

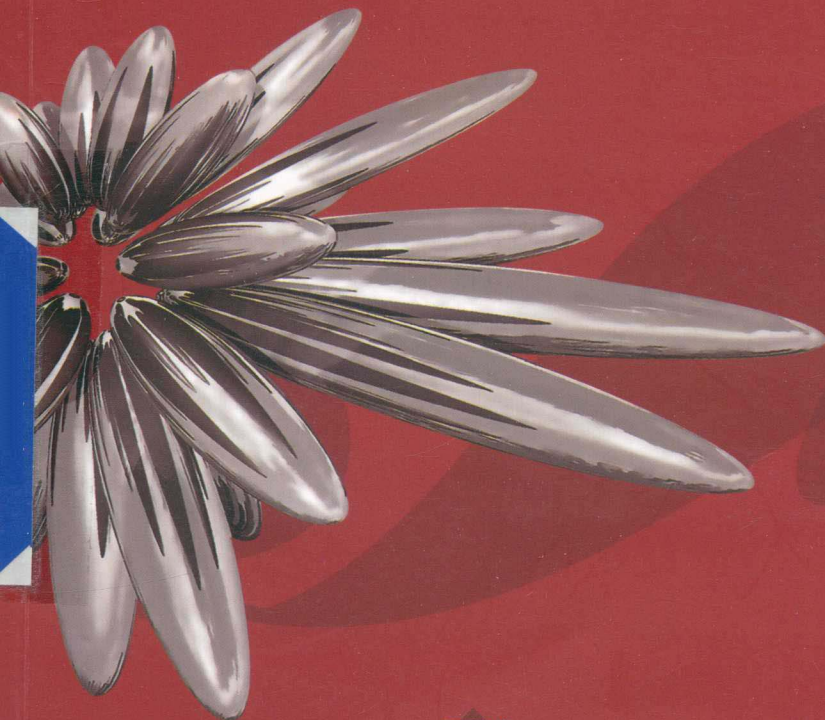


普通高等教育“十三五”规划教材

高等学校计算机规划教材

ASP.NET 4.5 程序设计基础教程 (C#版)

◆ 徐会杰 朱海 王凤科 主 编
◆ 朱丹辉 杨 玫 匡国防 刁文广 副主编



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

普通高等教育“十三五”规划教材
高等学校计算机规划教材

ASP.NET 4.5 程序设计基础教程

(C#版)

徐会杰 朱海 王凤科 主 编
朱丹辉 杨 玫 匡国防 刁文广 副主编

电子工业出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书主要对 ASP.NET 4.5 动态网站开发设计的相关知识进行介绍, 包含 C#语法、ASP.NET 服务器端控件、ADO.NET、AJAX 等。编者多年从事教学与实际项目开发, 对教材章节进行合理安排, 符合教学过程和学生学习的实际需求, 语言通俗易懂、案例典型丰富, 循序渐进地介绍 ASP.NET 程序设计的语言基础、界面设计、编程方法和数据库系统实例开发等方面的内容。

本书从浅显、实用的角度出发, 结合大量案例、代码对 ASP.NET 进行讲解, 并配有相关习题、课件等, 适合初学者作为入门教材, 同时也可供 ASP.NET 开发人员进行查阅。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

ASP.NET 4.5 程序设计基础教程: C#版 / 徐会杰, 朱海, 王凤科主编. — 北京: 电子工业出版社, 2016.2
普通高等教育“十三五”规划教材
ISBN 978-7-121-27328-5

I. ①A… II. ①徐… ②朱… ③王… 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 230213 号

策划编辑: 袁 玺

责任编辑: 郝黎明

印 刷: 三河市华成印务有限公司

装 订: 三河市华成印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 21 字数: 598.4 千字

版 次: 2016 年 2 月第 1 版

印 次: 2016 年 2 月第 1 次印刷

定 价: 45.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010)88254888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010)88258888。

前 言

随着互联网的迅速普及，Web 开发技术也有着日新月异的发展。从 HTML、CGI 到现在的 Web 2.0，在短短 20 多年的时间里，出现了众多优秀的开发技术，其中微软公司（Microsoft Corporation.）推出的 ASP.NET 就是其中的佼佼者。.NET Framework 作为 ASP.NET 的运行基础，为开发人员提供了一个一致的、面向对象的编程环境。从 2000 年 .NET 技术崭露头脚，到 2012 年推出 .NET Framework 4.5，微软为推广 .NET 技术投入了巨大的人力和财力，.NET 也因为其跨语言、跨平台、安全、以及对开放互联网标准和协议支持的优点吸引了全世界越来越多的开发人员。在 .NET Framework 的支持下，ASP.NET 构建的应用程序可以运行在多种平台上。在微软的平台战略支持下，ASP.NET 有着强大的开发工具（Visual Studio）和简单易用的运行环境（Internet Information System），所以 ASP.NET 也是当前最容易上手的网页开发环境。

本书作者根据多年从事 Windows 程序设计和讲授计算机专业相关课程的实际教学经验，以实用目的，精心选取教学内容，合理组织章节结构，抓住 ASP.NET 的知识体系，系统地讲解了各知识点的基础理论和使用方法。在内容设计上，本书以 ASP.NET 核心内容为切入点，降低入门学习难度，将难点以典型案例进行示范，提高学习效率；理论与实际紧密结合。在介绍每一个知识点的同时，均给出相应的代码（读者可按照书中提示信息找到每章的源码），并对同一知识点的不同解决代码进行横向对比，力求让读者在理解基础知识后，能够学以致用，快速上手。本书每章均附有小结和习题，有利于读者拓展思路并对所学知识进行深化理解。

本书共包含五个部分：

第 1 部分 .NET Framework 4.5 简介：包含 .NET Framework 4.5 简介、Visual Studio 2012 集成化开发环境。这一部分通过对 .NET Framework 架构及开发工具的介绍，让读者对 ASP.NET 的开发、运行、调试有一个直观的了解；

第 2 部分 ASP.NET 开发基础：包含 ASP.NET 技术简介、C# 语言基础、ASP.NET 内置对象、ASP.NET 页面语法、ASP.NET 服务器控件。这一部分介绍了开发 ASP.NET 的基础知识，对 C# 语言和服务端对象进行详细介绍。

第 3 部分 构建 ASP.NET 应用程序：包含 ASP.NET 中的样式、主题和母版页、ASP.NET 4.5 中的站点导航技术、用户控件与 Web 部件、ASP.NET 应用程序安全技术。这一部分通过对 ASP.NET 服务器端控件、安全验证控件等的介绍，让读者对 ASP.NET 的服务器端、客户端交互有初步的认识。

第 4 部分 数据访问技术：包含 ADO.NET 数据访问技术、ASP.NET 数据绑定技术与数据绑定控件、LINQ 技术。这一部分重点介绍了 ASP.NET 的数据存储和操作技术，以实例的形式对页面的数据交互进行展示。

第 5 部分 ASP.NET 高级应用：包含 Web 服务和 WCF 服务、ASP.NET 的 AJAX 开发技术、网站发布、打包与安装。这一部分主要介绍了 ASP.NET 的一些高级应用技术，为读者以后进一步的深入学习研究打下基础。

本书面向本专科计算机、非计算机类等相关专业学生，简化学习难度，可以作为学习 Web 应用程序和动态网站开发课程教材，也可以作为相关软件开发人员的学习参考用书。

本书由河南科技大学王凤科负责策划、统稿，本书第 3、4、17 章由河南科技大学徐会杰编写，

第 5~6 章由河南科技大学朱海编写,第 2 章由河南科技大学朱丹辉编写,第 7~9 章由河南科技大学杨玫编写,第 14~16 章由河南科技大学蔡羽编写,第 1、10~12 章由洛阳师范学院匡国防编写,第 13 章由洛阳理工学院刁文广编写,河南科技大学周晓蕴与洛阳理工学院周武强负责校对、审稿。同时,裴创、张军、张晓、杨爱其、董帅、颜同飞、刘海笑等在资料收集、代码调试等方面做了大量的工作,一并表示感谢。

由于编者水平有限、时间仓促,书中不足之处在所难免,敬请广大读者批评指正。衷心希望本书的出版能够对广大读者的学习和工作有所裨益。

目 录

第 1 章 .NET Framework 4.5 简介	1	3.2 ASP.NET 程序结构	17
1.1 .NET Framework 基础概念	1	3.2.1 ASP.NET 文件类型	17
1.1.1 什么是 .NET Framework	1	3.2.2 ASP.NET 目录结构	19
1.1.2 .NET Framework 的目标	1	3.3 ASP.NET 配置	20
1.1.3 公共语言运行时	2	3.3.1 Web.config 配置文件	20
1.1.4 .NET Framework 类库	2	3.3.2 嵌套配置设置	21
1.2 通用中间语言	3	3.3.3 在 Web.config 中存储自定义设置	24
1.3 .NET Framework 4.5 的新功能	3	3.3.4 ASP.NET Web 站点管理工具 WAT	24
1.4 小结	4	3.3.5 编程读取和写入配置设置	26
1.5 习题	4	3.4 Web 窗体基础	29
第 2 章 Visual Studio 2012 集成化开发环境	5	3.4.1 基于事件的编程模型	29
2.1 获取 Visual Studio 2012	5	3.4.2 自动回发特性 (AutoPostBack)	30
2.2 安装 Visual Studio 2012	5	3.4.3 Web 窗体处理流程	31
2.2.1 系统要求	6	3.4.4 ASP.NET 中的 Page 类	33
2.2.2 安装步骤	6	3.4.5 页面重定向	33
2.3 Visual Studio 2012 开发界面	7	3.4.6 HTML 编码	34
2.3.1 创建 web 应用程序	8	3.5 应用程序事件	35
2.3.2 菜单栏	9	3.5.1 Global.asax 全局文件	35
2.3.3 工具栏	10	3.5.2 基本应用程序事件	37
2.3.4 解决方案资源管理器	10	3.6 小结	38
2.3.5 控件工具箱	11	3.7 习题	38
2.3.6 服务器资源管理器	11	第 4 章 C#语言基础	39
2.3.7 错误列表	12	4.1 C#语言概述	39
2.3.8 页面设计窗口	12	4.2 C#语言的基本语法	39
2.3.9 代码编辑窗口	12	4.2.1 C#数据类型	39
2.4 Visual Studio 2012 的主要特性	13	4.2.2 变量和常量	42
2.5 小结	14	4.2.3 数据类型转换	44
2.6 习题	14	4.2.4 运算符与表达式	46
第 3 章 ASP.NET 技术简介	15	4.2.5 流程控制	48
3.1 ASP.NET 4.5 概述	15	4.2.6 数组	54
3.1.1 ASP.NET 基础概念	15	4.3 面向对象编程	57
3.1.2 ASP.NET 4.5 的新功能	16	4.3.1 类	57
3.1.3 ASP.NET 开发工具	16	4.3.2 属性、方法和事件	58
3.1.4 ASP.NET 客户端	17		

4.3.3	构造函数	60	第 7 章	ASP.NET 服务器控件	109
4.3.4	继承和多态	61	7.1	服务器控件概述	109
4.3.5	委托	70	7.1.1	HTML 服务器控件	109
4.3.6	事件	72	7.1.2	服务器控件的使用	109
4.3.7	C#语言其他概念和语言特色	76	7.1.3	Web 服务器控件	117
4.4	小结	80	7.2	服务器控件类	124
4.5	习题	80	7.2.1	服务器控件基本属性	124
第 5 章	ASP.NET 内置对象	82	7.2.2	服务器控件的事件	125
5.1	Page 类	82	7.3	文本服务器控件	127
5.1.1	页面的生命周期	82	7.3.1	标签 (Label) 控件	127
5.1.2	Page 类的属性、方法和事件	83	7.3.2	静态文本 (Literal) 控件	129
5.2	Response 对象	85	7.3.3	文本框 (TextBox) 控件	131
5.2.1	Response 对象的属性和方法	85	7.3.4	超链接文本 (HyperLink) 控件	132
5.2.2	应用 Response 对象	85	7.4	按钮服务器控件	133
5.3	Request 对象	86	7.4.1	普通按钮 (Button) 控件	133
5.3.1	Request 对象的属性和方法	86	7.4.2	超链接按钮 (LinkButton) 控件	134
5.3.2	应用 Request 对象	87	7.4.3	图像按钮 (ImageButton) 控件	135
5.4	Server 对象	87	7.5	图像服务器控件	135
5.4.1	Server 对象的属性和方法	88	7.5.1	图像 (Image) 控件	135
5.4.2	应用 Server 对象	88	7.5.2	图像地图 (ImageMap) 控件	137
5.5	Cache 对象	88	7.6	选择服务器控件	138
5.5.1	Cache 对象的属性和方法	89	7.6.1	复选框 (CheckBox) 控件	138
5.5.2	应用 Cache 对象	89	7.6.2	复选框列表 (CheckBoxList) 控件	140
5.6	状态管理	89	7.6.3	单选按钮 (RadioButton) 控件	141
5.6.1	ASP.NET 状态管理	89	7.6.4	单选按钮列表 (RadioButtonList) 控件	141
5.6.2	ViewState 对象	90	7.7	列表服务器控件	142
5.6.3	Cookie 对象	90	7.7.1	列表框 (ListBox) 控件	142
5.6.4	Session 对象	92	7.7.2	下拉列表框 (DropDownList) 控件	143
5.6.5	Application 对象	94	7.7.3	项目列表 (BulletedList) 控件	143
5.7	小结	95	7.8	容器服务器控件	144
5.8	习题	96	7.8.1	面板 (Panel) 控件	144
第 6 章	ASP.NET 页面语法	97	7.8.2	多视图 (Multiview) 控件	145
6.1	ASP.NET 网页扩展名	97	7.8.3	动态容器 (PlaceHolder) 控件	146
6.2	页面指令	98	7.9	高级服务器控件	147
6.3	ASPX 文件内容注释	105			
6.4	HTML 服务器控件语法	105			
6.5	ASP.NET 服务器控件语法	106			
6.6	代码块语法	106			
6.7	表达式语法	107			
6.8	小结	108			
6.9	习题	108			

7.9.1	日历 (Calendar) 控件	147	第 10 章	用户控件与 Web 部件	185
7.9.2	动态广告 (AdRotator) 控件	149	10.1	用户控件	185
7.10	小结	149	10.1.1	创建用户控件	185
7.11	习题	150	10.1.2	在 Web.config 中注册用户 控件	188
第 8 章	ASP.NET 中的样式、主题 和母版页	152	10.1.3	转换现有页为用户控件	190
8.1	在 ASP.NET 中应用 CSS 样式	152	10.2	编程处理用户控件	190
8.1.1	创建样式	152	10.2.1	公开用户控件中的属性	190
8.1.2	应用样式	154	10.2.2	使用自定义对象属性	192
8.2	主题	158	10.2.3	添加用户控件事件	194
8.2.1	创建主题	158	10.3	动态加载用户控件	196
8.2.2	创建外观	159	10.3.1	动态创建用户控件	196
8.2.3	应用主题和外观	160	10.3.2	使用 Reference 指令	198
8.3	母版页	160	10.4	Web 部件	198
8.3.1	创建母版页	160	10.4.1	使用 Web 部件	199
8.3.2	创建内容	162	10.4.2	WebPartManager 显示模式	206
8.3.3	母版页和相对路径	163	10.5	小结	206
8.3.4	在 web.config 中配置母版页	164	10.6	习题	206
8.3.5	修改母版页	164	第 11 章	ASP.NET 应用程序安全技术	207
8.3.6	动态加载母版页	165	11.1	身份验证	207
8.3.7	母版页的嵌套	168	11.1.1	基于 Windows 的身份验证	207
8.4	小结	170	11.1.2	基于 Forms 的身份验证	208
8.5	习题	170	11.2	安全代码的编写	209
第 9 章	ASP.NET 4.5 中的站点导航技术	172	11.2.1	防止 SQL 注入	209
9.1	ASP.NET 站点导航概述	172	11.2.2	合理使用错误页面	211
9.2	站点地图	172	11.3	使用 URL 授权	212
9.3	配置多个站点地图	173	11.4	小结	214
9.3.1	从父站点地图链接到子站点 地图文件	173	11.5	习题	214
9.3.2	在 Web.config 文件中配置多个 站点地图	174	第 12 章	ADO.NET 数据访问技术	215
9.4	SiteMapPath 控件	174	12.1	ADO.NET 概述	215
9.5	SiteMapDataSource 控件	175	12.1.1	ADO.NET 简介	215
9.6	Menu 控件	176	12.1.2	ADO.NET 对象模型	215
9.6.1	定义 Menu 菜单内容	176	12.1.3	数据访问模式	217
9.6.2	Menu 控件样式	177	12.2	数据库连接字符串	217
9.7	TreeView 控件	179	12.3	连接数据库	218
9.7.1	定义 TreeView 控件节点内容	179	12.3.1	Connection 对象概述	218
9.7.2	带复选框的 TreeView 控件	181	12.3.2	Connection 对象的属性及 方法	218
9.8	小结	184	12.3.3	使用 SqlConnection 对象连接 SQL Server 数据库实例	219
9.9	习题	184	12.4	获取数据	223

12.4.1	Command 对象概述	223	14.3.2	LINQ to SQL	264
12.4.2	Command 对象的属性及方法	223	14.4	LINQ to XML	267
12.4.3	使用 SqlCommand 对象执行 数据库命令	224	14.4.1	构造 XML 树	267
12.4.4	DataReader 对象概述	224	14.4.2	查询 XML 树	270
12.4.5	DataReader 对象的属性 及方法	224	14.4.3	操作 XML 树	272
12.4.6	使用 SqlDataReader 读取 数据库实例	225	14.5	LinqDataSource 控件	274
12.5	填充数据集	227	14.6	小结	278
12.5.1	DataAdapter 对象概述	227	14.7	习题	279
12.5.2	DataSet 对象概述	227	第 15 章	Web 服务和 WCF 服务	280
12.5.3	使用 DataAdapter 对象、 DataSet 对象综合实例	228	15.1	Web 服务	280
12.6	小结	229	15.1.1	Web 服务概述	280
12.7	习题	229	15.1.2	建立 ASP.NET Web 服务	280
第 13 章	ASP.NET 数据绑定技术与数据 绑定控件	230	15.1.3	调用 ASP.NET Web 服务	281
13.1	数据绑定概述	230	15.2	WCF 服务	284
13.1.1	简单数据绑定	230	15.2.1	建立 WCF 服务	284
13.1.2	复杂数据绑定	232	15.2.2	调用 WCF 服务	284
13.2	数据源控件	232	15.3	习题	287
13.2.1	SqlDataSource 数据源控件	232	第 16 章	ASP.NET 的 AJAX 开发技术	288
13.2.2	ObjectDataSource 数据源控件	237	16.1	ASP.NET AJAX 开发技术概述	288
13.2.3	LinqDataSource 数据源控件	240	16.1.1	AJAX 开发模式	289
13.3	数据绑定控件	242	16.1.2	AJAX 体系结构	294
13.3.1	GridView 控件	242	16.2	ASP.NET AJAX 核心控件	301
13.3.2	DetailsView 控件	249	16.2.1	ScriptManager 控件	302
13.3.3	DataList 控件	249	16.2.2	UpdatePanel 控件	303
13.3.4	ListView 控件和 DataPager 控件	251	16.2.3	UpdateProgress 控件	306
13.3.5	FormView 控件	255	16.2.4	Timer 控件	308
13.4	小结	255	16.3	AJAXControl Toolkit	310
13.5	习题	256	16.3.1	安装 ASP.NET AJAX Control Toolkit	310
第 14 章	LINQ 技术	258	16.3.2	AJAX Control Toolkit 控件 概览	313
14.1	LINQ 技术概述	258	16.4	小结	321
14.2	C#中的 LINQ	259	16.5	习题	321
14.2.1	LINQ 查询表达式	259	第 17 章	网站部署、打包与安装	323
14.2.2	LINQ 查询方法	260	17.1	Web 站点部署前的准备	323
14.3	LINQ to ADO.NET	263	17.2	IIS 的安装和配置	323
14.3.1	LINQ to DataSet	263	17.3	复制站点	325
			17.4	发布网站	326
			17.5	小结	328
			17.6	习题	328

第 1 章 .NET Framework 4.5 简介

本章要点或学习目标

- 理解 .NET Framework 的两个主要组件 CLR 和 FCL
- 理解通用中间语言 CIL
- 理解 .NET Framework 4.5 的新增功能

1.1 .NET Framework 基础概念

开发人员可以使用 .NET Framework 创建 Web 网站、Web 服务应用程序、Windows 系统以及智能设备应用程序等。微软公司从发布第一个 .NET Framework 以来，已经发布了 1.0 版、1.1 版、2.0 版、3.0 版、3.5 版、4.0 版，目前 .NET Framework 4.5 是最新的版本，也是功能最强大和最完善的一个版本。

1.1.1 什么是 .NET Framework

在传统的软件开发工作中，开发人员需要面对的是多种服务器和终端系统，包括用于个人计算机的 Windows 系统、用于服务器的 Windows 服务器系统、非 Windows 系统（如 FreeBSD、Linux 和 BSD）、用于平面设计的 Mac OS X 操纵系统，以及各种移动终端系统（如 Windows Mobile、IOS、Android）等。

在开发基于以上这些系统的软件时，开发者需要针对不同的硬件和操作系统，编写大量实现兼容性的代码，并使用不同的方式对代码进行编译。这一系列问题，都给软件设计和开发带来很多困难。

以 Windows 操作系统为例，目前主要使用的 Windows 操作系统内核包括 Windows 9X、NT4、NT5.0/5.1、NT6.0/6.1、Windows CE、Windows Mobile 6.X 和 Windows Phone OS 等，在这些操作系统下进行软件开发，可使用的技术包括以下几种：

- 用于图形图像开发的 GDI、DirectX、OpenGL 等技术。
- 用于数据库操作的 ADO、DAO、RDO、ODBC 等技术。
- 用于 Web 应用开发的 ASP、JSP、PHP 等技术。
- 用于移动终端开发的 XNA、HTML5 等技术。

以上这些技术都有各自的标准和接口，相互不兼容。有些软件开发人员必须学习和使用相同的技术才能实现协作；而企业在实施开发项目时，也需要聘用指定技术的开发人员，才能实现最终的产品。

基于以上问题，微软在 21 世纪初开发出了一种致力于敏捷而快速的软件开发框架，其更加注重平台无关化和网络透明化，以公用语言运行时（Common Language Runtime, CLR）为基础，支持多种编程语言，这就是微软的 .NET Framework。

1.1.2 .NET Framework 的目标

.NET Framework（框架）是支持生成和运行下一代应用程序和 Web 服务的内部 Windows 组件。事实上它的主要特色在于简化应用程序的开发复杂性，提供一个一致的开发模型，开发人员

可以选择任何支持 .NET 的编程语言来进行多种类型的应用程序开发, 如 Basic、C#、J#等。概括而言, .NET Framework 旨在实现下列目标:

- 提供一个一致的面向对象的编程环境。无论开发的程序是在本地存储并执行的 Windows 窗体程序, 还是基于 B/S 或者 C/S 架构的网络程序, 其编程界面风格及控件都是相似的。
- 提供一个将软件部署和版本控制冲突最小化的代码执行环境。
- 提供一个可提高代码(包括未知的或不完全受信任的第三方创建的代码)执行安全性的代码执行环境。
- 提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。
- 使开发人员的经验在面对类型大不相同的应用程序(如基于 Windows 的应用程序和基于 Web 的应用程序)时保持一致。
- 按照行业标准生成所有通信, 确保基于 .NET 框架的代码可与任何其他代码集成。

.NET Framework 由两个主要的组件组成: 分别为公共语言运行时 (Common Language Runtime, CLR) 和 .NET Framework 类库 (Framework Class Library, FCL)。

1.1.3 公共语言运行时

公共语言运行时 (CLR) 是 .NET 框架的基础。可以将运行时看作一个在执行时管理代码的代理, 它提供内存管理、线程管理和远程管理等核心任务, 并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准确性验证。以运行时为目标的代码称为托管代码, 而不以运行时为目标的代码称为非托管代码。 .NET 框架提供了托管执行环境, 简化了开发和部署并与各种编程语言的集成, 从而能够提高开发人员的工作效率。例如, 程序员在用自己选择的开发语言编写应用程序时, 可以利用其他开发人员用其他语言编写的运行时、类库和组件。

图 1-1 显示了公共语言运行时和类库、应用程序之间以及整个系统之间的关系。

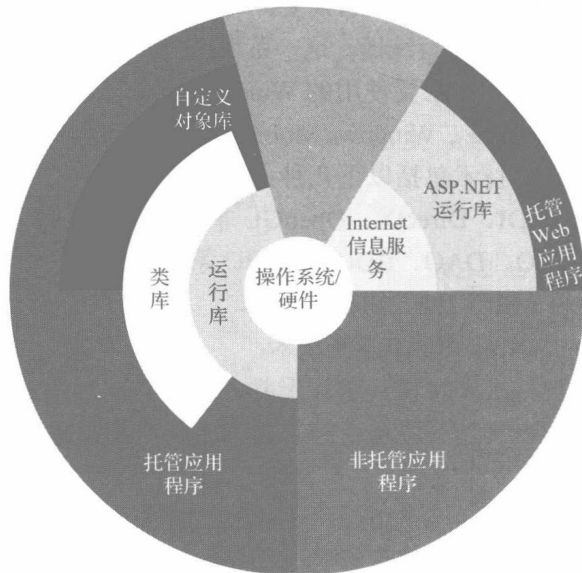


图 1-1 .NET 的总体结构

1.1.4 .NET Framework 类库

.NET Framework 类库 (FCL) 是一个与公共语言运行时紧密集成的可重用的类型集合, 包括类、接口和值类型的库, 提供对系统功能的访问, 且被设计为构建 .NET Framework 应用程序、组

件和控件的基础。这使得 .NET 框架类型不但易于使用，而且还减少了学习 .NET 框架新功能所需要的时间。此外，第三方组件可与 .NET 框架中的类无缝集成。

1.2 通用中间语言

通用中间语言（Common Intermediate Language, CIL）一般称为通用中间语言，类似于一个面向对象的汇编语言，独立于具体 CPU 和平台的指令集，它可以在任何支持 .NET 框架的环境下运行。

在 .NET 编程环境中，不管程序员使用 C++、C#、VB.NET 还是 J# 语言编写程序，在程序进行编译的时候，编译器都会将源代码编译为 CIL 语言，然后再通过实时（Just In Time, JIT）编译器编译为针对各种不同 CPU 的指令（注意，因为 JIT 是实时编译器，所以它只编译需要运行的 CIL 语言段，而不是全部一下编译完，这样可以提高程序编译效率）。

因为所有的 .NET 编程语言都基于 .NET 框架并生成 CIL，所以这些语言的编程风格非常相似，因此，学会一种 .NET 编程语言，其他 .NET 编程语言很快就能掌握。在本书中，采用的编程语言是 C#。 .NET 体系结构如图 1-2 所示。

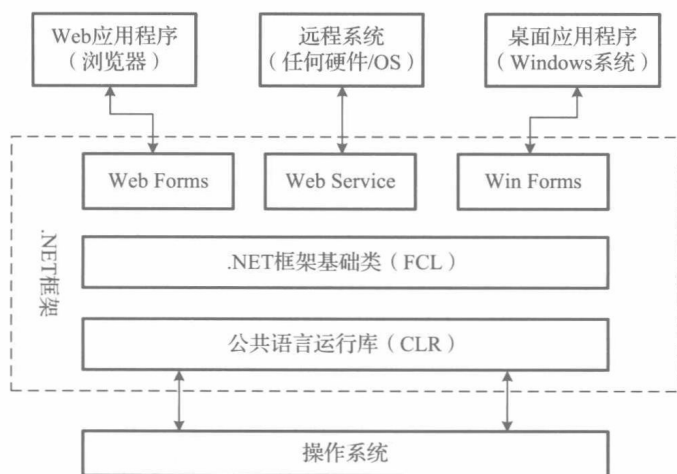


图 1-2 .NET 体系结构

1.3 .NET Framework 4.5 的新功能

.NET Framework 4.5 是在以前版本 .NET Framework 4.0 的基础上完善而成的，为方便团队开发，增强应用程序的安全性，适应网络技术的新发展，微软对 .NET 框架原有的功能进行了完善和改进，并增加了很多新功能：

- 新增了对 Windows 商店应用程序的支持。
- 新增了可移植类库功能。
- 可生成在多个 .NET 框架平台（如 Windows Phone 和 Windows 应用商店应用程序的 .NET）上处理的托管程序集。
- 使用异步操作提高文件输入/输出性能。
- 提高多核处理器的启动性能。

- 对多个客户端启用异步流消息, 提高 WCF 应用程序的可伸缩性。
- 新增资源文件生成器 (resgen.exe)。
- 对 HTML5 的全面支持。
- 增强了托管扩展框架 (Managed Extensibility Framework, MEF) 功能。
- 提供用于 HTTP 应用程序的新编程接口。
- 增强的 Windows Presentation Foundation (WPF) 功能, 向 WPF 应用程序添加功能区用户界面。
- 更新的工作流 (Windows Workflow Foundation) 技术。

1.4 小结

本章详细介绍了 .NET Framework 的概念、目标以及框架包含的两个重要组件 CLR 和 FCL; 然后介绍了通用中间语言 CIL; 最后阐述了 .NET Framework 4.5 相比以前版本新增的功能。

1.5 习题

1. 简述 .NET Framework 的基本结构。
2. 简述 .NET Framework 4.5 的新增功能。

第 2 章 Visual Studio 2012 集成化开发环境

本章要点或学习目标

- 认识 Visual Studio 2012 IDE
- 了解 Visual Studio 2012 的一些开发特性
- 能够安装 Visual Studio 2012 应用程序

2.1 获取 Visual Studio 2012

“工欲善其事，必先利其器”。好的开发工具可以让程序设计人员事半功倍，Visual Studio 2012 开发工具是目前所有开发工具中的佼佼者，该工具是微软公司未来配合 .NET 战略推出的 IDE 开发环境，本身包含 .NET Framework 及 ASP.NET 程序开发服务器。另外，在编写程序时该工具的智能代码提示功能会主动提示目前可用的属性、方法及参数等，程序开发人员可很方便地从中选择需要的代码，如图 2-1 所示为 Visual Studio 2012 智能显示可用的方法、属性及参数。

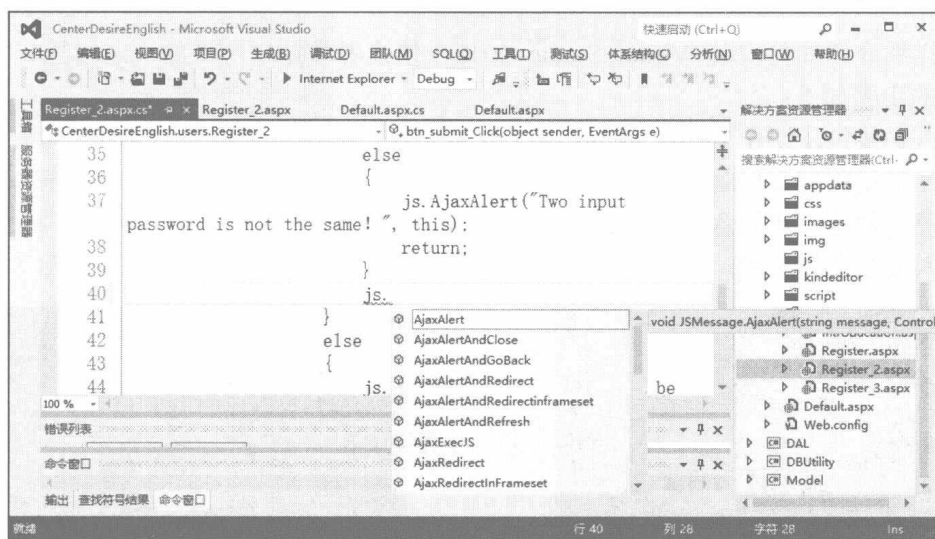


图 2-1 Visual Studio 2012 智能显示可用的方法、属性及参数

那么，我们如何获取 Visual Studio 2012 呢？方法就是通过搜索引擎搜索“VS2012 安装包下载”，然后找到适合的链接下载 Visual Studio 2012 安装包，下载完毕即可。本章提供一个 Visual Studio 2012 的安装包下载地址（Visual Studio 2012 Express For Win 8 简体中文版）：http://download.microsoft.com/download/8/1/b/81b0c41a-595f-4d5d-8c83-bb29addb265d/vs2012_winexp_chs.iso，读者可以打开该链接地址进行下载、安装。

2.2 安装 Visual Studio 2012

对于专业程序开发人员来说，一般会通过安装 Visual Studio 2012 集成开发环境，来开发 ASP.NET 应用程序。本节将对 Visual Studio 2012 的安装与配置进行介绍。

2.2.1 系统要求

1. 操作系统要求

- Windows XP Service Pack 2.0
- Windows Server 2003 Service Pack 1.0
- Windows 7
- Windows 8

2. 硬件要求

- CPU: 至少 600MHz (推荐使用 1GHz)
- 内存: 至少 1GB (推荐使用 4GB)
- 显示器: 至少 800×600 像素, 256 色 (建议用 1024×768 像素, 增强色为 16 位)

3. 磁盘要求

- 全部安装 (包括帮助文档, 即 MSDN), 安装盘上至少需要 2.86GB 磁盘空间, 系统盘至少剩余 6GB 磁盘空间

2.2.2 安装步骤

下面将详细介绍如何安装 Visual Studio 2012, 使读者掌握每一步的安装过程。安装 Visual Studio 2012 的步骤如下:

(1) 解压安装包, 打开 Visual Studio 2012 安装文件, 进入安装程序文件界面, 双击 setup.exe 可执行文件, 应用程序会自动跳转到如图 2-2 所示的“Visual Studio 2012 安装程序”界面。可以选择安装路径, 勾选“我同意许可条款和条件”, 第二个选项可以不勾选, 单击“下一步”按钮。

(2) 进入安装选择功能, 根据自己的需求, 单击“安装”按钮。如图 2-3 所示。

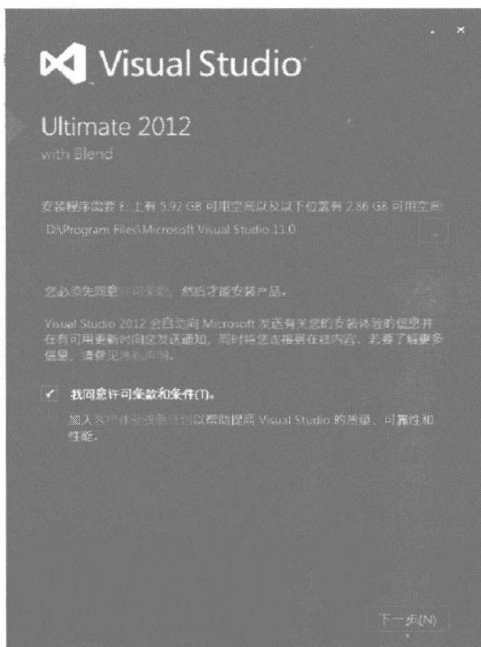


图 2-2 Visual Studio 2012 安装程序起始页

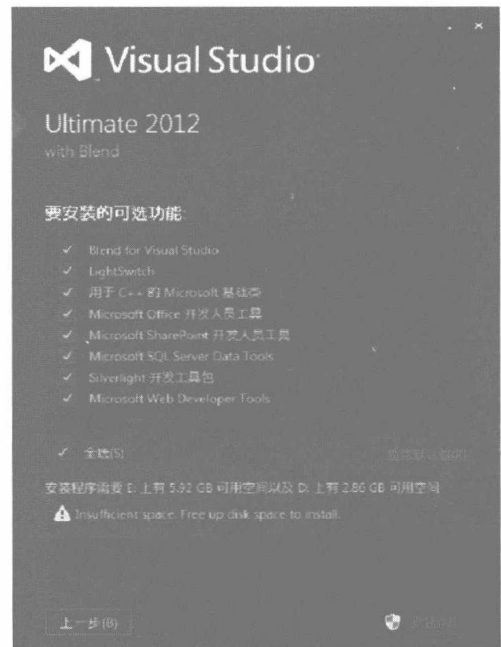


图 2-3 Visual Studio 2012 安装选择界面

(3) 进入安装界面, 如图 2-4 所示, 安装成功后就可以启动了。

(4) 首次使用 Visual Studio 2012 之前, 需要指定您从事最多的开发活动类型, 如 Visual Basic 或 Visual C#, 此信息用于将预定义的设置集合应用于针对您的开发活动而设计的开发环境, 如图 2-5 所示。另外, Visual Studio 2012 需要一个密钥, 这里提供一个供大家使用: YKCW6-BPFPF-BT8C9-7DCTH-QXGWC, 如图 2-6 所示。

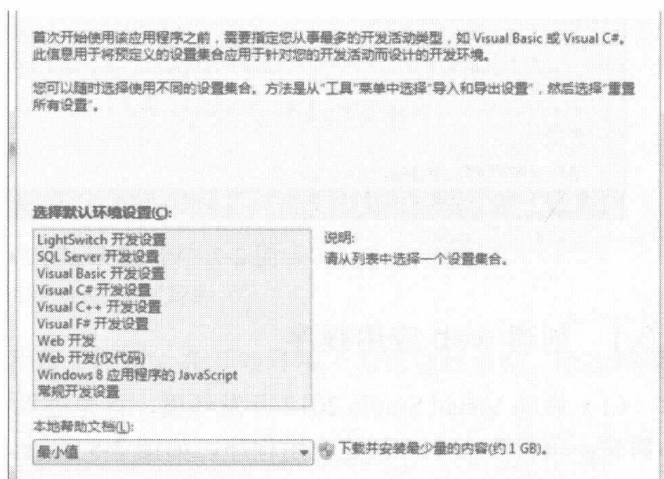
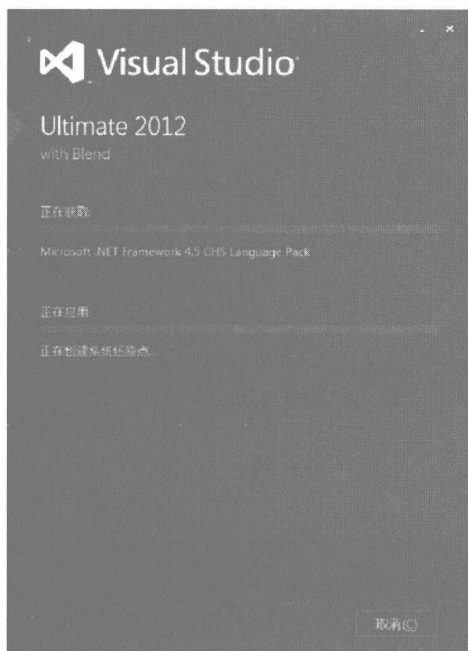


图 2-4 Visual Studio 2012 安装进度界面

图 2-5 Visual Studio 2012 选择默认环境设置



图 2-6 Visual Studio 2012 密钥填写

2.3 Visual Studio 2012 开发界面

当打开 Visual Studio 2012 时, 将显示如图 2-7 所示的界面, 该窗口与普通的 Windows 窗体区别不大, 同样具有菜单栏、工具栏, 然后是一些自动停靠的窗口。下面我们通过创建一个 Web 应用程序来介绍 Visual Studio 2012 的一些开发特性。



图 2-7 Visual Studio 2012 开发界面

2.3.1 创建 web 应用程序

(1) 启动 Visual Studio 2012 开发环境，首先进入“起始页”界面。在该界面中，单击“文件→新建→项目”命令，创建应用程序，如图 2-8 所示。

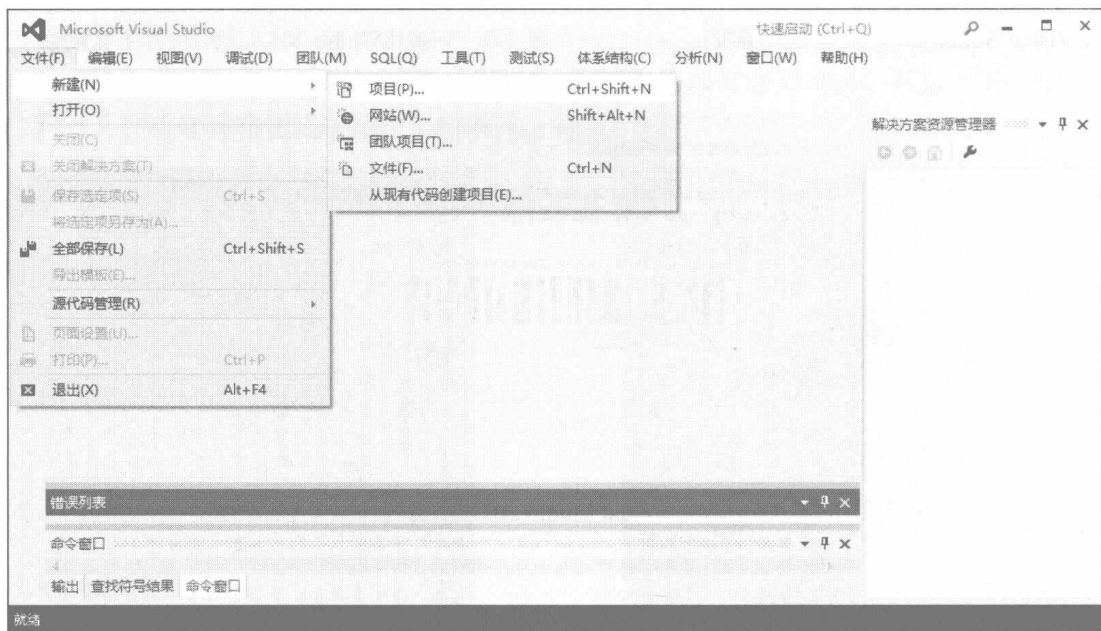


图 2-8 新建项目

(2) 选择“新建项目”后，将打开新 Web 应用程序窗口。在该窗口的“模板”区域内选择“Web”，再选择“ASP.NET 空 Web 应用程序”，然后确定 Web 应用程序的位置，并选择编程语言，如图 2-9 所示：