的普及,网页标准化 CSS+DIV 的设计方式正逐渐取代传统的表格布局模式。采用 CSS+DIV 布局的优势主要体现在:能缩减页面代码,提高页面浏览速度;结构清晰,容易被搜索引擎搜索到,天生优化了 SEO 等方面。所以对 CSS 的学习是计算机应用技术人员从事网站设计的必修课程。

本书的定位是零基础学习 CSS+DIV 布局技术的大中专学生,也可用于帮助有表格布局基础的学生向 CSS+DIV 布局技术过渡,还可以作为工程技术人员的参考书。

全书共分 9 章。第 1 章介绍 HTML 和 XHTML 基础,重点介绍 HTML 基本用法及如何转换现有的文档为 XHTML。第 2 章介绍 CSS 基础,重点介绍 CSS 选择器和 CSS 的引用。第 3 章介绍用 CSS 设置文字与背景。第 4 章介绍 CSS 滤镜的应用,重点介绍视觉滤镜和转换滤镜。第 5 章介绍 CSS 框模型与定位技术,重点介绍 CSS 框模型和 CSS 定位与浮动技术。第 6 章介绍 DIV+CSS 布局基础,重点介绍 CSS 排版观念和常见的 CSS+DIV 布局的样式的制作方法,并详细介绍两个 CSS+DIV 布局的例子。第 7 章介绍 CSS 与 JavaScript 的综合应用,重点介绍 JavaScript 语法、内置对象、浏览器对象和用 CSS 结合 JavaScript 制作网页的一些特效应用。第 8 章和第 9 章介绍两个综合案例,分别介绍博客网站和企业网站的布局方法。

本书内容翔实、结构清晰、循序渐进,通过大量的技术应用实例深入讲解 CSS+DIV 布局方法。书中精选的网页综合实例有很强的实际应用价值,大部分章节配备了习题和上机实训练习题,便于读者学习。

本书由黄玉春编写,王标为本书提供了部分案例。在本书编写的过程中,得到了不少专家和任课教师的大力支持,在此一并表示衷心的感谢。

编 者

目 录

第 1 章 HTML 和 XHTML 基础..... 1	2.4 CSS 的继承..... 33
1.1 HTML 基本用法..... 1	2.4.1 CSS 的继承关系..... 33
1.1.1 HTML 文档基本结构..... 2	2.4.2 CSS 继承的应用..... 34
1.1.2 常用的 HTML 标签..... 3	习题 2..... 34
1.2 升级到 XHTML..... 13	上机实验 2..... 35
1.2.1 为什么要升级..... 13	第 3 章 用 CSS 设置文字与背景..... 37
1.2.2 XHTML 与 HTML 比较..... 14	3.1 CSS 文字与段落样式..... 37
1.2.3 如何转换现有的文档为 XHTML..... 16	3.1.1 CSS 文字样式..... 37
习题 1..... 17	3.1.2 CSS 段落文字..... 43
上机实验 1..... 21	3.1.3 CSS 列表..... 46
第 2 章 CSS 基础..... 22	3.2 CSS 颜色与背景样式..... 50
2.1 CSS 基本概念..... 22	3.2.1 设置颜色..... 50
2.1.1 CSS 的基本概念..... 22	3.2.2 设置背景图像..... 52
2.1.2 CSS 的特点..... 23	3.2.3 背景图像的重复方式..... 54
2.2 CSS 选择器..... 23	3.2.4 固定背景图像..... 56
2.2.1 CSS 样式规则的定义..... 23	3.2.5 背景图片位置..... 58
2.2.2 标记选择器..... 24	3.2.6 背景样式的缩写..... 60
2.2.3 CSS 选择器分组..... 24	3.3 综合实例..... 61
2.2.4 派生选择器..... 25	3.3.1 文字的综合应用..... 61
2.2.5 id 选择器..... 25	3.3.2 背景的综合应用..... 63
2.2.6 类选择器..... 26	习题 3..... 65
2.2.7 伪类选择器..... 26	上机实验 3..... 67
2.3 CSS 的引用..... 27	第 4 章 CSS 滤镜的应用..... 68
2.3.1 内部样式表..... 27	4.1 滤镜概述..... 68
2.3.2 外部样式表..... 28	4.2 视觉滤镜..... 68
2.3.3 内联样式表..... 29	4.2.1 alpha 属性..... 69
2.3.4 输入样式表..... 30	4.2.2 blur 属性..... 72
2.3.5 多重样式..... 31	4.2.3 Chroma 属性..... 73
	4.2.4 DropShadow 属性..... 74

4.2.5 FlipH、FlipV 属性.....	75	6.1.2 <div>与标记的区别	131
4.2.6 Glow 属性.....	77	6.2 CSS 排版观念	132
4.2.7 Gray 属性.....	78	6.2.1 将页面用 div 分块	132
4.2.8 Invert 属性.....	79	6.2.2 设计各块的位置	133
4.2.9 Mask 属性.....	80	6.2.3 用 CSS 定位	134
4.2.10 Shadow 属性.....	81	6.3 固定宽度且居中的版式.....	135
4.2.11 Wave 属性	82	6.3.1 一列固定宽度居中	135
4.2.12 Xray 属性.....	84	6.3.2 二列固定宽度居中	137
4.3 转换滤镜.....	85	6.4 自适应宽度且居中的版式.....	139
4.3.1 融合转换滤镜.....	86	6.4.1 一列自适应宽度且居中	139
4.3.2 揭示转换滤镜.....	87	6.4.2 二列宽度自适应	140
4.3.3 网页特效转换.....	89	6.4.3 左列固定, 右列宽度 自适应	142
习题 4.....	91	6.4.4 三列浮动中间列宽度 自适应	144
上机实验 4.....	92	6.5 一个常用的 DIV+CSS 网站布局的 基本框架结构实例.....	147
第 5 章 CSS 框模型与定位	93	6.5.1 搭建框架	147
5.1 CSS 框模型	93	6.5.2 添加 CSS 样式	148
5.1.1 CSS 框模型的概述	93	6.6 排版实例: 淘宝商城.....	150
5.1.2 CSS 内边距	95	6.6.1 搭建 HTML 框架	151
5.1.3 CSS 边框	97	6.6.2 添加 CSS 样式	155
5.1.4 CSS 外边距	104	6.6.3 采用 CSS+DIV 布局优势	156
5.1.5 外边距合并.....	108	习题 6.....	157
5.2 CSS 定位和浮动	109	上机实验 6.....	158
5.2.1 块级元素.....	110	第 7 章 CSS 与 JavaScript 的 综合应用	159
5.2.2 CSS 定位机制	110	7.1 JavaScript 概述.....	159
5.2.3 相对定位.....	111	7.1.1 JavaScript 简介	159
5.2.4 绝对定位.....	113	7.1.2 JavaScript 的特点.....	160
5.2.5 z-index 空间位置.....	116	7.1.3 在网页中使用 JavaScript.....	160
5.2.6 CSS 浮动	117	7.2 JavaScript 语法基础.....	162
5.3 CSS 定位实例	123	7.2.1 JavaScript 的数据类型.....	162
习题 5.....	127	7.2.2 常量、变量与表达式	162
上机实验 5.....	128	7.2.3 函数的定义	165
第 6 章 DIV+CSS 布局基础	129		
6.1 div 标记与 span 标记	129		
6.1.1 div 标记与 span 标记概述	129		

7.2.4 条件分支语句.....	167	8.1.2 排版架构	204
7.2.5 循环控制语句.....	169	8.2 模块设计.....	205
7.3 JavaScript 内置对象.....	171	8.2.1 导航与 Banner.....	205
7.3.1 String 对象.....	171	8.2.2 左侧列表	206
7.3.2 Math 对象.....	174	8.2.3 内容部分	208
7.3.3 Array 对象	175	8.2.4 footer 脚注.....	209
7.3.4 Date 对象.....	177	8.3 整体调整.....	210
7.4 浏览器对象.....	179	8.4 网页源码.....	211
7.4.1 浏览器对象简介.....	179	第 9 章 整合案例二：企业网站	217
7.4.2 window 对象.....	181	9.1 分析架构.....	217
7.5 JavaScript 的事件处理.....	186	9.1.1 设计分析	217
7.5.1 事件及响应方法.....	186	9.1.2 排版架构	218
7.5.2 document 的常用事件.....	188	9.2 模块设计.....	219
7.6 实例一：图片的间歇滚动.....	189	9.2.1 导航与 Banner.....	219
7.7 实例二：用 CSS 制作的导航菜单	192	9.2.2 左侧部分	221
7.8 实例三：用 CSS 实现 PPT 幻灯片	195	9.2.3 中间部分	222
习题 7.....	196	9.2.4 右侧部分	224
上机实验 7.....	199	9.2.5 footer 脚注.....	226
第 8 章 整合案例一：我的博客	203	9.3 整体调整.....	227
8.1 分析架构.....	203	9.4 网页源码.....	228
8.1.1 设计分析.....	203	参考文献	236

第 1 章 HTML 和 XHTML 基础

本章要点

- HTML 文件结构
- HTML 常用的标记
- Web 标准
- 升级 HTML 到 XHTML

打开一个网页，查看它的源文件，就会看到一些有规律的程序代码。这些代码就是超文本置标语言(Hypertext Markup Language, HTML)。“超文本”就是指页面内可以包含图片、链接甚至音乐、程序等非文字的元素，“标记”就是说它不是程序语言，只是由文字及标记符号组合而成。

一个网页无论看上去多么五花八门、生动活泼，其实最本质的东西，就是由这些看着十分单调的 HTML 语言组成的，浏览器或者其他可以浏览网页的设备将这些 HTML 语言“翻译”过来，并按照定义的格式显示出来，转化成最终看到的网页。

现在有很多功能强大的网页编辑制作工具，如 Dreamweaver 等，它们使网页制作变得很简单。但是当制作者需要一些特殊的版式或者被一个莫名其妙的现象困扰的时候，最简单的解决方法，就是直接面对 HTML 源代码。对于要写脚本语言程序或者要做服务器端脚本编程的人来说，就更需要了解 HTML 语言。

1.1 HTML 基本用法

HTML 是一种用来制作超文本文档的简单标记语言。目前的版本是 HTML 4.0。

用 HTML 编写的超文本文档称为 HTML 文档，它能独立于各种操作系统平台(如 Unix、Windows 等)。使用 HTML 语言描述的文件需要通过 WWW 浏览器显示出效果。

HTML 是一种纯文本语言，也就是说，HTML 代码在运行时不用事先编译为二进制代码，而是直接通过网页浏览器逐行解释执行。所以，用一般的文本编辑器就可以编写 HTML 代码，保存时只需要把代码文件保存为 .htm 或 .html 格式即可。

1.1.1 HTML 文档基本结构

用 HTML 语言创建的文档称为 HTML 文档，由按照一定规则组合起来的各种标记组成。在 IE 浏览器中运行文件 1-1.html，用记事本查看其源文件，如图 1-1 所示。

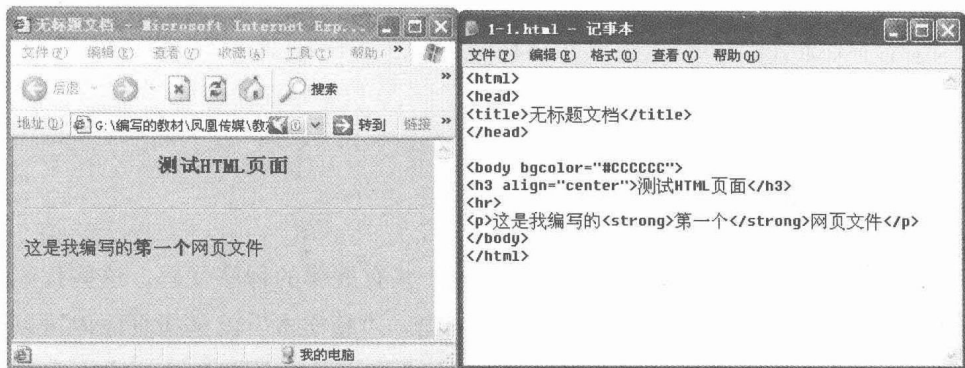


图 1-1 HTML 文件的构成

由图 1-1 可以知道，HTML 文档的基本特征如下。

(1) 用尖括号“<”和“>”括起来的部分称为标记，每个标记都必须有一个标记名称，用来作为该标记的唯一标识，如<html>中的 html。绝大部分标记都有其相关的属性及属性值，如<body bgcolor="#cccccc">，其中 bgcolor 是标记<body>的一个属性，#cccccc 是 bgcolor 的取值。取值可以用引号括起来，也可以不括，标记的属性通常都有一个默认值，如 bgcolor 的默认值是#ffffff。

(2) HTML 标记都是用尖括号“<”和“>”括起来的，大多数标记是成对出现和使用的，有开始标记和对应的结束标记，结束标记多一根斜杠，例如：

```
<title>文档标题文字</title>
```

(3) 很多标记还有自己的属性，利用这些属性，可以做进一步的详细设置。其语法格式为：

```
<标记名 name1="value1" name2="value2" ...>...</标记名>
```

各属性项间用空格分隔，属性值可用双引号或单引号，也可以不用引号。例如控制文本的标记为，该标记有 face、size、color 属性，分别用于控制字体名、字号和字体颜色，例如：

```
<font face="宋体" size=4 color="#ff0000">样本文字</font>
```

(4) HTML 标记可以嵌套使用，实现从不同角度对文本进行格式控制。各标记书写的先后顺序没有特别要求，只要不发生交叉嵌套就行。



例如以下 3 行效果等价：

```
<b><div align=center><font color=#ff0000>文字</font></div></b>
<div align=center><b><font color=#ff0000>文字</font></b></div>
<font color=#ff0000><b><div align=center>文字</div></b></font>
```

(5) 有些标记是单独使用的，没有对应的结束标记。例如换行标记
和画水平线标记<hr>。

(6) HTML 标记不区分大小写，<p>意思和<P>是一样的。

注意：在 XHTML 中规定标记必须为小写，这将在后面的任务中讲解。

由图 1-1 还可以看出，HTML 文档一般由 3 个部分构成，分别如下。

(1) <html> </html>

<html>标记用于定义网页文件的开始，对应的结束标记</html>则定义网页的结束。

(2) <head> </head>

该组标记用于定义网页的头部。在网页的头部可以用<title> </title>标记来定义网页的标题，可以用<meta>标记定义与文档相关的信息，可以放置 JavaScript 块或其他定义部分。

(3) <body> </body>

<body>用于定义网页正文的开始，</body>用于定义网页正文的结束。网页的正文必须放置在这两个标记之间。网页正文包含了网页显示的绝大部分信息，如文字、图片、表格、超链接、多媒体等，是 HTML 文档的核心部分。<body>具有很多属性，在后续章节中将会详细地介绍。

1.1.2 常用的 HTML 标签

在制作一般页面的过程中，经常使用的标记(有时也称标签)有以下几种。

1. 标题

标题(Heading)标记有 6 个级别，从<h1>到<h6>。<h1>为最大的标题，<h6>为最小的标题，可参见任务 2 中的例题。通过设定不同等级的标题，可以完成很多层次结构的设置，比如文档的目录结构或者一份写作大纲。

2. 段落

段落(Paragraphs)标签<p>是处理文字时经常用到的标签。段落内也可以包含其他的标签，如图片标签、换行符标签
、链接标签<a>等。

注意：有些网页中的段落没有结束标签</p>，虽然浏览器仍然能显示出这些段落，但是这是一种不提倡的做法，在 XHTML 中，不允许出现这种情况。

3. 换行

换行标签
是一个单标记，也就是说，它只有起始标记，而没有结束标记。当需要结束一行，并且不想开始新段落时，使用
标记。
标记不管放在什么位置，都能够强制换行。

注意：虽然直接在 HTML 文件内写
并没有错，但是为了向 XHTML 过渡，最好养成关闭标记的习惯，为空标签加上“/”，如
。

4. 超链接

在 HTML 中，通过标记<a>...来加入超链接。<a>和之间的部分称作超链接源，也就是鼠标所点击的区域，一般以文字或图片作为超链接源。该标记的用法为：

```
<a href="url" target="wname" title="*">文字或图片</a>
```

说明：标记<a>表示一个链接的开始，表示链接的结束。

href 属性：用于指定所要链接的目标地址；目标地址是最重要的，一旦路径上出现差错，该资源就无法访问。

target 属性：该属性用于指定打开链接的目标窗口(见表 1-1)，其默认方式是原窗口。

表 1-1 建立目标窗口的属性

属 性 值	描 述
_parent	在上一级窗口中打开，一般使用分帧的框架页会经常使用
_blank	在新窗口打开
_self	在同一个帧或窗口中打开，这项一般不用设置
_top	在浏览器的整个窗口中打开，忽略任何框架

title 属性：该属性用于指定指向链接时所显示的标题文字。

(1) 链接路径

若目标地址是网站内部的其他网页，这时目标地址使用相对路径。例如，当前网页有一个“首页”的菜单项，现定义一个超链接，当用户单击时切换到首页，则该链接的定义方法为：

```
<a href="index.htm">首页</a>
```

若目标地址是外部网站的网页，这时目标地址使用绝对路径。例如，在当前网页创建一个超链接，用以链接到“凤凰传媒集团”网站，则实现的代码为：

```
<a href="http://www.ppm.cn">凤凰传媒集团</a>
```



另外,也可以用图像作为超链接的标志。假设凤凰传媒集团的 logo 在 images 目录下,上述链接也可以表示为:

```
<a href="http://www.ppm.cn"></a>
```

(2) 锚点链接

若要跳转到网页的某一个指定位置,则必须事先在该位置定义一个锚点(anchor),定义锚点用<a>标记的 name 属性来实现,其用法为:

```
<a name="锚点名">
```

定义好锚点后,若要链接到网页的某一锚点,则链接的方法为:

```
<a href="#锚点名">文本</a>
```

(3) 邮件链接

若要使超链接指向电子邮件发送链接,则可以用以下格式来实现:

```
<a href="mailto:电子邮件地址">……</a>
```

单击该链接,就会启动默认的电子邮件发送程序。

(4) 下载链接

当超链接的 URL 是非网页的其他文件时,如果文件能够在浏览器中浏览,就直接在浏览器中打开,如*.jpg、*.gif、*.png、*.txt 等格式的文件;如果该文件不能在浏览器中浏览,就会出现下载对话框,要求下载目标文件,如*.rar、*.exe 等格式的文件。例如,要在当前网页中创建一个超链接,用于下载 download/zy1.rar 文件,则实现的代码为:

```
<a href="download/zy1.rar">点击下载</a>
```

5. 列表

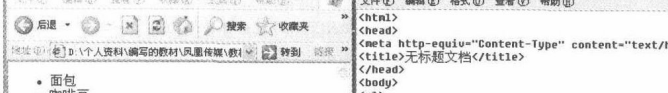
在利用表格排版的时代,列表(lists)的作用被忽略了,很多应该是列表的内容,也转用表格来实现。随着 DIV+CSS 布局的推广,列表的地位变得重要起来,配合 CSS 样式表,列表可以显示成样式繁杂的导航、菜单、标题等。

列表可以分为如下三种。

(1) 无序列表(Unordered Lists): 一个无序列表的开头标记是标记,每个项目的开始标记为,在列表项目中可以加入段落、图像、链接等。列表项在浏览器显示时,通常前面有黑色的圆点来表示,如图 1-2 所示。

(2) 有序列表(Order Lists): 有序列表每个项目前都有数字标记,开始标签是,每个项目的开始标签还是。列表项目中同样可加入段落、图像、链接等,如图 1-2 所示。

(3) 释义列表(Definition Lists): 释义列表是一列事物以及与其相关的解释。释义列表的开始标记是<dl>,每个被解释的事物开始标签为<dt>,每个解释内容的开始标记是<dd>。在<dd>标记中的内容可以是段落、图像、链接等。例 1-2.html 在 IE 浏览器中的预览效果和



6

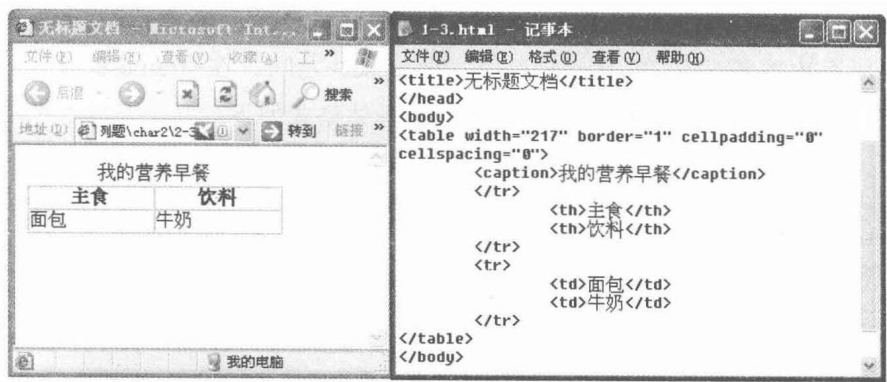


图 1-3 表格标记

- `<caption></caption>`用于定义表格标题的开始和结束，可以省略。
- `<tr></tr>`用于定义表行的开始和结束，一组`<tr></tr>`产生一个表行。
- `<td></td>`用于定义单元格的开始和结束，一组`<td></td>`产生一个单元格。
- `<th></th>`用于定义表头单元格的开始和结束，一组`<th></th>`产生一个表头单元格，该单元内的数据以加粗、居中的方式显示。

表格的`<table>`、`<tr>`、`<th>`、`<td>`等标记都可以设置宽度、高度、背景色等多种属性，但是一般不推荐在 HTML 内定义这些属性，而应该将其统一定义到 CSS 样式表内，以方便修改。

8. DIV 标记与 SPAN 标记

层(DIV)称为定位标记，它不像链接或者表格具有实际的意义，其作用就是设定文字图片等网页元素的摆放位置。现在的 DIV 标记主要用于网页布局，在后续的任务中将有详细的应用和说明。

范围(SPAN)和层的作用类似，只是`<SPAN>`标记一般应用在行内，用以定义一小块需要特别标识的内容，`<SPAN>`标记需要通过设置 CSS 样式表才能发挥作用。

说明：以前我们设置字体红色可通过`<font color="#ff0000">红色字体</font>`来实现。随着 HTML 向 XHTML 过渡，`<font>`标签将不被支持，要设置字体为红色，可通过`<span style="color:red;">红色字体</span>`来实现。

9. 框架

框架(frames)可以在物理上将一个浏览器窗口分成若干个区域，每个区域可以载入一个网页。

所有框架标记放在一个总的 HTML 文件中，这个文件只记录了该框架如何分割，不会显示任何资料，所以没有`<body>`标记，浏览器通过解释这个总文件而将其中划分的各个框

架分别对应的 HTML 文件显示出来。

框架的所有内容都应该在<frameset>和</frameset>之间。在<frameset>标记内,使用<frame>标记来指定框架中每个子窗口的内容。其具体格式如下:

```
<html>
<frameset>
  <frame src="url" name="value">
  <frame src="url" name="value">
  ...
</frameset>
</html>
```

注意: 对于含有框架结构的网页而言,其 HTML 形式和一般的 HTML 文件相似,只是一般文档中的<body>标记被框架的<frameset>标记取代。版本较旧的浏览器可能不支持框架结构。

<frameset>标记用于定义一个框架结构,其属性如下。

(1) rows 属性、cols 属性:用于设置多重框架的高度和宽度。一个窗口可以分割为几块,横向分用 rows 属性,纵向分用 cols 属性,每一块的大小可以由这两个属性的值来实现。

例如,<frameset cols="100,200,300">标记将窗体分割成纵向 3 块,分别占 100、200 和 300 像素;<frameset rows="10%,20%,70%">标记将窗体分割成横向 3 块,分别占 10%、20% 和 70%。

属性的值还可以用剩余的值的形式来表示。

例如,<frameset cols="100,200,*">标记将窗体分割成纵向 3 块,分别占 100、200 和剩余像素;<frameset cols="100,*,*">标记将窗体分割成纵向 3 块,第一块占 100 像素,其他两个将 100 像素以外的窗口平均分配;<frameset cols="*,*,*">该标记将窗口分为 3 等份。

如果要在浏览器中同时横向和纵向分割,则可嵌套使用<frameset>标记来实现。

例 1-4.html 的代码和预览效果如图 1-4 所示。

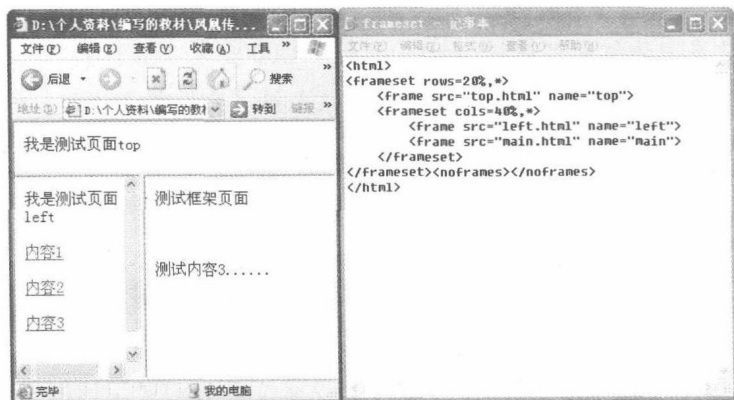


图 1-4 框架的嵌套

(2) **frameborder** 属性: 该属性用于设置是否显示边框, 取值为 **yes** 或 **no**, 分别表示显示或不显示边框。默认值为 **yes**。

(3) **border** 属性: 该属性用于设置边框的大小, 单位是像素。

<frame> 标记用于在网页中定义框架, **<frame>** 标记是一个单标记, 使用时放在 **<frameset>** 与 **</frameset>** 之间。其常用属性如下。

- **name** 属性: 该属性用于设置子窗口的名称。在设置超链接时, 若要使链接的页面在框架中显示, 则只需设置 **<a>** 标记的 **target** 属性值为框架的名称即可。
- **src** 属性: 该属性用于设置所要载入的文件名称。
- **scrolling** 属性: 该属性用于设置子窗口是否有滚动条, 取值为 **yes**、**no** 和 **auto**, 分别表示有滚动条、无滚动条和根据窗口内容的多少决定是否显示滚动条。默认值为 **auto**。
- **noresize** 属性: 该属性用于设置是否能够调整窗口的大小, 取值为 **yes** 或 **no**, 分别表示能够调整或不能够调整。默认值为 **yes**。

10. 在网页中使用内嵌框架

若要在一个网页中包含并显示另一个网页的内容, 则可通过使用内嵌框架来实现, 设置内嵌框架的标记为 **<iframe>**, 其用法为:

```
<iframe name=# src="url" scrolling=# frameborder=# height=# width=#>
</iframe>
```

属性说明如下。

- **name**: 该属性用于设置框架的名称。
- **src**: 该属性用于设置所要载入的网页文件名称。
- **scrolling**: 该属性用于设置子窗口是否有滚动条, 取值为 **yes** 或 **no**, 分别表示有滚动条或无滚动条。默认值为 **yes**。
- **frameborder**: 该属性用于设置是否显示边框, 取值为 **yes** 或 **no**, 分别表示显示或不显示边框。默认值为 **yes**。
- **height**、**width**: 用于设置框架的高度和宽度。

例如, 若要在当前网页中使用内嵌框架显示 **weather.htm** 网页, 框架的高度为 40, 宽度为 60, 不显示滚动条, 则实现的代码为:

```
<iframe src="weather.htm" scrolling=no frameborder=no height=40 width=60>
</iframe>
```

11. 表单

表单在网页设计(尤其是动态网页设计中)起着重要的作用, 它是用户与 Web 服务器进行信息交互的主要手段。在网络上, 通过填写表单并提交的方式完成用户信息的收集并将