

21 世纪计算机科学与技术实践型教程

崔艳春 主编
高云 夏平 副主编

C#程序设计项目化教程

清华大学出版社



崔艳春 主编
高云 夏平 副主编

C#程序设计项目化教程

清华大学出版社
北京

内 容 简 介

本书针对软件技术专业职业岗位的从业需求,重构教学内容,以“学生成绩管理系统”真实项目开发过程为主线,引导学生掌握使用 C# 语言开发 Windows 应用程序的方法和技能,达到学以致用目标。本书分为 C# 基础知识、学生成绩管理系统开发和 C# 其他技术扩展三个部分,以完成“学生成绩管理系统”为工作任务,每章实现系统的一个功能,先呈现工作任务的完成效果,再进行知识准备,最后给出具体实现步骤,完成工作任务。

本书切合实际,结构合理,内容丰富,操作方便,各章配有精心设计的习题,并为任课教师提供免费的电子课件和源代码。

本书可以作为应用型本科和高等职业教育计算机及相关专业的教材,也可作为软件开发人员参考用书,还可用于读者自学。

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

C# 程序设计项目化教程/崔艳春主编. —北京:清华大学出版社,2016

21 世纪计算机科学与技术实践型教程

ISBN 978-7-302-42822-0

I. ①C… II. ①崔… III. ①C 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(2016)第 028517 号

责任编辑:谢 琛

封面设计:傅瑞学

责任校对:梁 毅

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

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:14.5

字 数:359 千字

版 次:2016 年 6 月第 1 版

印 次:2016 年 6 月第 1 次印刷

印 数:1~2000

定 价:29.50 元

产品编号:068622-01

《21 世纪计算机科学与技术实践型教程》

编辑委员会

主 任：陈 明

委 员：毛国君 白中英 叶新铭 刘淑芬 刘书家
汤 庸 何炎祥 陈永义 罗四维 段友祥
高维东 郭 禾 姚 琳 崔武子 曹元大
谢树煜 焦金生 韩江洪

策划编辑：谢 琛

《21 世纪计算机科学与技术实践型教程》

序

21 世纪影响世界的三大关键技术：以计算机和网络为代表的信息技术；以基因工程为代表的生命科学和生物技术；以纳米技术为代表的新型材料技术。信息技术居三大关键技术之首。国民经济的发展采取信息化带动现代化的方针，要求在所有领域中迅速推广信息技术，导致需要大量的计算机科学与技术领域的优秀人才。

计算机科学与技术的广泛应用是计算机学科发展的原动力，计算机科学是一门应用科学。因此，计算机学科的优秀人才不仅应具有坚实的科学理论基础，而且更重要的是能将理论与实践相结合，并具有解决实际问题的能力。培养计算机科学与技术的优秀人才是社会的需要、国民经济发展的需要。

制订科学的教学计划对于培养计算机科学与技术人才十分重要，而教材的选择是实施教学计划的一个重要组成部分，《21 世纪计算机科学与技术实践型教程》主要考虑了下述两方面。

一方面，高等学校的计算机科学与技术专业的学生，在学习了基本的必修课和部分选修课程之后，立刻进行计算机应用系统的软件和硬件开发与应用尚存在一些困难，而《21 世纪计算机科学与技术实践型教程》就是为了填补这部分空白。将理论与实际联系起来，使学生不仅学会了计算机科学理论，而且也学会了应用这些理论解决实际问题。

另一方面，计算机科学与技术专业的课程内容需要经过实践练习，才能深刻理解和掌握。因此，本套教材增强了实践性、应用性和可理解性，并在体例上做了改进——使用案例说明。

实践型教学占有重要的位置，不仅体现了理论和实践紧密结合的学科特征，而且对于提高学生的综合素质，培养学生的创新精神与实践能力有特殊的作用。因此，研究和撰写实践型教材是必需的，也是十分重要的任务。优秀的教材是保证高水平教学的重要因素，选择水平高、内容新、实践性强的教材可以促进课堂教学质量的快速提升。在教学中，应用实践型教材可以增强学生的认知能力、创新能力、实践能力以及团队协作和交流表达能力。

实践型教材应由教学经验丰富、实际应用经验丰富的教师撰写。此系列教材的作者不但从事多年的计算机教学，而且参加并完成了多项计算机类的科研项目，他们把积累的经验、知识、智慧、素质融于教材中，奉献给计算机科学与技术的教学。

我们在组织本系列教材过程中，虽然经过了详细的思考和讨论，但毕竟是初步的尝试，不完善甚至缺陷不可避免，敬请读者指正。

本系列教材主编 陈明

2005 年 1 月于北京

前 言

微软公司的 Microsoft .NET 改变了开发人员开发应用程序的方式及思维方式,有利于创建各种全新的应用程序。C# 是微软公司基于 .NET 平台推出的新一代编程语言,其功能强大、简洁明快,使程序设计工作变得轻松快捷,因而成为众多程序员的首选编程语言,在各个领域都得到了广泛的应用。

本书特色

本书不是 C# 基本语法教程,而是一本软件开发教程,以“项目主导,任务驱动”的模式讲解真实项目“学生成绩管理系统”的开发过程。每章按照“布置任务→知识准备→实现功能”的体系结构完成,让读者掌握使用 C# 的一些关键技术,掌握应用软件开发中的常用技术与方法。

本书主要内容

第 1 章 Visual C# 简介。通过该部分的学习,可了解 C# 发展历程和 .NET Framework、C# 语言,会安装和卸载 Visual Studio .NET 2012,熟悉 Visual Studio .NET 2012 开发环境,能创建第一个 Windows 应用程序,为后面学习做好准备。

第 2 章基础知识积累。通过该部分的学习,读者应掌握 C# 基本语法,包括数据类型、变量、常量、表达式、运算符、程序流程控制、面向对象等基本概念。

第 3 章学生成绩管理系统介绍。通过该部分的学习,读者应了解软件开发的过程(需求分析、总体设计、数据库设计、模块设计、调试运行等),熟悉学生成绩管理系统数据库结构以及需要实现的主要功能模块。

第 4 章启动窗体设计。通过该部分的学习,读者应掌握 Form 窗体、Label 控件、LinkLabel 控件、TextBox 控件的使用方法,理解多窗体项目的创建方法,理解线程的基本操作,根据具体步骤完成启动窗体设计。

第 5 章学生登录功能设计。通过该部分的学习,读者应掌握 ListBox 控件、ComboBox 控件、消息框的使用方法,理解 ADO .NET 访问数据库理念,掌握 ADO .NET 访问数据的基本方法,根据具体步骤完成学生登录功能。

第 6 章学生主窗体设计。通过该部分的学习,读者应掌握菜单栏、工具栏、任务栏、Timer 控件的使用方法,理解属性概念,会自定义属性,根据具体步骤完成学生主窗体设计。

第7章修改学生密码功能设计。通过该部分的学习,读者应理解并掌握数组的定义方法和引用方法,掌握字符与字符串的使用方法,理解异常概念,会根据实际情况处理程序出现的异常,根据具体步骤完成修改学生密码功能。

第8章修改学生信息功能设计。通过该部分的学习,读者应掌握 RadioButton 控件、CheckBox 控件的使用方法,理解 DataSet 数据集概念,掌握使用数据集断开式访问数据库的方法,学会根据具体步骤修改学生信息功能。

第9章教师查询教授课程功能设计。通过该部分的学习,读者应掌握 DataGridView 控件的使用方法,理解调用存储过程访问数据库理念,掌握存储过程访问数据库方法,学会根据具体步骤完成教师查询教授课程功能。

第10章教师录入成绩功能设计。通过该部分的学习,读者应掌握 DataGridView 控件的高级应用,学会根据具体步骤完成教师录入成绩功能。

第11章学生查询成绩功能设计。通过该部分的学习,读者应学会根据具体步骤完成学生查询成绩功能。

第12章 Windows 应用程序的部署。通过该部分的学习,读者应理解部署情况、掌握部署策略,学会根据具体步骤完成学生成绩管理系统的部署工作。

第13章 Web 应用程序基础。通过该部分的学习,读者应了解 Web 应用程序特点,掌握 IIS 的用法及发布网站的方法,学会创建简单的 Web 应用程序并发布。

第14章其他技术。通过该部分的学习,读者应理解 GDI+绘图类、文件与流的概念,会使用 GDI+绘图技术绘制图像,会使用流完成程序与文件、内存之间的数据传输。

读者对象

本书可以作为应用型本科和高等职业教育计算机及相关专业的教材,也可作为软件开发人员参考用书,还可用于读者自学。

本书由崔艳春任主编,高云、夏平任副主编,其中第1、2章由夏平编写,第3~11章由崔艳春编写,第12~14章由高云编写,由崔艳春负责统稿。

限于作者水平,书中难免存在不当之处,恳请广大读者批评指正。

作 者

2015年11月

目 录

第 1 章 Visual C# 简介	1
1.1 Visual C# 概述	1
1.1.1 C# 语言发展历程	1
1.1.2 C# 语言特性	2
1.2 .NET Framework	3
1.3 安装与卸载 Visual Studio. NET 2012	4
1.3.1 系统必备	4
1.3.2 安装 Visual Studio. NET 2012	5
1.3.3 卸载 Visual Studio. NET 2012	9
1.4 Visual Studio. NET 开发环境介绍	10
1.4.1 Visual Studio 界面	10
1.4.2 菜单栏	10
1.4.3 工具栏	11
1.4.4 工具箱	12
1.4.5 代码编辑器	13
1.4.6 解决方案资源管理器	13
1.4.7 属性窗口	14
1.5 第一个 Windows 应用程序	15
1.5.1 编写 C# 程序的三个步骤	15
1.5.2 C# 应用程序文件	19
本章小结	20
习题	20
第 2 章 基础知识积累	22
2.1 数据类型	22
2.1.1 值类型	23
2.1.2 引用类型	26
2.1.3 类型转换	26

2.2 变量	28
2.2.1 变量的命名	28
2.2.2 变量的声明	28
2.3 常量	29
2.3.1 常量的声明	29
2.3.2 常量的使用	30
2.4 运算符与表达式	30
2.4.1 算术运算符	30
2.4.2 赋值运算符	31
2.4.3 关系运算符	32
2.4.4 逻辑运算符	32
2.4.5 其他特殊运算符	33
2.4.6 运算符优先级	33
2.5 程序流程控制	34
2.5.1 顺序结构	34
2.5.2 选择结构	35
2.5.3 循环结构	38
2.5.4 跳转语句	39
2.6 面向对象基本概念	40
2.6.1 类	40
2.6.2 对象	42
2.6.3 方法	42
2.6.4 属性	44
2.6.5 事件	45
本章小结	46
习题	46
第3章 学生成绩管理系统介绍	48
3.1 需求分析	48
3.2 总体设计	48
3.2.1 系统目标	48
3.2.2 构建开发环境	48
3.2.3 软件功能结构	49
3.2.4 业务流程图	50
3.3 数据库设计	51
3.3.1 数据库概要说明	51
3.3.2 实体 E-R 图	51
3.3.3 数据表设计	53

3.3.4 存储过程设计	56
3.4 系统功能介绍	57
3.4.1 系统启动窗体	57
3.4.2 学生(教师)登录窗体	57
3.4.3 主窗体	57
3.4.4 修改密码窗体	57
3.4.5 修改个人信息窗体	57
3.4.6 教师录入成绩窗体	57
3.4.7 学生查询成绩窗体	58
本章小结	58
第4章 学生成绩管理系统——启动窗体设计	59
4.1 本章任务	59
4.2 准备工作	60
4.2.1 Windows 窗体	60
4.2.2 Label 控件	68
4.2.3 LinkLabel 控件	68
4.2.4 TextBox 控件	69
4.2.5 多窗体项目	71
4.2.6 线程介绍	74
4.3 完成启动窗体	75
4.3.1 设计界面	75
4.3.2 设置属性	75
4.3.3 编写代码	77
本章小结	78
习题	79
第5章 学生成绩管理系统——学生登录功能设计	81
5.1 本章任务	81
5.2 准备工作	82
5.2.1 列表框和组合框	83
5.2.2 消息框的使用	86
5.2.3 ADO.NET 访问数据库文件	88
5.3 完成学生登录功能	90
5.3.1 设计界面	91
5.3.2 设置属性	91
5.3.3 编写代码	92
本章小结	96

习题	96
第6章 学生成绩管理系统——学生主窗体设计	98
6.1 本章任务	98
6.2 准备工作	99
6.2.1 菜单栏	99
6.2.2 工具栏	102
6.2.3 状态栏	104
6.2.4 Timer 控件	104
6.2.5 自定义属性	105
6.3 完成学生主窗体	106
6.3.1 设计界面	107
6.3.2 设置属性	107
6.3.3 编写代码	108
本章小结	109
习题	110
第7章 学生成绩管理系统——修改学生密码功能设计	112
7.1 本章任务	112
7.2 准备工作	113
7.2.1 数组	113
7.2.2 字符与字符串	114
7.2.3 程序调试与异常处理	116
7.3 完成修改密码功能	117
7.3.1 设计界面	118
7.3.2 设置属性	118
7.3.3 编写代码	119
本章小结	121
习题	121
第8章 学生成绩管理系统——修改学生信息功能设计	123
8.1 本章任务	123
8.2 准备工作	124
8.2.1 单选钮	124
8.2.2 复选框	124
8.2.3 DataSet	126
8.3 完成修改学生信息功能	127
8.3.1 设计界面	128

8.3.2 设置属性.....	128
8.3.3 编写代码.....	129
本章小结.....	132
习题.....	132
第9章 学生成绩管理系统——教师查询教授课程功能设计	134
9.1 本章任务	134
9.2 准备工作	135
9.2.1 DataGridView 控件	135
9.2.2 调用存储过程.....	140
9.3 完成教师查询教授课程功能	141
9.3.1 设计界面.....	141
9.3.2 设置属性.....	141
9.3.3 编写代码.....	142
本章小结.....	145
习题.....	145
第10章 学生成绩管理系统——教师录入成绩功能设计	146
10.1 本章任务.....	146
10.2 准备工作.....	147
10.3 实现教师录入成绩功能.....	151
10.3.1 设计界面.....	151
10.3.2 设置属性.....	151
10.3.3 编写代码.....	152
本章小结.....	155
习题.....	155
第11章 学生成绩管理系统——学生查询成绩功能设计	156
11.1 本章任务.....	156
11.2 准备工作.....	157
11.3 实现学生查询成绩功能.....	158
11.3.1 设计界面.....	158
11.3.2 设置属性.....	158
11.3.3 编写代码.....	159
本章小结.....	161
习题.....	161

第 12 章 Windows 应用程序的部署	162
12.1 部署概述	162
12.2 部署策略	162
12.2.1 ClickOnce	162
12.2.2 Windows Installer	163
12.3 部署学生成绩管理系统	163
12.3.1 生成 Windows 应用程序安装包	163
12.3.2 Windows 应用程序安装	167
本章小结	168
习题	168
第 13 章 Web 应用程序基础	170
13.1 ASP.NET 的特点	170
13.2 IIS	171
13.2.1 安装 IIS	171
13.2.2 设置虚拟目录	174
13.3 ASP.NET 控件简介	176
13.3.1 HTML 服务器控件	176
13.3.2 Web 服务器控件	178
13.3.3 输入验证控件	178
13.4 创建第一个 Web 应用程序	179
13.5 发布 Web 应用程序	183
本章小结	188
习题	188
第 14 章 其他技术	190
14.1 GDI+绘图	190
14.1.1 GDI+简介	190
14.1.2 Graphics 类	191
14.1.3 Pen 类	192
14.1.4 Brush 类	192
14.1.5 Font 类	193
14.1.6 GDI+绘图示例	193
14.2 文件与流	196
14.2.1 流	196
14.2.2 System.IO 命名空间	196
14.2.3 File 类	197

14.2.4 FileInfo 类	198
14.2.5 Directory 类	199
14.2.6 DirectoryInfo 类	201
14.2.7 FileStream 类	202
14.2.8 StreamWriter 类	203
14.2.9 StreamReader 类	203
本章小结	204
习题	204
附录 习题参考答案	206
参考文献	215

第 1 章 Visual C# 简介

本章要点

- Visual C# 语言介绍。
- .NET Framework 的基本概念。
- Visual Studio .NET 开发环境介绍。

学习目标

- 了解 C# 语言发展历程和特点。
- 掌握 .NET Framework 的基本概念。
- 掌握创建一个 Visual C# Windows 应用程序的步骤。

1.1 Visual C# 概述

1.1.1 C# 语言发展历程

C# (读作 C sharp) 发布于 2000 年 6 月, 是微软公司主导的一种编程语言, 是面向对象的、运行于 .NET Framework 之上的高级程序设计语言。

C# 是由 C 和 C++ 衍生出来的, 与 Java 程序设计语言看起来非常相似, 但也有着显著的不同, C# 与 COM (组件对象模型) 是直接集成的, 它是微软公司在 .NET 框架的中主推的一种编程语言。

C# 可以说融合了微软公司曾经主导过的很多编程语言的特点, 它继承了 C 和 C++ 强大功能的同时去掉了一些复杂特性 (没有宏, 也不允许多重继承), 具备了 VB 简单的可视化操作和 C++ 的高运行效率, 基于以上特点, 习惯使用 C、C++、VB 甚至 Java 的程序员都可以很容易地转向 C# 开发, 并能在其中找寻到自己喜欢的特性。

自 C# 语言发布以来, 微软公司不断地更新 C# 语言的版本, 每次升级都能带来让人们眼前一亮的新特性, 简而言之, C# 语言变得越来越出色, 也更加的好用了。表 1-1 列出了 C# 每个版本和对应的 .NET Framework 版本。

表 1-1 C# 版本发展历程

时 间	C# 版本	平台版本	集成开发工具
2003.4	C# 语言规范 1.2	.NET Framework 1.1	Visual Studio .NET 2003
2005.11	C# 语言规范 2.0	.NET Framework 2.0	Visual Studio .NET 2005

续表

时 间	C# 版本	平台版本	集成开发工具
2006.11	C# 语言规范 2.0	.NET Framework 3.0	Visual Studio .NET 2005
2007.8	C# 语言规范 3.0	.NET Framework 3.0	Visual Studio .NET 2005
2007.11	C# 语言规范 3.0	.NET Framework 3.5	Visual Studio .NET 2008
2010.4	C# 语言规范 4.0	.NET Framework 4.0	Visual Studio .NET 2010
2012.8	C# 语言规范 5.0	.NET Framework 4.5	Visual Studio .NET 2012

1.1.2 C# 语言特性

C# 是一种高级编程语言,专门用于微软的 .NET Framework 平台。C# 具有简单、功能强大、类型安全,面向对象等特性。在微软的 Visual Studio 集成开发环境中,C# 与 Visual Basic、C++ 等开发语言被集成到一起,并以统一的用户界面和安全机制为开发人员提供服务。C# 允许开发人员开发面向 Windows、Web 和移动设备的程序。

1. 简单

C# 继承了 C 和 C++ 的优点,并在此基础上进行了改善,使语言更加简单。C# 同时也改进了其他编程语言(例如 C++ 和 Java)中的一些复杂性和缺陷,例如,C# 中摒弃了 C++ 中功能强大但危机四伏的指针,这使得程序员能有效地减少开发过程中的致命错误。

2. 面向对象

C# 具有面向对象程序设计语言所有的一切特性:继承、封装和多态,C# 面向对象开发有 .NET 底层类库的支持,可以很轻松地创建各种类型的对象。

3. 类型安全

类型安全是编写代码优先考虑的问题,以前的 C/C++ 程序员在编程时必须小心谨慎对待这些问题,编程时压力很大。而 C# 默认能够提供类型安全机制,可以避免一些常见的类型问题,这样程序员可以把注意力集中到一些更重要的地方,比如说商业逻辑。

4. 与 Web 紧密结合

XML 是一种最流行的网络中结构化数据传送的标准,C# 对 XML 提供了很好的支持,可以轻松地创建 XML,也可以将 XML 数据应用到程序中,程序员能够利用简单的 C# 语言方便地开发 XML Web Service,有效地处理网络中的各种数据。C# 通过内置的服务,使组件可以转化为 XML 网络服务,这样就可以被其他应用程序调用,实现一次创造、重复利用的高效开发模式。由于 XML 技术真正融入到 .NET 之中,C# 的编程变成了真正意义的网络编程。

5. 基于 .NET Framework

.NET Framework 是一个强大的体系,它为用 C# 编写的应用程序提供了安全性保障和错误处理机制,另外,在 .NET 框架中,C# 可以自由地和其他语言(VB、J# 等)进行

转换。

6. 支持跨平台

随着 Internet 应用程序的普及,开发人员所设计的应用程序必须具有强大的跨平台性。C# 编写的应用程序具有这种跨平台特性,它包括了 C# 程序可以运行在不同类型的客户端上,比如 PDA、手机等非 PC 设备。

1.2 .NET Framework

Microsoft .NET Framework 是用于 Windows 平台的托管代码编程模型,主要由 .NET Framework 和 Visual Studio .NET 开发工具组成。

.NET Framework 是微软主推的应用程序开发框架,该框架提供跨平台和跨语言的特性,C# 是其主要的开发语言。它包含了操作系统上软件开发的所有层,简化了在高度分布式网络环境中的应用开发。使用 .NET Framework,配合微软公司推出的 Visual Studio 集成开发环境,开发人员可以比以往更轻松地创建出功能强大的应用程序。

.NET Framework 主要包括两个最基本的内核,即由公共语言运行时(Common Language Runtime,CLR)和 .NET Framework 基本类库,它们为 .NET 平台的实现提供了底层技术支持。

1. 公共语言运行时

公共语言运行时(见图 1-1)是所有 .NET 应用程序运行时环境,是所有 .NET 应用程序都要使用的编程基础。所有 .NET 语言(比如 C#、VB 等)公用的执行期组件、公共语言类型库提供了很多用来简化代码开发和应用程序部署的服务,同时在可靠性和安全性方面也提供了大量的服务。它为在其上的应用层提供统一的底层进程和线程管理、内存管理、安全管理、代码验证和编译以及其他的系统服务。



图 1-1 .NET Framework 体系结构图

2. .NET Framework 类库

.NET Framework 类库,也叫框架类库(Framework Class Library,FCL),提供在 Windows 平台上开发 Windows 应用程序所需的几乎所有常见的功能的类库,包括重新以组件的方式写成的 ASP.NET。这些基础类库可以为任何一种基于 .NET 的编程语言