



高等工科学学校适用教材

ASP.NET MVC 实训教程

赵鲁涛 李晔 等 编



机械工业出版社
CHINA MACHINE PRESS



高等工科学学校适用教材

ASP.NET MVC 实训教程

赵鲁涛 李 晔 汪兆洋 何森雨 编
王 策 杜云飞 薛美美



机械工业出版社

本书介绍了较为先进的 ASP.NET MVC 4 框架,对 MVC 4 进行了深入浅出的讲解,并通过实例实训的方式,让读者将理论联系实践,在动手操作中掌握核心知识;加入了 JQuery、MsChart、Flash 等前端应用技术,并将这些技术与 MVC 紧密结合,使知识和技能不是只停留在表层上,而是上升到深层应用上,提高了教材的整体层次,加强了学生对实际项目中实用技能的学习;充分发挥编者在实际项目中的经验,以多个实际工程项目为实例,从软件工程的角度出发,将整个工程从需求到最后完成的整个过程的关键技术完全复现在书中,不仅从技术上做到深化,而且从实际工程方面给学生更为清晰的范例,使学生能真实地感受到实际项目中 MVC 的具体应用。

图书在版编目 (CIP) 数据

ASP.NET MVC 实训教程/赵鲁涛等编. —北京:机械工业出版社, 2015. 6

高等工科学校适用教材

ISBN 978-7-111-49853-7

I. ①A… II. ①赵… III. ①网页制作工具—程序设计—高等学校—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字 (2015) 第 067304 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:郑 玫 责任编辑:郑 玫 罗子超

版式设计:赵颖喆 责任校对:王 欣

封面设计:鞠 杨 责任印制:刘 岚

北京中兴印刷有限公司印刷

2015 年 6 月第 1 版第 1 次印刷

184mm × 260mm · 19.5 印张 · 484 千字

标准书号:ISBN 978-7-111-49853-7

定价:45.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线:(010)88379833

机工官网:www.cmpbook.com

读者购书热线:(010)88379649

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

封面防伪标均为盗版

金书网:www.golden-book.com

前 言

随着互联网的发展, Web 开发技术得到了迅猛发展, 软件行业对 Web 应用软件开发人员的要求也越来越高。因此, 如何快速掌握目前开发所使用的主流技术和产品, 并应用于现实生活和实际工作中, 成为迫切需要解决的问题。为了适应这种需求, 各高等院校开设了计算机相关实训课程。计算机相关专业的学生在学习计算机技术的过程中, 需要通过生产实习, 才能初步接触社会和认识社会, 经受实际工作的基本训练, 学会收集和整理信息资料, 培养理论联系实际的能力, 综合运用所学的知识, 分析在科研或工程中实际遇到的问题, 掌握解决实际问题的基本思路和方法。

ASP.NET MVC 是现阶段主流的开发框架, 而国内目前对于 ASP.NET MVC 应用讲解的教材较少, 大多数已有教材仅从架构角度进行描述, 对于学生学习和教师讲授都不方便。有些教材没有将学生的学习特点考虑到其中, 更多的是对有编程基础的程序员的一种晋级培训, 无法为初学者提供一个较为完善的学习规范。

鉴于以上原因, 编者编写了本书。本书主要参考北京科技大学计算机科学与技术系多名优秀教师的讲义以及反映较好的国内外相关书籍。编者主持和参与了多个教学科研项目, 加上十几年的教学实践, 积累了不少实际经验, 对这门课程有了更深一层的领悟。本书的主要特点是:

(1) 介绍了较为先进的 ASP.NET MVC 4 框架, 对 MVC 4 进行了详解, 另外, 通过实例实训的方式, 理论联系实际, 让读者在动手操作中掌握核心知识。

(2) 加入了 JQuery、MsChart、Flash 等前端应用技术, 并将这些技术与 MVC 紧密结合。

(3) 充分利用编者在实际项目中的经验, 以多个实际工程项目为实例, 从软件工程师的角度出发, 将整个工程从需求到最后完成的整个过程的关键技术完全复现在书中。

“学以致用, 以用致学”是编者的教学理念, “学会学习”是学生在在学习过程中的更高级要求, 本书正是在这样的背景下, 对 MVC 技术进行实例讲解, 满足实践类课程的教学要求, 让学生不断学习和掌握新知识。随着 MVC 技术的发展和成熟, 本书亦将会不断更新和完善。

编 者

前言

ASP.NET 作为现在主流的开发平台，其 Web 开发一直采用 WebForm 模式，开发人员在体验着 WebForm 模式带来便利的同时，也体会到了视图与业务耦合造成的代码混乱。ASP.NET MVC 4 是微软公司官方提供的以 MVC 模式为基础的 ASP.NET Web 应用程序框架，其技术本身有着灵活、开发周期短、可重用性高等优点。

本书读者对象

本书由浅入深地对 ASP.NET MVC 4 进行了全面讲解，非常适合没有接触过编程的初学者学习使用，也适合作为计算机专业相关课程的教材使用。本书适合下列类型的读者使用：

- (1) 刚接触 MVC 4 开发，对 MVC 4 还不甚了解的初学者。
- (2) 对于 MVC 4 有一定的了解，但是并不娴熟，而且没有太多 MVC 4 开发经验的读者。
- (3) 希望找一本可以用于教授他人 MVC 4 开发方法的读者。

本书内容

本书从教学实际需求出发，合理安排知识结构，由浅入深、循序渐进地介绍较为先进的 ASP.NET MVC 4 框架。全书共分为 10 章，主要内容如下。

第 1 章 MVC 4 简介，主要介绍一般页面制作时需要掌握的知识、内容、思想、结构，并从 HTML、JavaScript、ASP.NET 的应用和 MVC 的原理出发，逐步深入讲解有关知识。通过本章的学习，读者能够对 MVC 4 有个大致的认识。

第 2 章模型，介绍 MVC 框架中模型（Model）层的执行机制，并介绍在 Visual Studio 2012 中如何利用 ADO.NET 实体模型来构造实体数据模型。在此基础上，介绍自定义数据模型和数据检验的相关内容。

第 3 章控制器，介绍控制器的功能与创建，Action 的处理流程，Action 在处理视图层和模型层之间的交互时的 4 种典型处理模式，以及 Action 的常见标签等内容。

第 4 章路由，介绍路由的基础知识，路由的解析、注册与管理，以及 ASP.NET MVC 执行生命周期等内容。

第 5 章视图，介绍如何写好页面的跳转逻辑以及运用一些原本在 WebForm 里面的 HTML 标签，使程序员可以顺利地 WebForm 过渡到 MVC 的设计逻辑之中。

第 6 章 ActionResult 类，介绍如何利用 ASP.NET MVC 4 新增的区域（Area）机制构建较大的工程项目，以及如何将独立性较高的功能切割成多个 ASP.NET MVC 子网站，以降低网站之间的耦合性，降低在多人同时开发一个项目时发生冲突的概率。

第7章 JavaScript 与 JQuery 技术, 介绍常用的前台技术: JavaScript 和 JQuery。从两种技术的基础开始介绍, 逐步深入到高级应用, 并汇集一些高级应用的范例使之更容易使用。

第8章 JQuery 高级应用, 介绍利用 JQuery 编写的控件: zTree 和 JQGrid。本章主要内容包括这些树形控件和表格控件的调用方式、语句格式、作用以及使用它们可以达到何种效果等。

第9章 AJAX 技术, 介绍从 AJAX 原理到 JQuery AJAX 中3种较为常用的实现方式, 以及如何利用 AJAX 方式实现多属性查询功能。

第10章服务器 (IIS) 的配置与使用, 选用 IIS 7 作为部署讲解的对象, 通过一步步讲解 IIS 的安装、属性与配置, 以及工程的发布, 使读者清楚整个软件发布的流程。

本书图文并茂、通俗易懂、结构合理、内容丰富, 在讲解每一部分时都给出相应的实例和表格进行说明, 使读者更加容易理解并掌握相关知识。此外, 本书侧重于技术层面, 偏重应用, 每一章后面都配有习题和综合应用, 让读者在实践中巩固所学理论知识, 快速提高操作技能。本书内容可按80学时讲授。本书第1、2、3、5、10章由赵鲁涛、王策、薛美美编写, 第4、6、9章由李晔、何森雨、杜云飞编写, 第7、8章由汪兆洋、何森雨、赵鲁涛、李晔编写。赵鲁涛负责本书的构思和统稿。

本书的编写得到了“十二五”期间北京科技大学教材建设经费资助和北京高等学校青年英才计划项目 (YETP0386) 的资助。在本书写作过程中, 还得到了北京科技大学信息与计算科学系有关领导和同志的热情帮助和大力支持, 在此表示衷心的感谢! 同时, 还要感谢北京科技大学李安贵教授和中国科学院软件研究所吴文玲研究员对本书提出的宝贵建议。

本书参考了中国建站之家 (www.jz123.cn/), .Net 源码服务专家 (www.51aspx.com/Search/MVC4), jQuery (jquery.com), W3School (www.w3school.com.cn), zTree—jQuery 树插件 (www.ztree.me/v3/main.php#_zTreeInfo), jqGrid Home (<http://www.jqgrid.com/>) 等网站的文章, 在此对作者们一并表示感谢。

由于计算机技术发展日新月异, 而 MVC 正是其中正在发展还未完全成熟的新兴技术, 同时也由于作者的水平所限, 书中错漏之处在所难免, 敬请广大读者批评指正。

编 者

目 录

序	习题	54
前言	综合应用	54
第1章 MVC 4 简介		1
1.1 基础知识		1
1.1.1 HTML		1
1.1.2 CSS		7
1.2 基于 WebForm 的 ASP.NET		9
1.2.1 .NET		9
1.2.2 ASP.NET		9
1.2.3 WebForm 的创建方法		11
1.2.4 ASP.NET 服务器控件		12
1.3 MVC 概念与原理		16
1.3.1 MVC 的概念		16
1.3.2 MVC 的工作原理		16
1.3.3 MVC 架构的优缺点		17
1.4 初识 MVC 4		18
1.4.1 创建新项目		18
1.4.2 MVC 4 项目		21
习题		31
综合应用		31
第2章 模型		34
2.1 模型层概述与执行机制		34
2.2 实体数据模型		35
2.3 LINQ 语句与使用		37
2.3.1 使用 LINQ 的好处		37
2.3.2 LINQ to SQL 的预备知识		38
2.3.3 LINQ to SQL 的查询		40
2.3.4 LINQ to SQL 进行插入		43
2.3.5 LINQ to SQL 进行更新		43
2.3.6 LINQ to SQL 进行删除		44
2.4 模型的数据校验		44
2.4.1 非数据库类 DataAnnotation 启用验证		44
2.4.2 数据库类 DataAnnotation 启用验证		52
第3章 控制器		60
3.1 控制器概述		60
3.2 控制器的创建		62
3.3 Action 的处理流程		64
3.3.1 参数获取		65
3.3.2 参数预处理		66
3.3.3 与模型层的交互		66
3.3.4 结果预处理		66
3.3.5 视图返回		67
3.3.6 实例分析		67
3.4 典型的处理模式		68
3.4.1 单个视图调用多个函数		69
3.4.2 多个视图调用单个函数		71
3.4.3 多个视图调用多个函数		74
3.5 Action 的常见标签		77
3.5.1 NonAction		78
3.5.2 HttpGet 和 HttpPost		79
3.5.3 ChildActionOnly		80
习题		82
综合应用		82
第4章 路由		87
4.1 路由的基础		87
4.1.1 网址路由的作用		87
4.1.2 默认的 Route Table		88
4.2 路由解析		89
4.2.1 非 MVC 控制器类路由解析		90
4.2.2 带单个参数的 MVC 路由地址的解析		90
4.2.3 带多参数的 MVC 路由地址的解析		91
4.3 路由注册		91
4.4 路由管理与匹配机制		93

4.5 MVC 执行的生命周期	94
4.5.1 网址路由比对阶段	94
4.5.2 执行 Controller 的 Action 阶段	95
4.5.3 执行 View 并返回结果页面	95
4.6 总结	95
习题	96
综合应用	96
第5章 视图	97
5.1 视图概述	97
5.2 视图页	98
5.2.1 视图页的创建	98
5.2.2 视图页介绍	101
5.3 从控制器层获取数据的方式	105
5.3.1 弱类型	105
5.3.2 强类型	109
5.3.3 Session 和 Cookies	113
5.4 HtmlHelper 类	114
5.4.1 ActionLink	115
5.4.2 BeginForm 和 EndForm	116
5.4.3 CheckBox	118
5.4.4 DropDownList	119
5.4.5 Hidden	122
5.4.6 Label	123
5.4.7 ListBox	124
5.4.8 Password	125
5.4.9 RadioButton	126
5.4.10 TextArea	127
5.4.11 TextBox	129
5.5 布局页和视图布局页	130
5.5.1 布局页的创建	130
5.5.2 视图布局页的创建	131
5.5.3 布局页和视图布局页介绍	133
5.5.4 布局页的嵌套	135
5.6 分部页	138
5.6.1 分部页的创建	138
5.6.2 分部页介绍	140
5.7 向控制器层传递数据的 方式	144
习题	147
综合应用	150

第6章 ActionResult 类	154
6.1 ActionResult 类概述	154
6.2 ViewResult	156
6.3 PartialViewResult	158
6.4 ContentResult	158
6.5 EmptyResult	160
6.6 FileContentResult、FileStreamResult 和 FilePathResult	160
6.7 JavaScriptResult	163
6.8 JsonResult	165
6.9 RedirectResult	167
6.10 RedirectToRouteResult	168
6.11 HttpUnauthorizedResult 和 HttpNotFoundResult	170
习题	171
综合应用	172
第7章 JavaScript 与 JQuery 技术	173
7.1 JavaScript	173
7.1.1 JavaScript 简介	173
7.1.2 JavaScript 的语法	175
7.1.3 JavaScript 函数	178
7.2 JQuery 简介	179
7.2.1 选择器	180
7.2.2 JQuery 中的文件对象模型与 方法	183
7.2.3 事件处理	186
7.3 JavaScript 与 JQuery 应用 实例	187
7.3.1 iPhone 界面制作	187
7.3.2 使用 JQuery 给 table 动态添加、 删除行	192
7.3.3 使用 JQuery 生成精美的 Tab 按钮	194
7.3.4 使用 JQuery 完成相框效果	200
习题	201
综合应用	202
第8章 JQuery 高级应用	207
8.1 zTree 控件	207
8.2 zTree 的 API	210
8.2.1 API 综述	210

8.2.2 常用 API 详解	215
8.3 zTree 应用实例	222
8.3.1 zTree 基本功能	222
8.3.2 zTree 单选按钮/复选框功能	232
8.3.3 zTree 的拖拽功能	235
8.3.4 zTree 实现节点的增加、删除、 修改功能	237
8.4 JQGrid 表格控件	241
8.4.1 JQGrid 的原理	241
8.4.2 JQGrid 的安装	242
8.4.3 JQGrid 的参数	242
8.4.4 JQGrid 中 ColModel 的 API	246
8.4.5 JQGrid 的代码格式	248
8.5 JQGrid 实例	249
习题	259
综合应用	259

第 9 章 AJAX 技术

9.1 AJAX 概述	261
9.2 原理简介	262
9.2.1 创建对象	262
9.2.2 发送请求	262
9.2.3 获取响应	267

9.2.4 onreadystatechange 事件	269
9.3 JQuery AJAX	269
9.3.1 load()	269
9.3.2 get()	274
9.3.3 post()	277
9.4 综合实例	281
9.4.1 多属性查询	281
9.4.2 分页显示	286
习题	292
综合应用	292

第 10 章 服务器 (IIS) 的配置与

使用	293
10.1 IIS 简介	293
10.2 IIS 安装	293
10.3 IIS 的属性与配置	296
10.4 工程在 IIS 上的发布	298
习题	301
综合应用	302

参考文献

134	CheckBox	2.4.7
135	DropDownList	2.4.8
136	Hidden	2.4.9
137	Label	2.4.10
138	ListBox	2.4.11
139	Password	2.4.12
140	RadioButton	2.4.13
141	Text	2.4.14
142	TextArea	2.4.15
143	TextBox	2.4.16
144	CityList	2.4.17
145	CityList	2.4.18
146	CityList	2.4.19
147	CityList	2.4.20
148	CityList	2.4.21
149	CityList	2.4.22
150	CityList	2.4.23
151	CityList	2.4.24
152	CityList	2.4.25
153	CityList	2.4.26
154	CityList	2.4.27
155	CityList	2.4.28
156	CityList	2.4.29
157	CityList	2.4.30
158	CityList	2.4.31
159	CityList	2.4.32
160	CityList	2.4.33
161	CityList	2.4.34
162	CityList	2.4.35
163	CityList	2.4.36
164	CityList	2.4.37
165	CityList	2.4.38
166	CityList	2.4.39
167	CityList	2.4.40
168	CityList	2.4.41
169	CityList	2.4.42
170	CityList	2.4.43
171	CityList	2.4.44
172	CityList	2.4.45
173	CityList	2.4.46
174	CityList	2.4.47
175	CityList	2.4.48
176	CityList	2.4.49
177	CityList	2.4.50
178	CityList	2.4.51
179	CityList	2.4.52
180	CityList	2.4.53
181	CityList	2.4.54
182	CityList	2.4.55
183	CityList	2.4.56
184	CityList	2.4.57
185	CityList	2.4.58
186	CityList	2.4.59
187	CityList	2.4.60
188	CityList	2.4.61
189	CityList	2.4.62
190	CityList	2.4.63
191	CityList	2.4.64
192	CityList	2.4.65
193	CityList	2.4.66
194	CityList	2.4.67
195	CityList	2.4.68
196	CityList	2.4.69
197	CityList	2.4.70
198	CityList	2.4.71
199	CityList	2.4.72
200	CityList	2.4.73

第 1 章

MVC 4 简介

MVC 模式最早由 Trygve Reenskaug 在 1974 年提出，是施乐帕罗奥多研究中心（Xerox PARC）在 20 世纪 80 年代为程序语言 Smalltalk 发明的一种软件设计模式。本章主要介绍一般页面制作时，需要掌握的知识、思想、结构，并从 HTML、JavaScript、ASP.NET 的应用和 MVC 的原理出发，逐步深入讲解有关知识。

1.1 基础知识

1.1.1 HTML

HTML 作为一种描述网页的语言，是超文本标记语言（Hyper Text Markup Language）的简称。HTML 是由标签（Tag）和属性（Attribute）组成，浏览器读取 HTML 标签和属性，进而解析成网页的形式显示出来，因此我们能在浏览器中方便地查看网页。

HTML 和我们所说的 C 语言、Java 有很大的不同，它是一种标记语言。从浏览器中可以看到文字、图片、视频，还可以听到音频，这些都是网页的内容，只是内容的种类不一样，HTML 就是通过向文本文件中添加标记的方式，告诉浏览器该如何显示需要显示的内容，比如样式如何安排、图片的位置等。

下面通过一个例子了解一下 HTML。我们新建一个文本文档，命名为“测试”，并填入如下内容。

【例 1.1】HTML 测试示例。

```
1      <html>
2          <head>
3              <title>
4                  测试
5              </title>
6          </head>
7          <body>
8              这是一个简单的测试
9          </body>
10     </html>
```

然后将其扩展名改成“html”，双击可以看到是通过浏览器打开的该文件，我们可以看到一个很简单的网页，如图 1.1 所示。



图 1.1 HTML 测试

HTML 元素是由一对尖括号包围的关键词，一般成对出现，比如 `<h1>` 和 `</h1>`。每个标签对中，第一个称为起始标签（开放标签），第二个称为结束标签（闭合标签）。某些没有内容的 HTML 元素称为空元素，即在起始标签中是关闭的，没有结束标签，用起始标签添加斜杠表示，如 `<br/>`。

目前，HTML 不区分大小写，即 `<HTML>`、`<Html>`、`<html>` 是等效的，本章节中使用小写。大多数的 HTML 标签可以嵌套使用，同时 HTML 标签具有对应的属性和属性值。下面是一些常用的 HTML 标签及其对应的属性（w3school, 2013）。

1. 基本标签

(1) `<html> </html>`——HTML 文档标签

表示文档类型为 HTML 类型，并且处于 HTML 文档的最外层，文档中的所有文本和 HTML 标签都包含在内。

(2) `<head> </head>`——文档头部标签

描述文件的头部信息，包括文档的标题、与其他文档关系以及在 Web 中的位置等，故如定义页面中所有连接的基准 URL 的 `<base>` 标签、定义资源引用的 `<link>` 标签、用于定义元信息的 `<meta>` 标签以及文档标题的 `<title>` 标签等可以写在 head 部分，同样 head 标签里也可以包含 JavaScript 和 CSS，其中 JavaScript 和 CSS 分别在第 7 章和 1.1.2 节中给出介绍。例如，在 `<meta>` 标签里加入关键词，就有助于搜索引擎搜索到该网页。

(3) `<title> </title>`——文档标题标签

表示文档的标题，通常将显示在浏览器窗口的标题栏或者状态栏上，若将文档加入收藏夹时，文档标题也作为其默认名称。

(4) `<body> </body>`——文档主体标签

表示文档的主体，包含文档的文本、超链接、图像、表格、列表等内容。

(5) `<!-- 注解 -->`——注释标签

在 HTML 文档中，有时为了方便自己和用户了解文件内容，需要编写注释文件。注释文件的内容并不会显示在浏览器中。

2. 排版标签

(1) `<p> </p>`——段落标签

表示一个段落，在浏览器中以段落的格式显示标签之间的内容。

`<p> </p>` 具有 `align` 属性，用以设置段落的对齐方式。

例如：`<p align="left">你好</p>`，表示“你好”在浏览器中以左对齐的方式显示。

(2)
——换行标签

表示在文档中插入换行符。

(3) <hr />——水平线标签

表示在 HTML 文档中加入一条水平线。<hr> 具有 align、size、color、width 和 noshade (无阴影) 属性, 分别用以设置水平线的水平对齐方式、水平线的高度 (单位为像素)、颜色、宽度和立体效果。

例如: <hr align = "center" color = "red" width = "33" size = "12" noshade />, 表示一条宽 33 像素、高 12 像素、无阴影并且居中对齐的红色水平线。

(4) <div> </div>——块定义标签

表示文档中的分区或者块, 作为块级元素标签, <div> </div> 标签具有 id、class、align、style 等属性, 分别标识元素唯一的 ID、类名、水平对齐方式以及元素的行内样式。下面通过一个例子来展示如何设置 div 标签。

【例 1.2】div 标签设置示例。

```
1 <div style = "border: 5px solid #000000; padding: 10px; margin: 15px;  
2 background-color: #FFFF00; width: 100px; height: 50px; text-align: center;">  
3 第一个 div  
4 </div>  
5 <div style = "background-color: #00FFFF; font-size: 28px; text-decora  
6 tion: underline; font-family:  
7 楷体; color: #FF0000;">  
8 第二个 div  
9 </div>
```

代码中设置的第一个边框为黑色, 边框宽 5 像素 (px), 外边距 10 像素, 内边距 15 像素, 宽 100 像素, 高 50 像素, 背景色为黄色, 居中显示 “第一个 div”。border、padding 和 margin 的关系如图 1.2 所示。

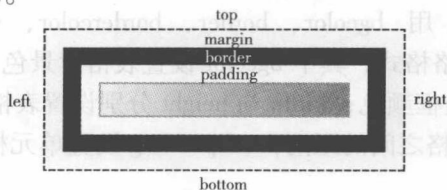


图 1.2 border、padding 和 margin 的关系

“第二个 div” 为红色楷体, 大小为 28 像素, 带下划线, 背景呈水蓝色。

页面运行结果如图 1.3 所示。

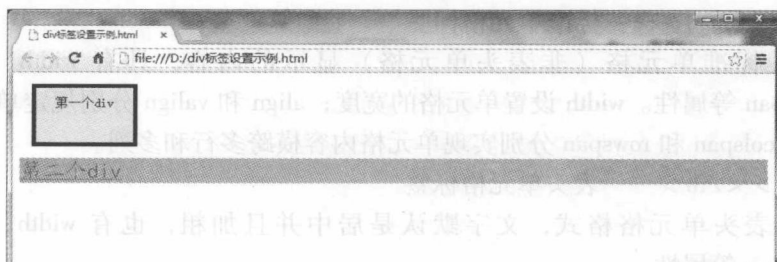


图 1.3 div 标签设置示例运行结果

3. 文本标签

(1) `<h $n$ >` `</h $n$ >`——标题标签

用于设置文本中标题的大小, n 可以取值 1~6, 其中 `<h1>` `</h1>` 对应最大的标题, `<h6>` `</h6>` 对应最小的标题。标题标签间的文字显示为黑体, 并自动插入一个空行。

(2) `<font>` `</font>`——文字格式标签

用于设置文本中文字的格式, 具有 `face`、`size`、`color` 属性, 分别表示文字的字体、大小、颜色。

例如: `<font face="宋体" size="2" color="red">文本文字</font>`, 表示显示为红色宋体且大小为 2 像素的文本文字。

(3) 特殊文字样式标签

HTML 中有设置文字显示为特殊格式的标签, 表 1.1 列出一些常用的特殊文字样式标签。

表 1.1 常用的特殊文字样式标签和显示效果

标签名称	标签示例	显示效果
粗体	<code>你好</code>	你好
斜体	<code><i>你好</i></code>	你好
下画线	<code><u>你好</u></code>	你好
删除线	<code><s>你好</s></code>	你好
强调	<code>你好</code>	你好
上标	<code><sup>你好</sup></code>	你好
下标	<code><sub>你好</sub></code>	你好

4. 表格标签

(1) `<table>` `</table>`——表格标签

用于创建表格, 使用 `bgcolor`、`border`、`bordercolor`、`width`、`height`、`cellspacing`、`cellpadding` 等属性来设置表格格式, 其中 `bgcolor` 设置表格背景色; `border` 设置边框宽度, 默认值为 0; `bordercolor` 设置边框颜色; `width` 与 `height` 分别设置表格的宽度和高度; `cellspacing` 设置单元格的间距, 即单元格之间的距离; `cellpadding` 设置单元格的边距, 即字体与单元格边框的距离。

(2) `<tr>` `</tr>`——单元格行标签

用于定义表格的行, 具有 `align`、`valign` 等属性, 分别设置行的水平对齐方式和垂直对齐方式等。

(3) `<td>` `</td>`——标准单元格标签

用于定义标准单元格 (非表头单元格) 显示的内容, 具有 `width`、`align`、`valign`、`colspan`、`rowspan` 等属性。 `width` 设置单元格的宽度; `align` 和 `valign` 分别规定单元格水平和垂直对齐方式; `colspan` 和 `rowspan` 分别实现单元格内容横跨多行和多列。

(4) `<th>` `</th>`——表头单元格标签

用于设置表头单元格格式, 文字默认是居中并且加粗, 也有 `width`、`align`、`valign`、`colspan`、`rowspan` 等属性。

5. 表单标签

(1) `<form>` `</form>`——表单标签

用以创建所有表单, `<form>` 具有 `action`、`method` 和 `target` 属性。`action` 是必选属性, 定义表单提交的位置 (相对地址或绝对地址); `method` 定义了表单数据传送到服务器的处理方法, 即 `get` 及 `post`, 在以 `get` 方式提交时, 客户端浏览器将每个表单域的 `name` 和 `value` 进行 URL encoding 处理, 而以 `post` 方式提交表单时, 客户端浏览器不会自动处理 URL 中的非法字符 (<http://www.nowamagic.net/libraris/veda/detail/182>, 2013); `target` 属性用来指定目标窗口。

举个简单的例子: `<form method = "post" action = "http://www.ustb.edu.cn/" >`
`</form>`, 则表示将表单处理结果通过 `post` 的方式传送到 `http://www.ustb.edu.cn` 上。

(2) `<input type = "type" name = "name">`——输入区标签

用来定义一个用户输入区, 不同的 `type` 属性值决定不同类型的输入区域, 具体内容见表 1.2。

表 1.2 `type` 取值和相应的输入区域类型

type 取值	输入区域类型
<code><input type = "text" name = "NAME" value = "value" size = "size" maxlength = "length" ></code>	文本输入区域, <code>value</code> 表示初始值, <code>size</code> 表示宽度, <code>length</code> 表示最大输入字符数
<code><input type = "submit" name = "NAME" ></code>	提交表单内容 (到服务器) 的按钮
<code><input type = "reset" name = "NAME" ></code>	重置表单内容的按钮
<code><input type = "checkbox" name = "NAME" value = "value" checked ></code>	复选框, <code>checked</code> 表示默认选中取值为 <code>value</code> 的选项
<code><input type = "radio" name = "NAME" value = "value" checked ></code>	单选框, <code>checked</code> 表示默认选中取值为 <code>value</code> 的选项
<code><input type = "password" name = "NAME" ></code>	密码输入框, 默认显示为 "****"
<code><input type = "image" name = "NAME" src = "name.gif" ></code>	将图像引入页面, 用于替换 <code>submit</code> 按钮, 但是需要添加 <code>onclick</code> 事件

(3) `<select>` `</select>`——列表框标签

用以创建一个列表框。`<select>` `</select>` 标签具有 `multiple`、`name`、`size` 等属性, 其中 `multiple` 属性表示可以有多个选择, `name`、`size` 属性分别用来设置列表框名字和高度。

`<select>` `</select>` 标签中的 `<option>` 标签代表一个选择项。`<option>` 具有 `selected` 和 `value` 属性, `<option selected>` 表示默认的选项, `value` 表示该选项的值。

【例 1.3】 一个基本的列表框的代码。

```
1      <select name = "水果" size = "6" >
2          <option>苹果</option>
3          <option selected>香蕉</option>
4          <option>梨</option></select>
```

创建的列表框默认选择香蕉，效果如图 1.4 所示。

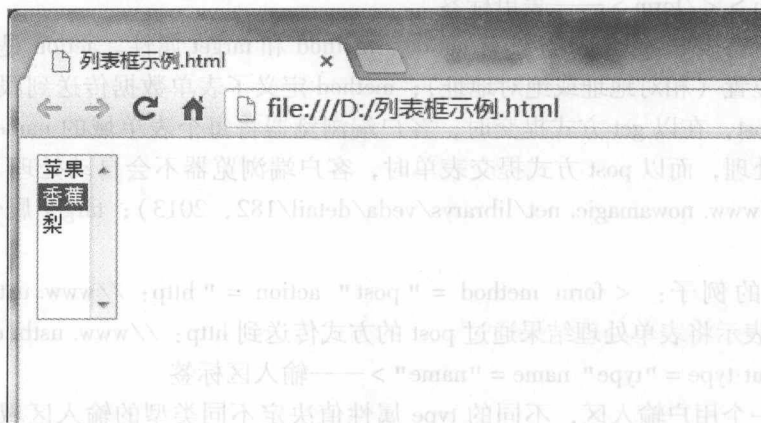


图 1.4 列表框示例

(4) `<textarea>` `</textarea>`——文本区域标签

用以创建一个文本区域，`<textarea>` `</textarea>` 标签具有 `name`、`cols`、`rows` 属性，分别表示文本区域的名称、列数和行数。

6. 列表标签

(1) `<ul>` `</ul>`——无序列表标签

用以创建一个无序的项目列表。`<ul>` `</ul>` 标签具有 `type` 属性，可以设置项目符号的样式。

(2) `<ol>` `</ol>`——有序列表标签

用以创建一个有序的编号列表。`<ol>` `</ol>` 标签具有 `type` 和 `start` 属性，`type` 属性可以改变编号样式本身；`start` 属性则允许改变开始值（默认从“1”开始对有序列表的条目进行编号）。

(3) `<li>` `</li>`——列表的项目标签

用以创建一个列表项。`<li>` `</li>` 标签必须处在 `<ol>` `</ol>` 标签或 `<ul>` `</ul>` 标签之间。

7. 其他标签

(1) `<img>`——图像标签

用以向网页中插入一幅图像，通过 `src` 属性指定的路径（绝对路径或相对路径）从网页上链接到图像。此外，`<img>` 标签还具有 `alt`、`align`、`border`、`width`、`height` 等属性，分别用以设置图像的描述性文本（即当浏览器无法显示图像或者鼠标移动到图像上时显示的内容）、对齐方式、周围边框，以及图像的宽度和高度等。

例如：`<img src = "/i/tulip.jpg" alt = "郁金香" width = "120" height = "100"/>`，表示插入一张格式为 jpg，名称为 tulip，宽为 120 像素，高为 100 像素，文字信息为“郁金香”的图片。

(2) `<a>`——链接标签

用以创建超链接。`<a>` 标签具有 `href`、`name`、`id` 等属性，其中 `href` 属性指向另外一个

文档的链接（或超链接）；name 或 id 属性实现了页面的跳转。

例如：`<a href = "http://www.ustb.edu.cn"> 北京科技大学首页 </a>`，表示创建一个链接到北京科技大学首页的超链接。

上述内容简单介绍了一些常用的 HTML 标签，其他一些重要的标签的语法可以参考 w3school 网站中 HTML 的相关内容。

1.1.2 CSS

CSS（Cascading Style Sheets）即“层叠样式表”或“级联样式表”，也简称为“样式表”。CSS 的引入分离了网页内容与视觉，因此页面维护工作、网页搜索都变得更加容易，而且增强了页面在不同媒介之间的呈现效果。

1. CSS 语法

CSS 定义的基本格式为：

选择符 {属性: 属性值}

其中选择符可以是多种形式的，同定义样式的 HTML 标记，如 body、table 等；对于多个属性的情况，需要用分号将各属性分隔开；若属性值是由多个单词组成的，则需要给该值加引号。

【例 1.4】 一个简单的 CSS 示例。

```
1      P
2      {
3          Color: red;
4          Font-family: "Times New Roman"
5      }
```

上面这段代码的效果为：页面中段落间的文字为红色 Times New Roman 字体。

2. CSS 常用属性及属性值

CSS 的字体、颜色、文本属性及对应的属性值见表 1.3。

表 1.3 CSS 常用属性及属性值

	属性	属性描述	属性值
字体	Font-family	字体	所有可用字体
	Font-style	是否斜体	normal、italic、oblique
	Font-variant	是否小写	normal、small-caps
	Font-weight	字体粗细	normal、bold、bolder、lighter
	Font-size	字体大小	absolute-size、relative-size、length 等
颜色	Color	前景色	颜色值
	Background-color	背景色	颜色值
	Background-image	背景图片	图片路径
	Background-repeat	背景图案重复样式	repeat-x、repeat-y、no repeat
	Background-attachment	设置滚动	scroll、fixed
	Background-position	设置背景图案初始位置	percentage、length、top、left、right、bottom

(续)

	属性	属性描述	属性值
文本	Word-spacing	定义单词间距	normal < length > (以长度为单位)
	Letter-spacing	定义字母间距	normal < length > (以长度为单位)
	Text-decoration	定义文字的“装饰”样式	none、underline、overline、line-through、blink
	Vertical-align	定义元素在垂直方向的位置	baseline super top text-top middle bottom text-bottom < percentage >
	Text-transform	使文本转化为其他形式	capitalize、uppercase、lowercase、none
	Text-align	定义文本对齐方式	left、right、center、justify
	Text-indent	定义文本首行缩进方式	< length >、< percentage >
	Line-height	定义文本行高	normal、< number >、< length >、< percentage >

【例 1.5】CSS 文本属性示例。

```

1 <p style="letter-spacing: 2cm; text-align: justify; text-indent: 4cm;
2 line-height: 17pt; color: red">
3 假如生活欺骗了你 <br/>
4 不要悲伤 不要心急 <br/>
5 忧郁的日子里需要镇静 <br/>
6 相信吧 快乐的日子将会来临 <br/>
7 心儿永远向往着未来 <br/>
8 现在却常是忧郁 <br/>
9 一切都是瞬息 <br/>
10 一切都将会过去 <br/>
11 而那过去了的 <br/>
12 就会成为亲切的回忆 <br/>
13 <p/>

```

上述代码片段中，字间距为 2cm，文本对齐方式为两端对齐，缩进 4cm，行高为 17pt。运行效果如图 1.5 所示。

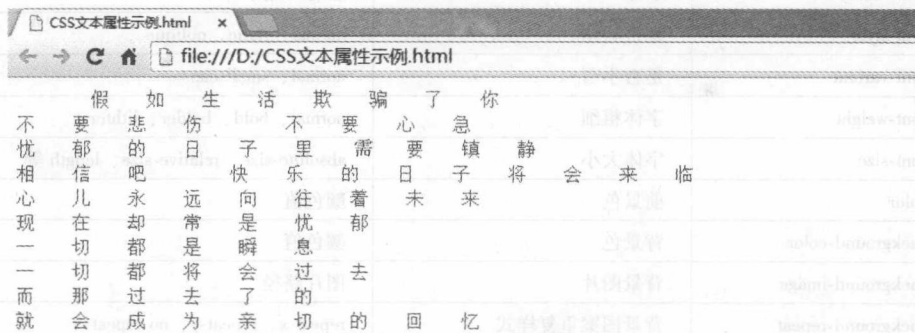


图 1.5 CSS 文本属性示例