



工业和信息化普通高等教育“十二五”规划教材

21世纪高等教育计算机规划教材

COMPUTER

HTML5 程序设计 基础教程

The Fundamentals of HTML5

■ 李雯 李洪发 主编

■ 闫俊伢 徐会彩 李建华 副主编

— 体系结构合理——基础知识 + 界面设计 + 构建 Web 高级应用

— 教材内容丰富——HTML 5+JavaScript+CSS3+jQuery

— 注重实用效果——结合大量实用案例，提升实战能力



人民邮电出版社
POSTS & TELECOM PRESS

013046891



工业和信息化普通高等教育“十二五”规划教材
21世纪高等教育计算机规划教材

TP312HT-43

02



HTML5 程序设计 基础教程

The Fundamentals of HTML5

■ 李雯 李洪发 主编

■ 闫俊伢 徐会彩 李建华 副主编



北航

C1652625

人民邮电出版社

北京

图书在版编目(CIP)数据

HTML5程序设计基础教程 / 李雯, 李洪发主编. --
北京: 人民邮电出版社, 2013.7
21世纪高等教育计算机规划教材
ISBN 978-7-115-31405-5

I. ①H… II. ①李… ②李… III. ①超文本标记语言
—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2013)第083991号

内 容 提 要

HTML5 是最新的 HTML 标准。尽管 HTML5 到目前为止还只是草案, 离真正的规范还有相当的一段路要走, 但 HTML5 已经引起了业内的广泛兴趣, Google Chrome、Firefox、Opera、Safari 和 Internet Explorer 9 等主流浏览器都已经支持 HTML5 技术。HTML5 无疑会成为未来 10 年最热门的互联网技术。本书涵盖了 HTML5 的各种新特性, 主要内容包括 HTML5 概述、JavaScript、HTML5 表单及文件处理、CSS、拖放、Canvas API 画图、绘制可伸缩矢量图形、播放多媒体、Web 通信、本地存储、开发支持离线的 Web 应用程序、获取浏览器的地理位置信息、支持多线程编程的 Web Workers、jQuery 等。读者在阅读本书时可以充分了解和体验 HTML5 的强大功能。

本书既可以作为大学本科“Web 应用程序设计”课程的教材, 也可作为高职高专院校相关专业的教材, 或作为 Web 应用程序开发人员的参考用书。

-
- ◆ 编 著 李 雯 李洪发
 - 副 主 编 闫俊伢 徐会彩 李建华
 - 责任编辑 邹文波
 - 责任印制 彭志环 杨林杰
 - ◆ 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 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 中国铁道出版社印刷厂印刷
 - ◆ 开本: 787 × 1092 1/16
 - 印张: 24.75 2013 年 7 月第 1 版
 - 字数: 650 千字 2013 年 7 月北京第 1 次印刷
-

定价: 49.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

互联网技术的不断发展和普及已经改变了人们的工作和生活习惯，电子商务已经成为许多企事业单位的业务发展方向。因此，如何开发 Web 应用程序，设计精美、独特的网页已经成为当前的热门技术之一。高校的许多专业都开设了相关的课程。

HTML5 是最新的 HTML 标准。之前的版本 HTML4.01 于 1999 年发布。10 多年过去了, 互联网已经发生了翻天覆地的变化。原有的标准已经不能满足各种 Web 应用程序的需求。编者在多年开发 Web 应用程序和研究相关课程教学的基础上编写了本书。全书内容分为 5 个部分。第 1 部分介绍基础知识, 由第 1、2 章组成, 介绍了 HTML5 基础和 JavaScript 编程基础; 第 2 部分介绍 HTML5 设计网页界面的相关技术, 由第 3 章、第 4 章和第 5 章组成, 比较详尽地讲解了 HTML5 表单及文件处理、最新版本的层叠样式表 CSS3 和 HTML5 拖动技术; 第 3 部分介绍 HTML5 画图和多媒体的相关技术, 由第 6 章、第 7 章和第 8 章组成, 包括使用 Canvas API 画图、绘制可伸缩矢量图形 SVG 和播放多媒体技术; 第 4 部分介绍构建桌面式 Web 应用的相关技术, 由第 9 章、第 10 章和第 11 章组成, 包括 Web 通信、本地存储和开发支持离线的 Web 应用程序技术。第 5 部分为高级应用, 由第 12 章、第 13 章和第 14 章组成, 包括获取浏览器的地理位置信息、支持多线程的 Web Workers 和 jQuery 程序设计。另外, 本书每章都配有相应的习题, 帮助读者理解所学习的内容, 使读者加深印象、学以致用。

本书提供教学 PPT 课件和源程序文件等，需要者可以登录人民邮电出版社教学服务与资源网（<http://www.ptpedu.com.cn>）免费下载。

本教材在内容的选择、深度的把握上充分考虑初学者的特点。为了方便初学者阅读和学习,本书在关注 HTML5 最新技术的同时,介绍了 HTML4 和 JavaScript 基础。因为 HTML5 兼容 HTML4,所以了解 HTML4 基础对学习 HTML5 有很大帮助,比直接学习 HTML5 更易于理解和接受。而 HTML5 提供的 API 则需要在 JavaScript 程序中调用,因此具备必要的 JavaScript 知识也是阅读本书的前提。内容安排上力求做到循序渐进,不仅适合教学,也适合开发 Web 应用程序的各类人员自学使用。

为了提高读者的实战能力，本书还介绍了与 HTML5 相关的热点技术，例如 CSS3 和 jQuery。HTML5、CSS3、jQuery 被称为未来 Web 应用的三驾马车，是设计网页特效的最新技术、也是读者最感兴趣的技术组合。

本书由中北大学李雯、中国人民解放军后勤学院李洪发任主编，山西大学商务学院闫俊伢、大同大学徐会彩、大同大学李建华任副主编。其中，李雯编写了第1章~第3章，李洪发编写了第4章~第6章，闫俊伢编写了第7章~第9章，徐会彩编写了第10章~第12章，李建华编写了第13章、第14章和附录。全书最后由李雯和李洪发共同审定。

由于编者水平有限，书中难免存在不足之处，敬请广大读者批评指正。

编者
2013年3月

目录

第 1 章 HTML5 概述	1
1.1 HTML 基础	1
1.1.1 什么是 HTML	1
1.1.2 HTML 的历史	2
1.2 HTML4 基础	3
1.2.1 设置网页背景和颜色	3
1.2.2 设置字体属性	3
1.2.3 超级链接	5
1.2.4 图像和动画	6
1.2.5 表格	6
1.2.6 使用框架	8
1.2.7 其他常用标签	10
1.3 HTML5 的新特性	12
1.3.1 简化的文档类型和字符集	12
1.3.2 HTML5 的新结构	13
1.3.3 HTML5 的新增内联元素	17
1.3.4 支持动态页面	19
1.3.5 全新的表单设计	22
1.3.6 强大的绘图和多媒体功能	22
1.3.7 打造桌面应用的一系列新功能	23
1.3.8 获取地理位置信息	23
1.3.9 支持多线程	23
1.3.10 废弃的标签	24
1.4 支持 HTML5 的浏览器	25
练习题	26
第 2 章 JavaScript 编程	29
2.1 在 HTML 中使用 JavaScript 语言	29
2.1.1 在 HTML 中插入 JavaScript 代码	29
2.1.2 使用 js 文件	30
2.2 基本语法	30
2.2.1 数据类型	30
2.2.2 变量	30
2.2.3 注释	31
2.2.4 运算符	32
2.3 常用语句	35

2.3.1 条件分支语句	35
2.3.2 循环语句	39
2.4 函数	42
2.4.1 创建自定义函数	42
2.4.2 调用函数	43
2.4.3 变量的作用域	44
2.4.4 函数的返回值	45
2.5 面向对象程序设计	45
2.5.1 面向对象程序设计思想简介	45
2.5.2 JavaScript 内置类	46
2.5.3 HTML DOM	54
2.5.4 Window 对象	55
2.5.5 Navigator 对象	60
2.5.6 document 对象	62
2.6 JavaScript 事件处理	63
2.6.1 常用 HTML 事件	63
2.6.2 Window 对象的事件处理	65
2.6.3 Event 对象	65
练习题	67
第 3 章 HTML5 表单及文件处理	68
3.1 HTML4 表单	68
3.1.1 定义表单	68
3.1.2 文本框	69
3.1.3 文本区域	70
3.1.4 单选按钮	71
3.1.5 复选框	71
3.1.6 组合框	72
3.1.7 按钮	72
3.2 HTML5 表单的新特性	73
3.2.1 新的 input 类型	73
3.2.2 新的表单元素	77
3.2.3 新的表单属性	79
3.2.4 表单验证	80
3.3 文件处理	82
3.3.1 选择文件的表单控件	82
3.3.2 检测浏览器是否支持 HTML5 File API	82

3.3.3	FileList 接口	83	5.3.1	拖放 HTML 元素	138
3.3.4	FileReader 接口	85	5.3.2	拖放文件	139
	练习题	87		练习题	141
第 4 章	最新版本的层叠		第 6 章	使用 Canvas API 画图	142
	样式表——CSS3	88	6.1	Canvas 元素	142
4.1	CSS 基础	88	6.1.1	Canvas 元素的定义语法	142
4.1.1	什么是 CSS	88	6.1.2	使用 JavaScript 获取网页 中的 canvas 对象	143
4.1.2	在 HTML 文档中应用 CSS	89	6.2	坐标与颜色	143
4.1.3	颜色与背景	91	6.2.1	坐标系统	143
4.1.4	设置字体	92	6.2.2	颜色的表示方法	143
4.1.5	设置文本属性	93	6.3	绘制图形	145
4.1.6	超链接	99	6.3.1	绘制直线	145
4.1.7	列表	100	6.3.2	绘制贝塞尔曲线	147
4.1.8	表格	102	6.3.3	绘制矩形	149
4.1.9	CSS 轮廓	106	6.3.4	绘制圆弧	152
4.1.10	浮动元素	107	6.4	描边和填充	153
4.2	CSS3 的新技术	109	6.4.1	描边	153
4.2.1	实现圆角效果	109	6.4.2	填充图形内部	156
4.2.2	多彩的边框颜色	113	6.4.3	渐变颜色	157
4.2.3	阴影	114	6.4.4	透明颜色	162
4.2.4	背景图片	115	6.5	绘制图像与文字	163
4.2.5	多列	117	6.5.1	绘制图像	163
4.2.6	嵌入字体	119	6.5.2	输出文字	166
4.2.7	透明度	120	6.6	图形的操作	169
4.2.8	HSL 和 HSLA 颜色表现方法	122	6.6.1	保存和恢复绘图状态	169
4.3	CSS3 应用实例	124	6.6.2	移动	170
4.3.1	HTML5 + CSS3 设计 页面布局	124	6.6.3	缩放	171
4.3.2	设计漂亮的导航菜单	128	6.6.4	旋转	172
4.3.3	设计登录页面	130	6.6.5	变形	172
	练习题	134	6.7	组合和阴影	174
第 5 章	HTML5 拖放	135	6.7.1	组合图形	175
5.1	概述	135	6.7.2	绘制阴影	177
5.1.1	什么是拖放	135	6.8	HTML5 Canvas 应用实例	178
5.1.2	设置元素为可拖放	135	6.8.1	绘制漂亮的警告牌	179
5.1.3	拖放事件	136	6.8.2	动画实例：小型太阳系模型	181
5.2	传递拖曳数据	136		练习题	184
5.2.1	dataTransfer 对象的属性	137	第 7 章	绘制可伸缩矢量	
5.2.2	dataTransfer 对象的方法	137		图形 (SVG)	185
5.3	HTML5 拖放的实例	138	7.1	SVG 概述	185

7.1.1 SVG 的特性	185	练习题	224
7.1.2 XML 基础	185	第 9 章 Web 通信	225
7.1.3 一个 SVG 实例	187	9.1 跨文档消息机制	225
7.1.4 SVG 坐标系	187	9.1.1 检测浏览器对跨文档消息 机制的支持情况	225
7.1.5 在 HTML5 中使用 SVG	188	9.1.2 使用 postMessage API 发送消息	226
7.2 SVG 形状	188	9.1.3 监听和处理消息事件	227
7.2.1 绘制直线	188	9.2 XMLHttpRequest Level 2	228
7.2.2 绘制折线	189	9.2.1 创建 XMLHttpRequest 对象	229
7.2.3 绘制矩形	190	9.2.2 发送 HTTP 请求	229
7.2.4 绘制圆形	190	9.2.3 从服务器接收数据	230
7.2.5 绘制椭圆	191	9.2.4 进行 HTTP 头 (HEAD) 请求	233
7.2.6 绘制多边形	191	9.2.5 超时控制	236
7.2.7 路径	192	9.2.6 使用 FormData 对象 向服务器发送数据	236
7.3 线条和填充	194	9.2.7 使用 FormData 对象 上传文件	239
7.3.1 设置线条的属性	194	9.3 WebSocket	243
7.3.2 填充	197	9.3.1 什么是 Socket	244
7.4 SVG 文本与图片	198	9.3.2 WebSocket API 概述	246
7.4.1 输出文本	198	9.3.3 WebSocket API 编程	246
7.4.2 SVG 图片	201	9.3.4 WebSocket 服务器	248
7.5 SVG 滤镜	201	练习题	251
7.5.1 定义滤镜	202	第 10 章 本地存储	252
7.5.2 应用滤镜	203	10.1 概述	252
7.6 渐变颜色	203	10.1.1 HTML4 的本地数据 存储方式	252
7.6.1 线性渐变	204	10.1.2 HTML5 本地存储技术概述	254
7.6.2 放射性渐变	204	10.2 localStorage	254
7.7 变换坐标系	205	10.2.1 浏览器对 localStorage 的 支持情况	254
7.7.1 视窗变换——viewBox 属性	206	10.2.2 使用 localStorage 保存数据	255
7.7.2 用户坐标系的变换—— transform 属性	206	10.2.3 获取 localStorage 中的数据	255
练习题	208	10.2.4 删除 localStorage 中的数据	257
第 8 章 播放多媒体	210	10.2.5 storage 事件	257
8.1 HTML5 音频	210	10.3 sessionStorage	259
8.1.1 audio 标签	210	10.3.1 判断浏览器是否支持 sessionStorage	259
8.1.2 播放背景音乐	211	10.3.2 使用 sessionStorage 保存数据	260
8.1.3 设置替换音频源	212		
8.1.4 使用 JavaScript 语言访问 audio 对象	212		
8.2 HTML5 视频	216		
8.2.1 video 标签	216		
8.2.2 使用 JavaScript 语言访问 video 对象	218		

10.3.3 获取 sessionStorage	
中的数据	260
10.3.4 删除 sessionStorage	
中的数据	261
10.4 webSQL Database API	261
10.4.1 判断浏览器是否支持	
webSQL Database API	261
10.4.2 新建数据库	262
10.4.3 执行 SQL 语句	262
10.5 IndexedDB	266
10.5.1 数据库的相关概念	266
10.5.2 判断浏览器是否	
支持 IndexedDB	267
10.5.3 创建和打开数据库	268
10.5.4 创建对象存储空间	
ObjectStore	268
10.5.5 创建索引	271
10.5.6 事务	272
10.5.7 游标	275
练习题	280

第 11 章 开发支持离线的

Web 应用程序

11.1 HTML5 离线 Web 应用程序概述	281
11.1.1 什么是离线 Web 应用程序	281
11.1.2 开发离线 Web 应用程序	
需要完成的工作	283
11.2 开发 HTML5 离线 Web 应用程序	283
11.2.1 Application Cache API	283
11.2.2 Cache manifest 文件	284
11.2.3 更新缓存	286
11.2.4 检测在线状态	289
练习题	290

第 12 章 获取浏览器的

地理位置信息

12.1 概述	291
12.1.1 什么是浏览器地理位置	291
12.1.2 浏览器对获取地理位置	
信息的支持情况	291
12.2 获取地理位置信息	292
12.2.1 getCurrentPosition()方法	292

12.2.2 watchPosition()方法	295
12.2.3 clearWatch()方法	296
12.3 数据保护	296
12.3.1 在 Internet Explorer 9 中	
配置共享地理位置	297
12.3.2 在 Chrome 中配置共享	
地理位置	298
12.3.3 在 Firefox 中配置共享	
地理位置	299
练习题	300

第 13 章 支持多线程编程的

Web Workers

13.1 概述	301
13.1.1 什么是线程	301
13.1.2 什么是 HTML5 Web Workers	302
13.1.3 浏览器对 Web Workers	
的支持情况	303
13.2 Web Workers 编程	303
13.2.1 创建 Web Workers 对象	303
13.2.2 终止 Web Workers 对象	306
13.2.3 共享线程 (Shared Worker)	306
练习题	308

第 14 章 jQuery 程序设计

14.1 jQuery 基础	309
14.1.1 下载 jQuery	309
14.1.2 初识 jQuery	310
14.2 jQuery 选择器	311
14.2.1 基础选择器	312
14.2.2 层次选择器	314
14.2.3 基本过滤器	318
14.2.4 内容过滤器	320
14.2.5 可见性过滤器	322
14.2.6 属性过滤器	322
14.2.7 子元素过滤器	324
14.3 设置 HTML 元素的属性与	
CSS 样式	325
14.3.1 设置 HTML 元素的属性	325
14.3.2 设置 CSS 样式	328
14.4 表单编程	332
14.4.1 表单选择器	332

14.4.2 表单过滤器	335
14.4.3 表单 API	337
14.5 事件和 Event 对象	340
14.5.1 事件处理函数	340
14.5.2 Event 对象	340
14.5.3 绑定到事件处理函数	343
14.5.4 键盘事件	344
14.5.5 鼠标事件	345
14.5.6 文档加载事件	346
14.5.7 浏览器事件	346
14.6 jQuery 动画	347
14.6.1 执行自定义的动画	348
14.6.2 显示和隐藏 HTML 元素	349
14.6.3 淡入淡出效果	350
14.6.4 滑动效果	354
14.6.5 动画队列	357
练习题	360

附录 实验	362
实验 1 HTML4 基础	362
实验 2 HTML5 的新特性	365
实验 3 JavaScript 编程	367
实验 4 HTML5 表单及文件处理	369
实验 5 使用 CSS3	370
实验 6 HTML5 拖放	372
实验 7 使用 Canvas API 画图	373
实验 8 绘制可伸缩矢量图形 (SVG)	375
实验 9 播放多媒体	377
实验 10 Web 通信	378
实验 11 本地存储	379
实验 12 开发支持离线的 Web 应用程序	381
实验 13 获取浏览器的地理位置信息	382
实验 14 支持多线程编程的 Web Workers	383
实验 15 jQuery 编程	384

第 1 章

HTML5 概述

互联网上的应用程序被称为 Web 应用程序，Web 应用程序使用 Web 文档（网页）来表现用户界面，而 Web 文档都遵循标准 HTML 格式。HTML5 是最新的 HTML 标准。之前的版本 HTML4.01 于 1999 年发布。10 多年过去了，互联网已经发生了翻天覆地的变化。原有的标准已经不能满足各种 Web 应用程序的需求。本章就和读者一起来了解一下最新的 HTML5 标准的概貌。

1.1 HTML 基础

对于初学者而言，在学习 HTML5 之前应该了解一下 HTML 的基础知识。

1.1.1 什么是 HTML

HTML（HyperText Markup Language，超文本标记语言）是通过嵌入代码或标记来表明文本格式的国际标准。用它编写的文件扩展名是.html 或.htm，这种网页文件的内容通常是静态的。

HTML 语言中包含很多 HTML 标记，它们可以被 Web 浏览器解释，从而决定网页的结构和显示的内容。这些标记通常成对出现，例如<HTML>和</HTML>就是常用的标记对，语法格式如下：

<标记名> 数据 </标记名>

本小节将介绍一些基本结构标记。HTML 文档可以分为两部分，即文件头与文件体。文件头中提供了文档标题，并建立 HTML 文档与文件目录间的关系；文件体部分是 Web 页的实质内容，它是 HTML 文档中最主要的部分，其中定义了 Web 页的显示内容和效果。

基本的 HTML 结构标记如表 1-1 所示。

表 1-1

基本的 HTML 结构标记

结 构 标 记	具 体 描 述
<HTML>...</HTML>	标记 HTML 文档的开始和结束
<HEAD>...</HEAD>	标记文件头的开始和结束。HTML 文档的头部中可以包含脚本、CSS 样式表和网页标题等信息。这里指的脚本通常是 Java Script 脚本，具体情况将在第 2 章介绍；关于 CSS 样式表的具体情况将在第 4 章介绍
<TITLE>...</TITLE>	标记文件头中的文档标题

续表

结构 标 记	具 体 描 述
<BODY>...</BODY>	标记文件体部分的开始和结束
<!--...-->	标记文档中的注释部分

【例 1-1】 一个使用基本结构标记文档的 HTML 文档实例。

```
<HTML>
  <HEAD>
    <TITLE> HTML 文件标题.</TITLE>
  </HEAD>
  <BODY>
    <!-- HTML 文件内容 -->
  </BODY>
</HTML>
```

这些标记只用于定义网页的基本结构，并没有定义网页要显示的内容。因此，在浏览器中查看此网页时，除了网页的标题外，其他部分与空白网页没有什么区别。



“<!--”和“-->”是 HTML 文档中的注释符，它们之间的代码不会被解析。

1.1.2 HTML 的历史

1990 年，欧洲原子物理研究所的英国科学家 Tim Berners-Lee 发明了 WWW（World Wide Web）。通过 Web，用户可以在一个网页里比较直观地表示出互联网上的资源。因此，Tim Berners-Lee 被称为互联网之父。

最早的关于 HTML 的公开描述是由 Tim Berners-Lee 于 1991 年发表的一篇叫做《HTML 标签》的文章，其中描述了 18 个元素，这就是关于 HTML 的最简单的设计。其中的 11 个元素还保留在 HTML4 中。

1993 年，Internet 工程任务组（Internet Engineering Task Force，IETF）发布了第 1 部 HTML 规范建议。1994 年，IETF 成立了 HTML 工作组，该工作组于 1995 年完成了 HTML 2.0 设计，并于同年发布了 HTML 3.0，对 HTML 2.0 进行了扩展。

HTML 4.01 发布于 1999 年，直到现在仍然有大量的网页是基于 HTML 4.01 的，它的应用周期超过 10 年，因此是到目前为止，影响最广泛的 HTML 版本。

2004 年，超文本应用技术工作组（Web Hypertext Application Technology Working Group，WHATWG）开始研发 HTML5。2007 年，万维网联盟（World Wide Web Consortium，W3C）接受了 HTML5 草案，并成立了专门的工作团队，于 2008 年 1 月发布了第 1 个 HTML5 的正式草案。

尽管 HTML5 到目前为止还只是草案，离真正的规范还有相当的一段路要走，但 HTML5 还是引起了业内的广泛兴趣，Google Chrome、Firefox、Opera、Safari 和 Internet Explorer 9 等主流浏览器都已经支持 HTML5 技术。

2010 年，时任苹果公司 CEO 的乔布斯发表了一篇名为《对 flash 的思考》的文章，指出随着 HTML5 的完善和推广，以后再观看视频等多媒体时就不再依靠 flash 插件了。这引起了主流媒体

对 HTML5 的兴趣。

目前，HTML5 的标准草案已进入了 W3C 制定标准 5 大程序的第 1 步。预期要到 2022 年才会成为 W3C 推荐标准。HTML5 无疑会成为未来 10 年最热门的互联网技术。

1.2 HTML4 基础

HTML5 是在 HTML 4.01 的基础上进行的升级和扩充，它保留了大多数 HTML4 的标签和功能。为了便于读者全面了解 HTML5，本节介绍 HTML4 的基础知识，这些内容也是进行 HTML 编程和阅读本书的基础。

1.2.1 设置网页背景和颜色

在设计网页时，首先需要设置网页的属性。常见的网页属性就是网页的颜色和背景图片。可以在<BODY>标签中通过 background 属性设置网页的背景图片，例如：

```
<BODY background="Greenstone.bmp">
```

可以在<BODY>标签中通过 bgcolor 属性设置网页的背景图片，例如：

```
<BODY bgcolor="#00FFFF">
```

<BODY>标签中的常用属性如表 1-2 所示。

表 1-2 <BODY>的常用属性	
属 性	说 明
BACKGROUND	文档的背景图像
BGCOLOR	文档的背景色
TEXT	文档中文本的颜色
LINK	文档中链接的颜色
VLINK	文档中已被访问过的链接的颜色
ALINK	文档中正被选中的链接的颜色

1.2.2 设置字体属性

可以使用...标签对网页中的文字设置字体属性，包括选择字体和设置字体大小等，例如：

```
<font face="黑体" size="4">设置字体.</font>
```

face 属性用于设置字体类型，size 属性用于设置字体大小。也可以使用 color 属性设置字体的颜色。

还可以设置文本的样式，包括加粗、倾斜和下划线等。使用...定义加粗字体，使用<i>...</i>定义倾斜字体，使用<u>...</u>定义下划线字体。这些标签可以混合使用，定义同时具有多种属性的字体。

【例 1-2】 定义加粗、倾斜和下划线字体，代码如下：

```
<p><b>加粗</b> <i>倾斜</i> <u>下划线</u></p>
```

上面代码定义的网页如图 1-1 所示。

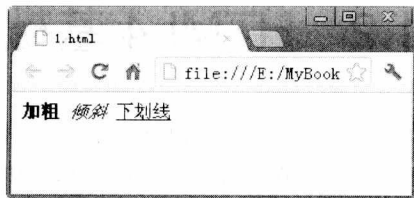


图 1-1 浏览【例 1-2】的结果

在【例 1-2】的代码中，可以看到一对<p>...</p>标签，它们用于定义字体的分段。可以单独定义<p>和</p>之间元素的属性。比较常用的属性是 align=#，#可以是 left、center 或 right。Left 表示文字居左，center 表示文字居中，right 表示文字居右。

【例 1-3】 将【例 1-2】定义的文字居中显示，代码如下：

```
<p align="center"><b>加粗</b> <i>倾斜</i> <u>下划线</u></p>
```

也可以通过选择样式来设置字体。HTML 语言中有一些默认样式，标题是常用的样式之一。标题元素有 6 种，分别为 H1、H2...H6，用于表示文章中的各种题目。标题号越小，字体越大。

【例 1-4】 下面的代码可以定义 H1、H2...H6 标题的文字。

```
<h1>这是标题 1</h1>
<h1>这是标题 2</h1>
<h2>这是标题 3</h2>
<h4>这是标题 4</h4>
<h5>这是标题 5</h5>
<h6>这是标题 6</h6>
```

浏览【例 1-4】的结果如图 1-2 所示。

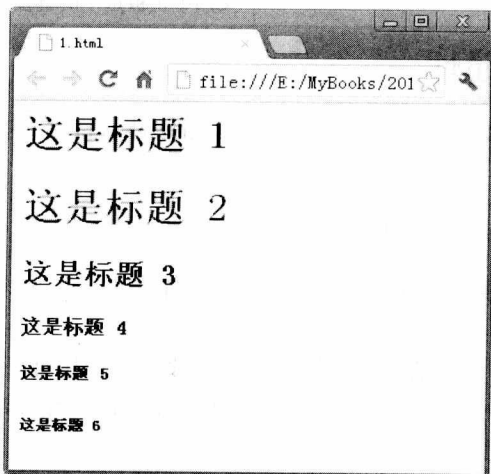


图 1-2 浏览【例 1-4】的结果

1.2.3 超级链接

超级链接是网页中一种特殊的文本，也称为超链接，通过单击超级链接可以方便地转向本地或远程的其他文档。超级链接可分为两种，即本地链接和远程链接。本地链接用于连接本地计算机的文档，而远程链接则用于连接远程计算机的文档。

在超级链接中必须明确指定转向文档的位置和文件名。可以使用 URL（统一资源定位器，Uniform Resource Locator）指定文档的具体位置，它的构成如下：

```
protocol:// machine.name[:port]/directory/ filename
```

其中 protocol 是访问该资源所采用的协议，即访问该资源的方法，主要的协议包括：

- HTTP，超文本传输协议，该资源是 HTML 文件。
- File，用于访问本地计算机上的文件资源。
- FTP，文件传输协议。
- News，表明该资源是网络新闻。

machine.name 是存放该资源主机的 IP 地址或域名，如 www.microsoft.com。port 端口号，是服务器在该主机所使用的端口号。一般情况下端口号不需要指定，默认为 80。只有当服务器所使用的端口号不是默认的端口号时才指定。

directory 和 filename 是该资源的路径和文件名。

下面是一个典型的 URL：

```
http://www.php.net/download.php
```

通常网站都会指定默认的文档，所以直接输入 http://www.php.net 就可以访问到 PHP 网站的首页文档。

下面是一个定义超级链接的例子：

```
<a href="http://www.php.net">PHP 网站</a>
```

在<a>和标签之间定义超级链接的显示文本，href 属性定义要转向的网址或文档。

在超级链接的定义代码中，除了指定转向文档外，还可以使用 target 属性来设置单击超级链接时打开网页的目标框架。可以选择_blank（新建窗口）、_parent（父框架）、_self（相同框架）和_top（整页）等目标框架。比较常用的目标框架为_blank（新建窗口）。

【例 1-5】 定义一个新的超级链接，显示文本为“在新窗口中打开 PHP 网站”，代码如下：

```
<a target="_blank" href="http://www.php.net">在新窗口中打开 PHP 网站</a>
```

如果没有使用 target 属性，单击超级链接后将在原来的浏览器窗口打开新的 HTML 文档。

在 HTML 语言中，电子邮件超级链接的定义代码如下：

```
<a href="mailto:johnney2008@sina.com">我的邮箱</a>
```

超级链接还可以定义在本网页内跳转，从而实现类似目录的功能。比较常见的应用包括在网页底部定义一个超级链接，用于返回网页顶端。首先需要在跳转到的位置定义一个标识（锚），在 DreamWeaver 中这种定义位置的标识被称为命名锚记（在 FrontPage 中被称为书签）。


```

        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
</table>

```

 是 HTML 语言中的空格。border 属性用于定义表格边框的宽度。浏览【例 1-6】的结果如图 1-3 所示。

下面介绍表格的常用属性。

1. 通栏

被合并的单元格会跨越多个单元格，这种合并的单元格被称为通栏。通栏可以分为横向通栏和纵向通栏两种，<td colspan=#>用于定义横向通栏，<tr rowspan=#>用于定义纵向通栏。#表示通栏占据的单元格数量。

2. 表格大小和边框宽度

在<table>标记中表格的大小用 width=#和 height=#属性说明。前者为表宽，后者为表高，#是以像素为单位的整数，也可以是百分比。在【例 1-6】中，可以看到 width 属性的使用。

边框宽度由 border=#属性定义，#为宽度值，单位是像素。例如，下面的 HTML 代码定义了一个边框宽度为 4 的表格。

```

<table border="4" width="100%" id="table1">
.....
</table>

```

3. 背景颜色

在 HTML 语言中，可以使用 bgcolor 属性设置单元格的背景颜色，格式为 bgcolor=#。#是 16 进制的 6 位数，格式为 rrggbb，分别表示红、绿、蓝三色的分量，或者是 16 种已定义好的颜色名称。

【例 1-7】 下面的 HTML 代码定义表格的背景颜色为 c0c0c0（灰色）。

```

<table border="1" width="100%" id="table1">
    <tr>
        <td colspan="2" bgcolor="#C0C0C0">
            <p align="center">表格</p>
        </td>
    </tr>
    <tr>
        <td bgcolor="#C0C0C0">
            <p align="center">域名</p>
        <td bgcolor="#C0C0C0">
            <p align="center">说明</p>
        </td>
    </tr>
.....
</table>

```

浏览【例 1-7】的结果如图 1-4 所示。

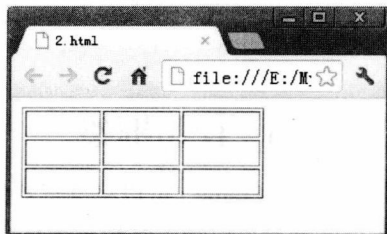


图 1-3 浏览【例 1-6】的结果

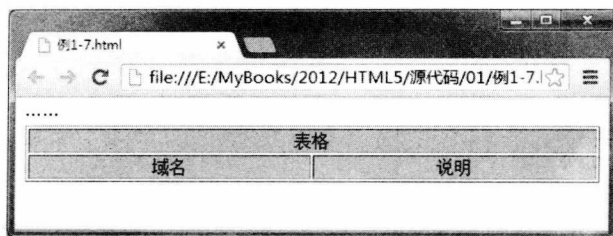


图 1-4 浏览【例 1-7】的结果

1.2.6 使用框架

框架 (Frame) 可以将浏览器的窗口分成多个区域, 每个区域可以单独显示一个 HTML 文件, 各个区域也可以相关联地显示某一个内容。例如, 可以将索引放在一个区域, 文件内容显示在另一个区域。框架通常的使用方法是在一个框架中放置可供选择的链接目录, 而将 HTML 文件显示在另一个框架中。

定义框架的基本代码如下:

```
<html>
<head>
<title>...</title>
</head>
<noframes>...</noframes>
<frameset>
<frame src="url">
<frame src="url">
<frame src="url">
.....
</frameset>
</html>
```

1. <noframe>标签

noframe 元素中包含了框架不能被显示时的替换内容。

2. <frameset>标签

frameset 元素是一个框架容器, 它将窗口分成长方形的子区域, 即框架。在一个框架设置文档中, <frameset>取代了<body>位置, 紧接<head>之后。

<frameset>的基本属性包括 rows 和 cols, 它们定义了框架设置元素中的每个框架的尺寸大小。rows 值从上到下给出了每行的高; cols 值从左到右给出了每列的宽。

框架是可以嵌套的, 也就是说在<frameset>中还可以包含一个或多个<frameset>标签。

3. <frame>标签

<frameset>标签包含多个<frame>标签。每个<frame>元素定义一个子窗口。<frame>标签的属性说明如下。

- name: 框架名称。
- src: 框架内容 URL。
- longdesc: 框架的长篇描述。
- frameborder: 框架边框。
- marginwidth: 边距宽度。
- marginheight: 边距高度。