

北京科海培训中心

动态网页设计案例教程

杨正华 编著

中科多媒体电子出版社

2001.9

内 容 提 要

本书介绍动态网页技术的核心内容CSS（全称为Cascading Style Sheets，层叠样式表）。掌握CSS，已成为网页制作人员的必备技能。通过CSS可以实现页面视觉效果的一切变化，如字体、对齐方式、颜色、位置、大小、定位、分层与重叠、滤镜效果等，几乎可以达到用户网络带宽和处理器运行速度允许范围内的一切动态交互、神奇变幻的网页效果。

本书精选出55个案例进行教学，将CSS的知识点有机地分散在这55个案例中。通过案例，使得CSS众多的对象、属性以及各种设置和操作都变得具体形象、通俗易懂，让读者从实际的操作、具体的设计中体会、领悟、积累CSS和动态网页制作的经验技巧，循序渐进，深入浅出，可操作性极强。同时，这些案例非常实用，读者可以直接“拿来”或者简单修改后再加入自己的网页中，轻轻松松使自己的网页多姿多彩、变幻莫测，达到令人惊叹的美观布局和令人震撼的动态效果。

本书及其配套光盘（含各案例的动态网页特效及源代码）适合于各个层次的网页制作爱好者使用。

品 名：动态网页设计案例教程

作 者：杨正华

责任编辑：杨光达 魏胜

出 品：中科多媒体电子出版社

印 刷 者：北京门头沟胶印厂

发 行：新华书店总店北京科技发行所

开 本：787×1092 1/16 印张：18.75 字数：429千字

版 次：2001年9月第1版 2001年9月第1次印刷

印 数：0001~5000

盘 号：ISBN 7-900084-19-3

定 价：28.00元（ICD）

前 言

动态 HTML (DHTML) 是近年来网络发展进程中最振奋人心, 也最具实用性的创新之一, 而 CSS 是动态网页制作的核心技术之一。掌握 CSS, 已成为网页制作人员的必备技能。

什么是 CSS

CSS 技术诞生于 1996 年底, 全称是 Cascading Style Sheets (层叠样式表), 简称 CSS。通过 CSS, 几乎能够实现网页在视觉效果上的一切变化。通过改变网页元素的 CSS 属性(如颜色、位置、大小), 几乎可以达到用户网络带宽和处理器运行速度允许范围内的一切效果。

CSS 简化了 HTML 标识符, 并且减轻了 HTML 对表现形式的责任。它们使设计者和用户都可以控制文档的表现形式——字体信息、对齐方式、颜色、定位、分层与重叠等。样式信息可以为单独的元素或一组元素定制, 它既可以定制在 HTML 文档内, 也可以以外部层叠样式表的形式出现。这实在是一件非常美妙的事情。

本书的内容和目标

由 CSS 技术实现的网页, 带来的浏览效果是令人震撼的。掌握 CSS 并不需要枯燥地学习众多的对象、繁杂的属性。本书通过精选的案例进行教学, 尽量减少冗长乏味的说明, 代之以具体实用的案例。通过简明精炼地讲解, 精彩明晰的案例, 引导读者把握网页制作的精髓所在。使得学习过程成为循序渐进的、轻轻松松的过程。本书适合于各个层次的网页制作爱好者。对初学者而言, 可迅速地设计出颇具水准的个人网页; 对于有一定制作经验的读者, 也可以由此深入掌握应用 CSS 制作动态网页的方法。

本书精选出 55 个案例, 这些案例可以指导读者学习、掌握 CSS 技术, 体会、积累动态网页制作的经验。同时, 这些案例非常实用, 读者可以直接“拿来”或者简单修改后即加入自己的网页代码中, 轻轻松松便可使自己的网页多姿多彩、变幻莫测, 达到令人惊叹的美观布局和令人震撼的动态效果。

本书具有如下特点:

- 深入浅出, 可操作性强: 将 CSS 的知识点有机地分散在 55 个案例中。通过案例, 使得 CSS 众多的对象、属性以及各种设置和操作都变得具体形象、通俗易懂。让学习者从实际的操作、具体的设计中体会、领悟、积累 CSS 和动态网页制作的经验技巧。

- 循序渐进：本书从最开始的 HTML 基础讲起，这时读者可以是一个完全的门外汉；到 CSS 的基本概念，到设置字体样式和控制页面布局，这时读者已经被领进门；然后渐入佳境，讲解 CSS 非常重要的用途——定位和叠放；最后讲解 CSS 技术中的高级部分——CSS 滤镜以及 CSS 和脚本语言 JavaScript 的结合使用。至此，读者已经从“入门”到“精通”了。
- 充分考虑到了 CSS 与浏览器的兼容性：由于制订 CSS 标准的国际组织 W3C 已发布了 CSS1 和 CSS2 两个版本的规范，而主要的浏览器又有 Internet Explorer 和 Netscape Navigator 两种，并且它们各自又有从 2.0 版到目前的 6.0 版的多种版本。各种版本的浏览器对 CSS 标准的支持存在很大不同，而且浏览器又各自对 CSS 作出了不同的扩展，这些情况使得 CSS 与浏览器的兼容性问题成为网页设计者不得不考虑的一个极其重要而又极其棘手的问题。本书中充分考虑了这个问题，在附录 1 中对 CSS 规则逐条给出了与之兼容的浏览器版本信息，以及应注意的兼容性问题。同时在附录 2、附录 3 和附录 4 中给出了 CSS2 新增的选择符，以及 CSS 度量颜色和 CSS 颜色值，这些都和兼容性问题有关。解决了这些问题，网页设计者就可以不再为长期困扰的、繁杂的兼容性问题疑惑和头痛了。

这时你会发现，CSS 和动态网页的实现，原来是如此简单！拥有本书，你就会拥有这种感觉。

准备工作

为了使用本书，需要一个支持动态 HTML 的浏览器。本书除了特别指明所需的浏览器环境之外，其余所有的例子都在 Internet Explorer 4 下调试成功。目前 Internet Explorer 浏览器的最新版本是 Internet Explorer 6。Internet Explorer 5 和 Internet Explorer 5.5 已成为目前的主流浏览器。因此，可以说本书中所给出的全部案例在当前的一般环境下都是可用的。Internet Explorer 5 和 Internet Explorer 5.5 是我们推荐使用的浏览器。用户也可以安装 Netscape Navigator 6.0，以测试不同浏览器下的显示效果。

同时，还需要一个顺手的编辑工具。事实上，如果仅仅是学习和测试本书中的案例效果，用户只需要一个文本编辑器就足够了。这可以是 Windows 的记事本或者写字板。

然后，就可以阅读本书。

之后，使用 CSS 技术，制作动态网页，一切都会容易起来。

编者

2001 年 8 月

目 录

第 1 章 HTML 4.0 简介	1
1.1 初识 HTML	1
1.1.1 HTML 是什么	1
1.1.2 初步尝试	1
1.1.3 HTML 文档的注释	4
1.2 文档头	4
1.2.1 可在文档头部分使用的标识符	5
1.2.2 meta 标识符	5
1.3 网页的页面属性	5
1.3.1 定义背景色、背景图片和文字色彩	6
1.3.2 设置网页空白	6
1.4 链接	7
1.4.1 基本语法	7
1.4.2 在新窗口中打开链接	8
1.4.3 用链接实现发信	9
1.5 字体样式	9
1.5.1 标题字体	9
1.5.2 设置字号	10
1.5.3 设置物理字体和逻辑字体	11
1.5.4 设置字模	12
1.5.5 设置文字颜色	13
1.5.6 字体设置标识符的组合使用	13
1.6 网页布局	13
1.6.1 设置标尺线	13
1.6.2 行的控制	14
1.6.3 文字的对齐方式	15
1.6.4 列表	16
1.6.5 预排格式化文本	19
1.6.6 标识符<address>和</address>	20
1.6.7 标识符<blockquote>和</blockquote>	20
1.6.8 空白区域	20
1.7 图像	21
1.7.1 基本语法	21

1.7.2	图像在网页中的布局.....	22
1.7.3	设置图像的大小和边框宽度.....	24
1.8	表格.....	25
1.8.1	定义表格的基本语法.....	25
1.8.2	设置表格外观.....	26
1.8.3	跨多行多列的表元.....	28
1.8.4	控制表格内文字的布局.....	29
1.8.5	控制表格在网页中的布局.....	30
1.9	多窗口结构.....	30
1.9.1	基本语法.....	30
1.9.2	设置各窗口的尺寸.....	31
1.9.3	框架的嵌套.....	33
1.9.4	框架各窗口之间的关系.....	33
1.9.5	设置框架的外观.....	33
1.9.6	浮动窗口.....	34
1.10	表单.....	35
1.10.1	表单的基本结构.....	35
1.10.2	普通按钮和图像按钮.....	35
1.10.3	下拉菜单.....	36
1.10.4	复选框.....	37
1.10.5	单选按钮.....	38
1.10.6	提交和重置按钮.....	39
1.10.7	文字和密码输入.....	39
1.10.8	文本区域.....	40
1.11	<Script>、<Applet>和<Style>标识符.....	40
1.12	本章知识点回顾.....	41
第 2 章	CSS 基础.....	50
2.1	范例 1：设置链接效果.....	50
2.1.1	效果说明.....	50
2.1.2	样式表的基本结构.....	50
2.1.3	设置链接效果的样式规则.....	52
2.1.4	把样式表嵌入到 HTML 文档头.....	52
2.1.5	样式表和 HTML 的关系.....	53
2.2	范例 2：设置鼠标光标的形状.....	54
2.2.1	效果说明.....	54
2.2.2	用于设置鼠标光标形状的样式规则.....	54
2.2.3	在代码行中加入样式.....	55
2.2.4	指定鼠标光标的形状.....	57

2.3 范例 3：首字下沉与首行大写	59
2.3.1 效果说明	59
2.3.2 分类	60
2.3.3 使用分类实现首字下沉和首行大写	62
2.3.4 情景选择	64
2.3.5 使用情景选择实现首字下沉和首行大写	65
2.4 范例 4：为不同文本定义不同样式	68
2.4.1 效果说明	68
2.4.2 class 属性	68
2.4.3 id 属性	71
2.4.4 span 元素	72
2.4.5 div 元素	76
2.4.6 使用多个样式规则时的优先级	77
2.5 范例 5：外部 CSS 文件	77
2.5.1 将样式表嵌入 HTML 文档的方法	77
2.5.2 链接到外部 CSS 文件	78
2.5.3 输入外部样式表	80
2.6 本章知识点回顾	82
第 3 章 设置字体样式	88
3.1 范例 6：字体风格	88
3.1.1 效果说明	88
3.1.2 设置字体风格的样式规则	88
3.1.3 创建步骤	88
3.2 范例 7：按照字模系列名称调用字体	89
3.2.1 效果说明	89
3.2.2 设置字模系列的样式规则	90
3.2.3 创建步骤	90
3.3 范例 8：自由控制字号	91
3.3.1 控制字号的样式规则	91
3.3.2 以 point 为单位控制字号	92
3.3.3 以 em 为单位控制字号	92
3.3.4 以 pixel 为单位控制字号	93
3.3.5 以其他单位控制字号	93
3.3.6 使用关键字控制字号	94
3.3.7 使用比例尺寸控制字号	95
3.4 范例 9：控制字重	96
3.4.1 效果说明	96
3.4.2 设置字重的样式规则	96

3.5 范例 10：文字变形.....	97
3.5.1 效果说明.....	97
3.5.2 设置文字变形的样式规则.....	97
3.6 范例 11：文字修饰.....	98
3.6.1 效果说明.....	98
3.6.2 设置文字修饰的样式规则.....	98
3.7 范例 12：设置文字的颜色.....	99
3.7.1 使用英文名称来设置颜色.....	99
3.7.2 使用 16 进制数来设置颜色.....	100
3.7.3 使用 rgb 模式来设置颜色.....	102
3.8 范例 13：设置文字的背景颜色.....	104
3.8.1 设置黑底白字效果.....	104
3.8.2 为多种网页元素设置背景颜色.....	105
3.9 范例 14：设置文字的背景图像.....	107
3.9.1 设置背景图像的样式规则.....	107
3.9.2 设置背景图像的范例.....	107
3.10 范例 15：控制文字是否显示.....	108
3.10.1 display 属性.....	108
3.10.2 范例.....	108
3.11 范例 16：设置列表项的标记.....	109
3.11.1 list-style-type 属性.....	109
3.11.2 使用 list-style-type 属性设置 UL 标记.....	110
3.11.3 使用 list-style-type 属性设置 OL 标记.....	111
3.11.4 list-style-image 属性.....	112
3.11.5 list-style 属性.....	113
3.12 范例 17：笔记本网页效果.....	113
3.12.1 效果说明.....	113
3.12.2 代码.....	114
3.12.3 font 属性.....	116
3.13 本章知识点回顾.....	117
第 4 章 控制页面布局.....	121
4.1 范例 18：设置字母间距和单词间距.....	121
4.1.1 设置字母间距.....	121
4.1.2 设置单词间距.....	123
4.2 范例 19：控制行距.....	124
4.2.1 控制行距的方法.....	124
4.2.2 用数字设置行高.....	124
4.2.3 使用长度单位设置行高.....	126

4.2.4 使用比例设置行高.....	126
4.3 范例 20：让文字互相重叠.....	127
4.4 范例 21：文字对齐.....	128
4.4.1 样式规则.....	128
4.4.2 水平对齐的一个范例.....	129
4.4.3 对表格的各表元内容进行对齐的范例.....	129
4.5 范例 22：文字缩行.....	133
4.5.1 样式规则.....	133
4.5.2 缩行的范例.....	133
4.6 范例 23：设置空白区域.....	135
4.6.1 box 与间隙、边框、空白区域.....	135
4.6.2 设置空白区域的样式规则.....	135
4.6.3 设置空白区域的范例.....	136
4.7 范例 24：设置间隙和边框.....	139
4.7.1 设置间隙的样式规则.....	139
4.7.2 边框的 4 个属性.....	140
4.7.3 设置边框的宽度.....	140
4.7.4 设置边框的颜色.....	141
4.7.5 设置边框的风格.....	142
4.7.6 设置边框的范例.....	142
4.7.7 设置间隙和边框的范例.....	147
4.8 范例 25：设置环绕效果.....	149
4.8.1 设置环绕的样式规则.....	150
4.8.2 设置环绕的范例.....	150
4.9 范例 26：设置背景.....	151
4.9.1 添加背景颜色和背景图像.....	151
4.9.2 控制背景图像是否平铺.....	151
4.9.3 控制背景图像是否固定.....	154
4.9.4 控制背景图像的显示位置.....	155
4.9.5 使用 background 属性设置背景.....	157
4.10 范例 27：错乱的网页效果.....	158
4.10.1 效果说明.....	158
4.10.2 代码.....	158
4.11 范例 28：报纸网页效果.....	162
4.11.1 效果说明.....	162
4.11.2 代码.....	162
4.12 本章知识点回顾.....	167

第 5 章 定位与叠放	172
5.1 范例 29：绝对定位与相对定位	172
5.1.1 绝对定位	172
5.1.2 相对定位	175
5.2 范例 30：设置网页元素的宽度和高度	178
5.2.1 设置宽度	178
5.2.2 同一图像以不同宽度显示	178
5.2.3 阶梯状显示的文本	179
5.2.4 设置高度	180
5.3 范例 31：设置网页元素的可视性	181
5.3.1 visibility 属性	181
5.3.2 范例：突然消失的按钮	182
5.4 范例 32：叠放文字和图像	183
5.4.1 z-index 属性	183
5.4.2 范例：将两张图片组合为一	184
5.5 范例 33：定位方法的综合应用	185
5.5.1 效果说明	185
5.5.2 创建步骤	186
5.6 范例 34：图片文字效果	188
5.6.1 效果说明	188
5.6.2 创建步骤	189
5.7 范例 35：光标瞄准器特效	194
5.7.1 效果说明	194
5.7.2 创建步骤	194
5.8 本章知识点回顾	196
第 6 章 CSS 滤镜 (Filter)	199
6.1 范例 36：认识滤镜	199
6.1.1 什么是滤镜	199
6.1.2 滤镜效果示例	199
6.2 范例 37：定义一个或多个滤镜效果	201
6.2.1 定义一个滤镜效果	201
6.2.2 定义多个滤镜效果	202
6.2.3 多个滤镜效果的应用顺序	203
6.2.4 应用滤镜效果时的注意事项	204
6.3 范例 38：Alpha 通道滤镜效果	206
6.3.1 Alpha 通道效果的效果及定义	206
6.3.2 一个范例	207

6.3.3 代码的解释：使用脚本动态设置滤镜的属性	208
6.3.4 滤镜数组	210
6.4 范例 39：移动模糊（Motion Blur）	213
6.4.1 效果和定义	213
6.4.2 范例	213
6.5 范例 40：色度（Chroma）	216
6.5.1 效果及定义	216
6.5.2 范例	216
6.6 范例 41：下落阴影（Drop Shadow）	217
6.6.1 效果及定义	217
6.6.2 范例	217
6.7 范例 42：翻转（Flip）	218
6.7.1 效果及定义	218
6.7.2 范例	218
6.8 范例 43：光晕（Glow）	218
6.8.1 效果及定义	218
6.8.2 范例	219
6.9 范例 44：灰度（Grayscale）	219
6.9.1 效果及定义	219
6.9.2 范例	220
6.10 范例 45：反相（Invert）	220
6.10.1 效果及定义	220
6.10.2 范例	220
6.11 范例 46：遮蔽（Mask）	221
6.11.1 效果及定义	221
6.11.2 范例	221
6.12 范例 47：阴影（Shadow）	222
6.12.1 效果及定义	222
6.12.2 范例	222
6.13 范例 48：波形（Wave）	223
6.13.1 效果及定义	223
6.13.2 范例	223
6.14 范例 49：X 光效果（X-ray）	224
6.14.1 效果及定义	224
6.14.2 范例	224
6.15 本章知识点回顾	225
第 7 章 JavaScript 与 CSS 的结合应用	230
7.1 范例 50：图片播放效果	230

7.1.1	效果说明.....	230
7.1.2	客户端映射图.....	232
7.1.3	使用“层”使图文重叠.....	234
7.1.4	完整代码.....	234
7.2	范例 51：光晕文字特效.....	236
7.2.1	效果说明.....	236
7.2.2	实现方法.....	237
7.3	范例 52：固定于网页某一位置的菜单.....	239
7.3.1	效果说明.....	239
7.3.2	实现方法.....	240
7.4	范例 53：可拖动的图文.....	243
7.4.1	效果说明.....	243
7.4.2	实现方法.....	244
7.4.3	进一步实践：可拖动的 DIV.....	247
7.4.4	同时兼容 Netscape 和 IE.....	248
7.5	范例 54：神奇的导航图.....	252
7.5.1	效果说明.....	252
7.5.2	客户端映射图.....	254
7.5.3	定位和叠放.....	255
7.5.4	设置和导航菜单各项相对应的显示层（DIV）.....	256
7.5.5	实现图像变形的动画效果.....	257
7.6	范例 55：飞行的图片.....	264
7.6.1	效果说明.....	264
7.6.2	实现方法.....	266
7.7	本章知识点回顾.....	268
附录 1 CSS 与浏览器的兼容性.....		271
f1.1	样式表基础.....	271
f1.2	字体样式.....	272
f1.3	文字.....	274
f1.4	颜色与背景.....	275
f1.5	页面布局.....	276
f1.6	层和定位（CSS2）.....	279
附录 2 CSS2 新增选择符.....		280
f2.1	*（通配符选择符）.....	280
f2.2	>（子选择符）.....	280
f2.3	+（近亲选择符）.....	280
f2.4	[]（属性搭配选择符）.....	281

附录 3 样式表度量单位值	282
f3.1 %	282
f3.2 cm	282
f3.3 em	282
f3.4 ex (x-height)	283
f3.5 in	283
f3.6 mm	283
f3.7 pc	284
f3.8 pt	284
f3.9 px	284
附录 4 样式表颜色值	285
f4.1 英文名称表示的颜色值	285
f4.2 6 个数字的 16 进制颜色值	285
f4.3 3 个数字的 16 进制颜色值	285
f4.4 RGB 颜色值	286
f4.5 使用百分比值的 RGB 颜色值	286

第 1 章 HTML 4.0 简介

主页是 Web 上信息发布的主要形式，由一些 HTML 文本文件组成。HTML (HyperText Markup Language) 即超文本标识语言，是万维网上的标准语言，它把不同计算机上的资源有机地联系在一起，形成一个整体 (即 Web 页)，以方便用户访问。HTML 由 1.0 版本已经发展到今天的 4.0 版本。由于 CSS (全称 Cascading Style Sheets, 层叠样式表, 简称样式表) 是嵌入到 HTML 文档中，以控制网页的外观样式显示的。因此，学习 HTML 是学习 CSS 的基础，本章将系统介绍 HTML 4.0 规范，以打好这个基础。

1.1 初识 HTML

1.1.1 HTML 是什么

HTML 规范定义了一系列标识符，这些标识符用于控制网页的文档格式、字符格式、段落格式等，还包括为达到网页交互效果，而对脚本语言、Java 小程序、CGI 程序等的支持和调用的规则。这些标识符是非常简单易学的，HTML 文档是文本文件，可以由任何一个文本编辑器进行编写，它通过浏览器来解释执行。

动态 HTML (DHTML) 是近年来网络发展进程中，最振奋人心也最具实用性的创新之一。什么是动态 HTML 呢？动态 HTML 作为浏览器自带的功能，实际上只是一种概念，并不属于一种专门技术 (如 JavaScript, ActiveX 等)。它既不是标签、插件，也不是浏览器，而是一种通过各种技术的综合发展而得以实现的概念 (当然，不同的浏览器实现的程度也不同)。这些技术包括 JavaScript、VBScript、Document Object Model (文件目标模块) Layers 和 CSS 等。

HTML 4.0 通过以下机制扩展了 HTML：样式表、脚本、框架、嵌入对象、混合文本方向、功能更强的表格、增强的表单，并且为残疾人提供了访问的可能。HTML 4.0 是一种动态 HTML 规范。

1.1.2 初步尝试

下面开始编写第一个 HTML 文档。使用一个较熟悉的文本编辑器，比如 Windows 自带的写字板，编写如下代码：

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>例 1</title>
</head>
<body>
```

```
<p>例 1：初识 HTML</p>
</body>
</html>
```

编写完后，以.html 或.htm 为后缀名保存文件为 1.htm。双击打开该文件，网页显示效果如图 1.1 所示，其中文档标题将显示在浏览器上方标题栏的最左端。

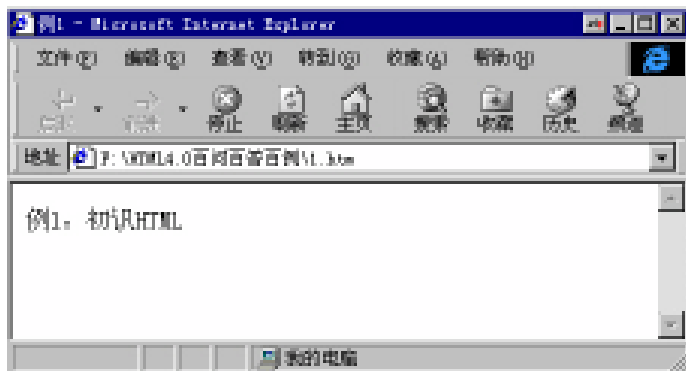


图 1.1 初识 HTML

从上面的示例中，可以看出 HTML 文档的基本结构，如图 1.2 所示。

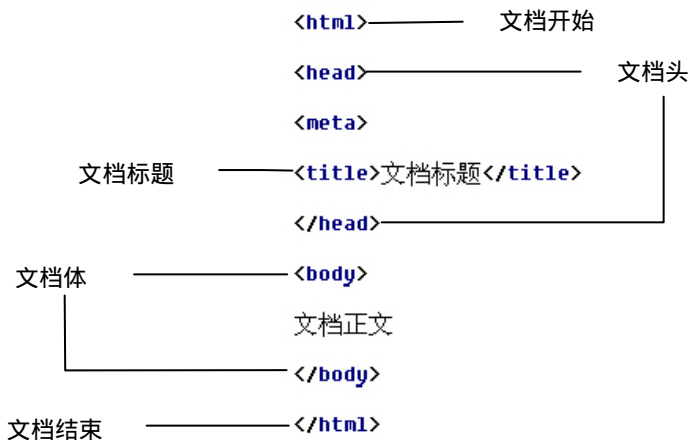


图 1.2 HTML 文档基本结构

HTML 虽然不是“所见即所得”的程序设计语言，但是 HTML 非常易学易用。一个 HTML 文档由一系列标识符组成，这些标识符均包含在尖括号<>中，而且大多成对出现，在成对出现的标识符中，后一个以</标识符>结束。标识符中的内容称为 Element（元素），元素代表了标识符的意义。一对标识符仅对包含在其中的文档部分发生作用。

标识符一般格式为：

```
<element>object</element>
```

或

```
<element attribute=argument>object</element>
```

或

```
<element>
```

HTML 文档总是由标识符<html>开始，然后是文档头部分，由<head>和</head>标识。文档头中可以包含文档的标题 (<title></title>) <meta>属性以及脚本代码等。接着是文档的正文部分，由<body>和</body>标识，最后由</html>结束。当然这些部分并非都是不可缺少的，文档的标题甚至整个文档头都可以省略，结束的标识符</body>和</html>也可以省略。

下面再看一个例子，使用文本编辑器编写一个 HTML 文档，代码如下：

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>文档标题</title>
</head>
<body>
文档正文
</body>
</html>
```

可以做如下尝试：

- 设置文档标题为自己想要的标题，比如“我的主页”。方法是修改<title>和</title>之间的文字即可。
- 在文档体部分写一段文字。方法是将文字插入到<body>和</body>之间。保存文件，然后打开文件观察显示结果，如图 1.3 所示。

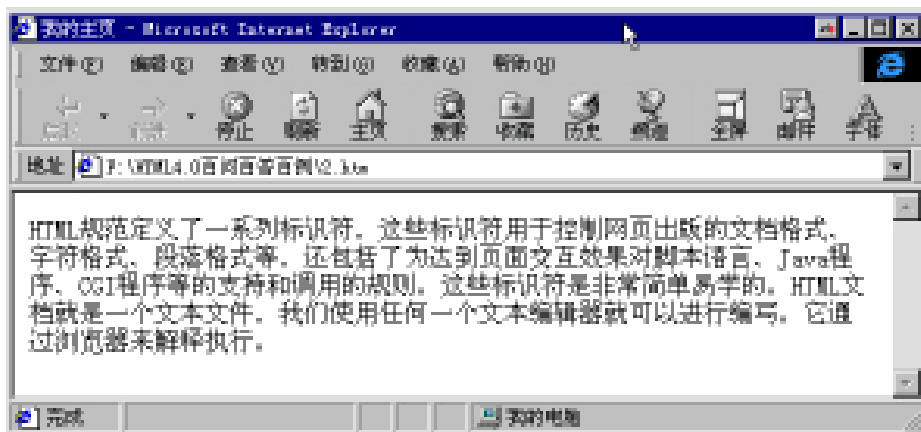


图 1.3 HTML 文档

- 删去文档头部分，即删去<head>和</head>以及它们之间的全部文字。然后保存，并打开文件观察显示结果。
- 删去</body>和</html>标识符。保存文件，然后打开文件观察显示结果。

需要说明的是，HTML 文档的标识符并不区分大小写，即大写、小写或者混合使用都可以。比如：<HTML>和</HTML>，<html>和</html>以及<HTMl>和</htMl>都是有效的。



注意

- (1) 在 HTML 文档中嵌入的脚本代码（比如 JavaScript），通常对变量名是区分大小写的，这时变量 Number 和变量 number 就是不同的两个变量。
- (2) 在 HTML 文档中使用链接、插入图像等调用外部文件时，最好注意大小写，这样具有更好的可移植性。因为虽然 Windows 系统对文件名称不区分大小写，但是 UNIX 系统对大小写是敏感的。

另外，在对标识符的属性赋值时，有时使用了引号，有时则没有。是否一定要使用引号呢？

一般来说，引号并不是必需的。而且，使用单引号、双引号皆可。但是必须前后配对，即如果左引号是单引号，则右引号也应该是单引号。

给标识符的属性赋值时，使用引号是一个好习惯，而且使用引号赋值是 HTML 规范的一个发展趋势。在使用网页制作工具制作生成的 HTML 代码中，都使用引号。

1.1.3 HTML 文档的注释

在 HTML 文档中进行注释的方法有两种：

- 使用标识符<!--和-->可在 HTML 网页内插入注释性的说明文字，包含在这两个标识符之间的代码不会被浏览器处理和显示；
- 在使用标识符<Script>和</Script>进行标识的脚本代码中，使用“//”进行注释。“//”之后的一行文字将不被浏览器处理和显示。

使用注释的目的常常有以下两种：

- 给出提示信息或者描述信息；
- 用于隐藏脚本代码，通常将脚本代码包含在<!--和-->标识符之间。这是因为一些低版本的浏览器有时不支持这些脚本语言，将其写在注释标识符之间，就不会使显示出的网页乱七八糟。而支持这些脚本语言的浏览器仍然会执行这些脚本代码。

1.2 文档头

习惯上，标识符<head>和</head>之间的内容用于描述网页的内容，对网页的外观没有实际的影响，这一部分被称为文档头。