高等职业教育“十二五”规划教材
计算机类核心课程教改项目成果系列教材

C#程序设计案例教程

（第二版）

王明福　主编

科学出版社

免费提供教学资源

高等职业教育“十二五”规划教材
计算机类核心课程教改项目成果系列教材

C#程序设计案例教程

（第二版）

王明福　主编

科学出版社
北京

内 容 简 介

本书以微软 Visual Studio 2010 作为开发平台，全书共分 12 章。其中，前 8 章介绍 C# 语言面向对象程序设计基础和编程环境，主要内容包括 Visual Studio 2010 开发平台介绍、C#语言基础、流程控制、数组与结构，以及类与对象、继承与多态、接口、委托、事件、泛型和文件等；第 9～12 章介绍 Visual C#窗体应用开发，通过开发计算器、记事本和学生信息管理系统等 Windows 应用程序，详细介绍了包括窗体与常用控件、菜单与工具栏、对话框等在内的界面设计，以及 Windows 窗体应用程序进阶，包括文件操作和数据库技术等编程技术。

书中所有程序全部运行通过，并免费提供所有源程序代码，以及模仿练习、拓展训练和自我测试练习等源程序代码。

本书可以作为高职高专院校计算机信息管理、软件技术等相关专业“C#面向对象程序设计”课程的教材，也可以作为应用型本科相关专业“C#面向对象程序设计”课程的教材。

图书在版编目（CIP）数据

C#程序设计案例教程/王明福主编. —2 版. —北京：科学出版社，2016

（高等职业教育“十二五”规划教材·计算机类核心课程教改项目成果系列教材）

ISBN 978-7-03-047171-0

Ⅰ. ①C… Ⅱ. ①王… Ⅲ. ①C 语言-程序设计-高等职业教育-教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2016）第 019422 号

责任编辑：孙露露/责任校对：王万红

责任印制：吕春珉/封面设计：耕者设计工作室

科学出版社 出版

北京东黄城根北街 16 号

邮政编码：100717

http://www.sciencep.com

三河市骏杰印刷有限公司印刷

科学出版社发行 各地新华书店经销

*

2009 年 3 月第 一 版 开本：787×1092 1/16

2016 年 3 月第 二 版 印张：18

2016 年 3 月第一次印刷 字数：437 000

定价：39.00 元

（如有印装质量问题，我社负责调换〈骏杰〉）

销售部电话 010-62136230 编辑部电话 010-62138978-2010

前 言

面向对象设计技术已成为当今流行的软件设计技术。C#是在面向对象的大潮流中诞生的宠儿，同时由于它的广泛运用又极大地推动了面向对象技术的发展。

本书以面向对象的基本思想、方法为主要内容，以微软 Visual Studio 2010 作为开发平台，兼顾面向过程与面向对象程序设计的适度分离和高度融合的原则。本书前 4 章是面向过程的C# 语言基础和语法规则的学习；第 5～8 章是面向对象基本特征和基本技术的学习，主线突出 C# 面向对象的抽象、封装、继承、多态和动态联编五大特征的知识讲授，主要内容包括 C# 语言基础、流程控制、方法、数组与结构，以及类与对象、继承与多态、接口、委托、事件、泛型和文件等；第 9～12 章是 C#窗体应用程序开发，通过开发简单的计算器程序、记事本和学生信息管理系统，简单介绍了 Windows 窗体程序框架、对话框和常用控件、菜单和工具栏、数据库技术等 Windows 程序的开发方法。

本版是在第一版的基础上，根据大量的教学反馈意见和职业教育的发展需要，吸收行业发展的新知识、新技术和新方法，以适应培养技能型人才的新要求而编写的。结合在国家示范性院校、省示范性专业建设过程中所取得的课程改革成果，从如下三个方面进行了修订。

1. 编写思想和组织形式

本书的编写理念是：以就业为导向、以学生为主体，着眼于学生职业生涯发展，注重职业素养的培养。采用“项目驱动+知识学习+情境应用+自测练习”的四位一体教学模式组织教学内容。前 8 章安排“模仿练习”和“拓展训练”两个层次的实训环节，用于模仿、验证概念、语法规则及其应用，以适应自主学习、合作学习和个性化教学。第 9～12 章选择综合案例“简单计算器、记事本和学生信息管理系统”程序，分解提炼项目的功能模块和程序，按照 C# 面向对象知识结构分配到各个子项目中，伴随系统的设计、开发、优化到最后完善，使学生在项目实施的过程中掌握 C# 面向对象设计技术，在职业情境中实现知识构建。

2. 内容的新增和取舍

为突出“代码重用性”在实际项目开发中的重要意义，新增加了“泛型方法”、“泛型类”和“异常处理”等方面的内容；同时，还将“运算符重载”纳入了本书，以完善方法重载多态性的特征。另外，新增了一章“目录与文件操作（第 8 章）”，本书的 C# 面向对象程序设计知识系统更为完整，以满足开发实际项目的需要。

删除了第一版中的多媒体技术、图形编程、多线程和网络编程等知识的开发案例内容。其目的是突出 C#面向对象的五大特征，使教材内容与“面向对象程序设计”知识内容归属划分相一致。

3. 遵循“趣味、实用、易学”的特点，更换和新增了部分案例和项目

部分案例、项目来自企业和近几届全国蓝桥杯软件大赛的变形考题，充分反映产业升

级、技术进步和职业岗位变化的要求，从而使本书内容体现新知识、新技术和新方法。

本书可以作为高职高专院校计算机信息管理、软件技术等相关专业“C#面向对象程序设计”课程的教材，也可以作为应用型本科相关专业学习C#面向对象程序设计的教材。

尽管我们在本书的编写方面做了很多努力，但由于作者水平所限，不当之处在所难免，恳请各位读者批评指正，并将意见和建议及时反馈给我们，以便下次修订时改进（联系邮箱：360603935@qq.com）。

目 录

第1章

绪　论

学习目标

- 了解 C#程序设计语言发展过程及其特点
- 掌握 C#程序的基本结构和开发工具
- 掌握 C#程序的基本输入/输出语句
- 了解本课程的学习目标和技能要求

C#是为 .NET 平台应用开发而设计的一种现代编程语言，它继承了 C++、Java 等语言的优点，因此，问世后很快成为 Windows 应用开发语言中的宠儿。C#语言及其.NET 开发环境，被认为是近年来最重要的程序设计与开发新技术。

本书通过知识讲解、模仿练习、案例拓展和综合案例相结合的方式介绍学习内容，读者学完本书全部内容后，就可以掌握 C#面向对象程序设计的基本技能，并能开发简单的 Windows 应用程序。

1.1 C# 简　介

1.1.1 Visual C#.NET 简介

过去的几十年，计算机语言一直在加速发展。流行的说法是，20 世纪 80 年代学 C 语言，90 年代则是属于 C++的黄金时代。那么，21 世纪的理想语言是什么呢？

C#是微软公司在 2000 年 7 月发布的一种全新的简单、安全、面向对象的程序设计语言，它充分吸收了过去几十年中计算机科学发展的经验教训，体现了当前最新的程序设计技术的功能和精华，从 C#中以看到 C++、Visual Basic、Delphi、Java 等语言的优点。而且，C#于 2001 年 12 月获得了国际标准组织 ECMA 批准的 C#标准 ECMA-334，并于 2003 年 4 月被国际标准化组织 ISO 采纳为 ISO/IEC 23270 标准，这一切都为 C#发展推广奠定了坚实的基础。

通常，我们对于 C#和 Visual C#.NET 可以不加区分，但严格地说，两者是有区别的。C#只是一门语言或者说是一个标准，它是专门为微软的 .NET 平台设计的。作为 Visual Studio.NET 套件中的语言之一（该套件还包括 Visual Basic.NET、Visual C++.NET 和 J#.NET 语言），C#充当了微软推行 .NET 战略的拳头产品。但是，难保今后不会出现其他使用 C#语言的开发工

具（就像有 Visual C++和 C++ Builder 一样）。Visual C#.NET 则是指“C#语言+.NET 框架”。由于目前 C#必须和.NET 平台一同使用，因此，C#语言的学习与以往的语言学习有显著不同，对以往的编程工具可以先学习语言语法，再学习编程环境（例如，先学习 C++再使用 Visual C++）；而对 C#的学习一开始就要进入到 Visual Studio.NET 中进行编程学习。

C#语言是建立在.NET Framework 环境之上的，.NET Framework 是一个类库。其为 C#语言开发的应用程序提供了强大的类库支持，但是，它也支持 Visual Basic.NET 和 C++的托管方式。C#语言是 .NET Framework 平台首选的开发语言。也可以这样说，C#语言就是为 .NET Framework 平台而产生的语言。

注意

C#语言用于生成面向.NET 环境的代码，但是它不是.NET 环境的一部分，初学者很容易把 .NET 和 C#语言混为一谈。C++运行在 .NET Framework 之上的方式称为 C++的托管方式。由公共语言运行库环境 .NET Framework 执行，而不是直接由操作系统执行的代码就是托管代码。

1.1.2 Visual C#.NET 的特点

Visual C#.NET 以 Visual Studio.NET 为开发工具，包括交互式开发环境、可视化设计器、编译器和调试器。Visual C#.NET 的特征主要体现在以下两个方面。

1. 语言的变化

C#是在 C、C++的基础上改进而来的，作为一种全新的语言，它继承了 C、C++的强大功能，同时，吸收了 Visual Basic 语言简单易用的特点。虽然从整体来说，它基本继承了 C 语言的语法风格，但还是有明显的区别和改进，具体的语言变化细节将在本书相关地方体现。

2. .NET 框架支持

Visual C#.NET 完全集成了.NET 框架。.NET 框架封装了传统的 Windows API，为用户提供了全新的编程接口，并吸收了微软 20 世纪 90 年代中后期发展的各种新技术（COM+组件、ASP 技术、XML 支持等），为程序提供了对语言互操作性、垃圾回收、增强的安全性和改进支持。

其中，.NET 框架的支持是 C#运行和功能实现的基础，离开了.NET 框架，C#就是无源之水、无本之木。因此，对于 C#的学习，大部分精力要放在对.NET 框架的熟练掌握和应用方面。

1.1.3 C#与其他语言的关系

1. 与 C、C++的关系

就像名字上的相似性一样，C#是从 C、C++语言演变改进而来的，存在着血缘关系。C#基本上继承了 C 语言的语法风格，同时，又从 C++那里继承了面向对象特性。但是，不能够简单地将 C#看成 C++在.NET 框架上的翻版。毕竟，它们之间的不同点也是很明显的，

主要体现在以下三点。

第一，C#的对象模型已经面向 Internet 进行了重新设计，使用的是.NET 框架的类库，与 C++的对象模型结构完全不一样。因此，在编程中没有太大的相识感。

第二，C#语言不再提供对指针类型的支持，使得程序不能随便访问内存地址空间，从而使程序更加稳定可靠。用惯了指针的程序员需要一定的适应过程。

第三，在面向对象技术方面，C#不再支持多重继承，避免了以往类层次结构中由于多重继承带来的可怕后果。相应的功能可以通过对接口的多重继承来实现。

.NET 框架的支持，为 C#提供了一个强大的、易用的、逻辑结构一致的程序设计环境。同时，公共语言运行时（Common Language Runtime）为 C#程序提供了一个托管的运行时环境，使程序比以往更加稳定、安全。C#被设计用来满足新的在线环境功能的开发需要，解决新环境中的程序设计问题，而不是用来替代 C++，两者在未来一段时期将共存。

2. 与 Java 的关系

从整体上来说，C#与 Java 极其相似，甚至超过了与 C、C++的相似程度，不过，两者还是有区别的。例如，Java 通过虚拟机来实现平台的可移植性，而 C#则是首先被编译成一种中间语言（类似 Java 的字节码），然后，在执行时由公共语言运行中的即时编译器编译本机代码交由 CPU 处理。而且，Java 虚拟机只能执行 Java 程序，而即时编译器能够编译任何由.NET 框架支持的语言（如 C#、Visual Basic、J#）编写的程序。

鉴于 C#与 Java 的相似性，学习过 Java 的人员对 C#掌握起来不会感到太难，反过来也一样。由于两者支持平台（或者说运行环境）的不同，相互之间不会取代，而会长期共存。

1.2 Visual Studio 2010 开发环境

Microsoft Visual Studio 2005/2008/2010 等都可以作为 C#程序的开发平台。本书采用 Microsoft Visual Studio 2010 集成开发环境作为 C#程序的开发平台。

注意

本书的所有源程序代码都在 Visual Studio 2010 集成开发环境中测试通过，但也同样适合 Microsoft Visual Studio 2005/2008 等集成开发环境。读者可以自主选择不同版本的开发环境。

1.2.1 起始页窗口

单击“开始→程序→Microsoft Visual Studio 2010→Microsoft Visual Studio 2010”菜单项，启动 Microsoft Visual Studio 2010 开发环境，进入起始页。可以看到，整个起始页窗口主要包含 7 个区域：菜单栏、工具栏、工具箱、解决方案资源管理器、输出窗口、状态栏和中间的起始页，如图 1-1 所示。

下面分别简单介绍其中的几个主要区域，以帮助读者快速了解 Visual Studio 2010 开发

环境的使用。

图 1-1 Visual Studio 2010 起始页窗口

1.2.2 菜单栏和工具栏

1. 菜单栏

菜单栏中包括了 Visual Studio 2010 的大多数功能，菜单项目众多。Visual Studio 2010 的菜单随着不同的项目、不同的文件会进行动态变化。此处就不对所有的菜单进行一一介绍了，而只对常用的“文件”菜单、“编辑”菜单和“视图”菜单进行简单的介绍，以方便读者尽快熟悉 Visual Studio 2010 常用菜单的使用。

“文件”菜单提供了对 Visual Studio 2010 中文件操作的各种功能，其主要菜单项的功能如表 1-1 所示。

表 1-1 “文件”菜单项功能

菜 单 项	功 能
新建	新建项目、网站、文件等
打开	打开项目、网站、文件等
关闭	关闭当前页面
关闭解决方案	关闭当前解决方案
保存选定项	保存当前打开的项目或文件
将选定项另存为	将项目另存为其他项目或文件
全部保存	将所有未保存的文件保存
最近的文件	最近打开的文件
最近使用的项目和解决方案	列表最近打开的项目和解决方案
退出	退出 Visual Studio 2010

“编辑”菜单提供了 Visual Studio 2010 中大多数常见的文本编辑操作，其菜单项的功能

如表 1-2 所示。

表 1-2 “编辑”菜单项功能

菜 单 项	功 能
撤销	撤销上次的操作
重复	重复上次的操作
撤销上次全局操作	撤销上次全局操作
重复上次全局操作	重复上次全局操作
剪切	剪切选中内容到剪贴板
复制	复制选中内容到剪贴板
粘贴	粘贴剪贴板中的内容

“视图”菜单中各菜单项提供的功能比较简单，主要是对各种窗口的显示和隐藏的控制，此处就不一一列出其功能了。感兴趣的读者可以进行尝试，以获得直观的认识。

2. 工具栏

工具栏提供了最常用的功能按钮。对工具栏的熟悉可以大大节省工作时间，提高工作效率。同菜单一样，Visual Studio 2010 的工具栏也是动态变化的。随着工作的不同，工具栏也不尽相同。工具栏的内容还可以根据个人的使用习惯进行自定义，以方便不同开发人员的使用。图 1-2 给出的是位于菜单栏下的第一个工具栏。

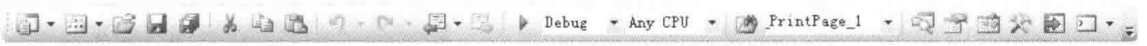

图 1-2 工具栏

这个工具栏提供了本节中提到的“文件”菜单、“编辑”菜单和“视图”菜单中的部分功能。另外，它还提供了部分编译选项功能。具体功能可以参见菜单栏中的介绍，此处就不一一列举了。感兴趣的读者可以将鼠标光标悬停于工具栏中相应的按钮上，观察 Visual Studio 2010 给出的提示，提示中一般会给出简单的帮助，可以协助读者快速熟悉相应功能。

1.2.3 工具箱

工具箱是 Visual Studio 2010 的重要工具，它提供了进行 Windows 窗体应用程序开发所必需的控件。通过工具箱，开发人员可以方便地进行可视化窗体设计，从而简化了程序设计，提高了工作效率。

图 1-3 所示的是工作箱的外观。如果看不到工具箱，可以从“视图”菜单中选择“工具箱”菜单项。展开工具箱中的“所有 Windows 窗体”列表，可以看到如图 1-4 所示的效果。

由于“所有 Windows 窗体”列表中的内容太多，故此处仅展示了部分内容。在图 1-4 中可以发现几乎任何程序中都会用到的 Button（按钮），还有常用的 CheckBox（复选框）等许多控件。在工具箱的其他列表中还有许多其他控件，读者可以自行查看。工具箱的使用将在以后的学习中逐渐地向读者介绍，此处只需要有一个直观的认识。

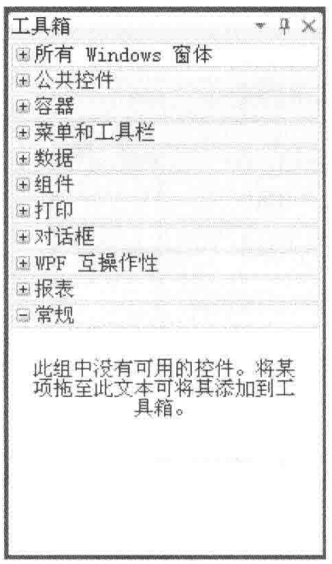

图 1-3　工具箱图示 1

图 1-4　工具箱图示 2

1.2.4 “类视图”面板

“类视图”面板是一个非常方便的工具，该面板提供了观察类结构的非常直观的功能。通过“类视图”面板可以对类的内部构造进行方便的查看。图 1-5 即为使用“类视图”面板查看类结构的示意图。

读者还可以对“类视图”面板进行自定义设置。设置方法为右击“类视图”面板，之后根据提示进行相应的操作。由于设置方法比较简单，此处就不进行介绍了。

1.2.5 代码编辑器/窗体设计器

代码编辑器是 Visual Studio 中开发人员需要面对和消耗时间最多的一个工具。该工具提供了强大的代码编辑功能，当打开或新建一个项目时，就会在“起始页”的位置打开代码编辑器和窗体设计器。如图 1-6 所示，窗体设计器用来设计程序的用户界面，初始显示

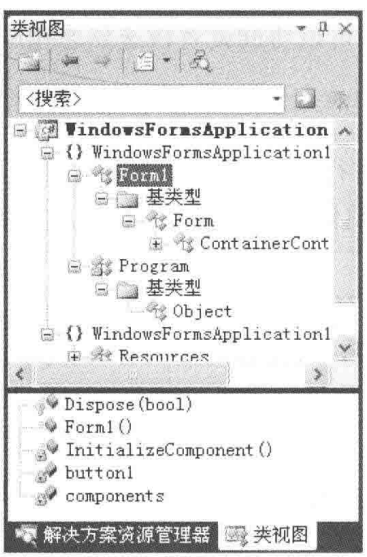

图 1-5　“类视图”面板

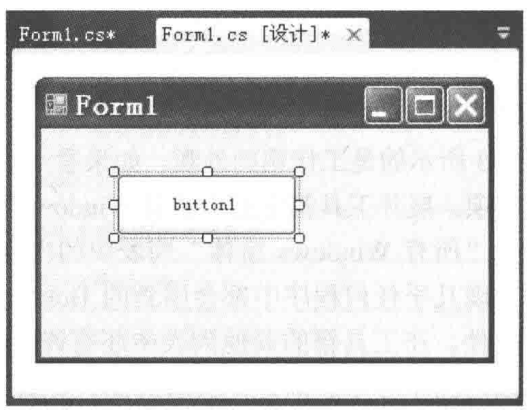

图 1-6　代码编辑器/窗体设计器

一个空的窗体代码编辑器用来编辑项目的源代码。可以通过选择“视图”菜单中的“代码”和“设计器”菜单项在代码编辑器和窗体设计器之间切换，也可以单击解决方案资源管理器窗口上部的“查看代码”和“视图设计器”按钮进行切换。

1.2.6 “属性”面板

“属性”面板是 Visual Studio 2010 中另一个重要的工具。该面板为 Windows 窗体应用程序的开发提供了简单的属性修改方式。对窗体应用程序开发中的各个控件的属性修改都可以由“属性”面板来完成。“属性”面板不仅提供了属性的修改功能，还提供了事件的管理功能。“属性”面板可以管理控件的事件，方便编程时对事件的处理。

“属性”面板同时采用了两种方式管理属性和方法，即按分类方式和按字母顺序方式，读者可以根据自己的习惯采取不同的方式。面板的下方还有简单的帮助内容，方便开发人员对控件的属性和方法进行操作和修改。图 1-7 是按分类方式列出窗体属性的“属性”面板。

在“布局”列表中可以看到窗体的位置（Location）、大小（Size）等属性，通过“属性”面板可以对其进行方便的设置。

1.2.7 定制开发环境

除了前面介绍的 Visual Studio 默认窗口布局，用户还可以根据自己的需要自由地定制环境布局。最常用的方法是选择“工具”菜单中的“自定义”和“选项”两个命令。

1）选择“自定义”命令时，会打开“自定义”对话框，如图 1-8 所示。其中有两个选项卡：“工具栏”和“命令”。它们的使用与 Word 中类似，这里就不介绍了。

图 1-7 “属性”面板

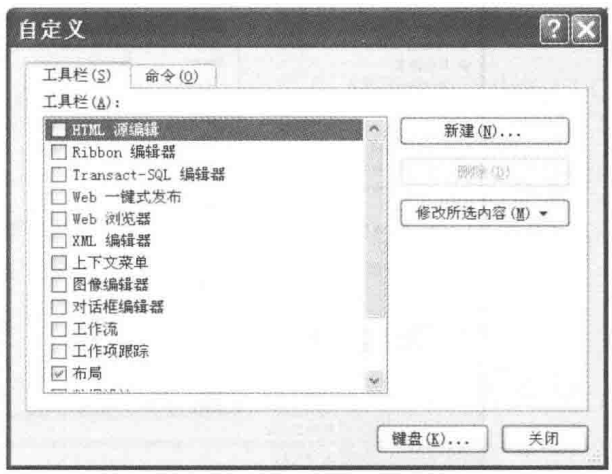

图 1-8 “自定义”对话框

2）选择“选项”命令时，会打开“选项”对话框，如图 1-9 所示，这里是设置继承开发环境的重要场所。

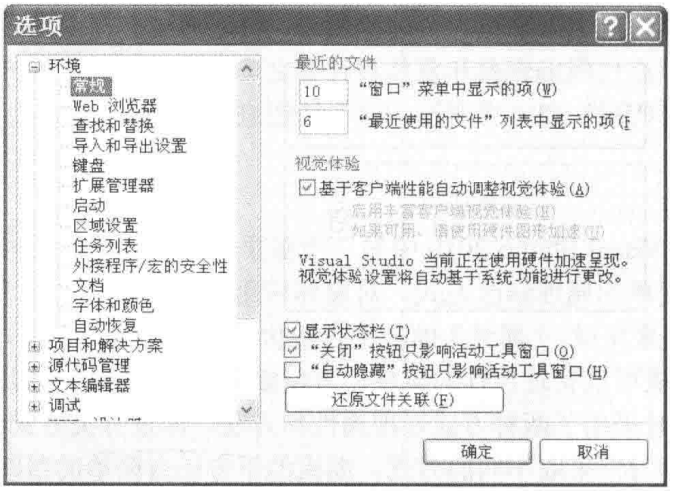

图 1-9 “选项”对话框

1.3 简单的 C#程序

下面来编写本书中的第一个 C#程序，了解 C#程序的结构和上机开发步骤。

1.3.1 创建 C#程序项目

下面将使用 Visual Studio 2010 提供的项目模板来创建一个控制台应用程序（Console Application）。这个程序将在窗口中显示“欢迎使用 C#”字符串。操作步骤如下。

1）选择“文件→新建→项目”命令，打开“新建项目”对话框，如图 1-10 所示。

图 1-10 “新建项目”对话框