GENERAL EDUCATION

高等学校通识教育系列教材

C#程序设计经典教程（第三版）

罗福强　杨剑　张敏辉　编著

清华大学出版社

高等学校通识教育系列教材

C#程序设计经典教程（第三版）

罗福强　杨剑　张敏辉　编著

清华大学出版社
北京

内 容 简 介

C#经过近20年的不断发展和完善，已经成为一种跨平台的开发语言，如今它不仅能用于开发Windows系统中的应用程序，还可以用来开发Android、iOS、Windows Phone和Mac App应用程序，甚至还能开发物联网嵌入式系统。

全书共14章，在介绍C#语法的基础之上，深入阐述面向对象的程序设计方法、基于事件驱动的程序设计方法以及面向服务的程序设计方法。不仅如此，本书还全面揭示C#的各种应用技术，包括Windows程序设计技术、数据库编程技术、文件操作与编程技术、XML与LINQ高级数据访问技术、面向服务编程技术和多媒体处理编程技术等。本书内容丰富，可操作性强，叙述简洁流畅，语言通俗易懂，所有实例都经过精心设计，能够使学生轻松愉快地掌握C#的基本语法、编程方法和应用技巧。

本书可作为高等院校计算机相关专业的教材，也可作为初、中级读者和培训班学员的参考用书。

图书在版编目(CIP)数据

C#程序设计经典教程/罗福强等编著. —3版. —北京：清华大学出版社，2018
(高等学校通识教育系列教材)
ISBN 978-7-302-49807-0

Ⅰ. ①C… Ⅱ. ①罗… Ⅲ. ①C语言－程序设计－高等学校－教材 Ⅳ. ①TP312.8

中国版本图书馆CIP数据核字(2018)第037258号

责任编辑：刘向威
封面设计：文　静
责任校对：李建庄
责任印制：丛怀宇

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：三河市君旺印务有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：25.5　　**字　　数**：621千字
版　　次：2012年2月第1版　2018年7月第3版　　**印　　次**：2018年7月第1次印刷
印　　数：1～1500
定　　价：65.00元

产品编号：077519-01

前 言

C#是由微软公司推出的完全面向对象的计算机高级语言。经过近20年的发展，如今它不仅能用于开发传统Windows环境中的应用程序，还可以用来开发原生的Android、iOS、Windows Phone和Mac App应用程序，甚至还能整合Azure或Hadoop技术开发云计算和大数据应用系统。相对于C++来说，C#更容易被理解和接受；相对于Java来说，C#更好用，开发软件的效率更高。

本书自2012年2月出版第1版以来，受到广大师生的欢迎。2014年我们组织修订，推出第2版。如今，3年过去了，微软公司已经推出多个C#新版本，使C#具有大量新特性。为此，我们再次组织编写本教材第3版，针对第2版主要进行以下修订。

(1) 在第1章中增加.NET技术体系结构的介绍，使读者对.NET技术有更全面的了解。为了便于读者更早和更快地理解C#程序，把C#程序的特点独立编成1节。

(2) 如今海量的文本日志成为构建大数据技术的主要研究内容，特征提取与转换、数据分析与挖掘成为程序设计的重点，为此，第2章加强了字符串的内容，包括文本格式化处理的内容等。

(3) 自C# 3.0开始，C#添加很多新特性，例如，引入表达式主体(expression-bodied)来简化对象属性和索引器的定义，引入Lambda表达式简化匿名函数的定义，不仅降低了C#程序的复杂度，还使C#源代码更加优雅。

(4) 云计算和大数据技术的基础是面向服务的程序设计思想。要想快速适应云计算和大数据时代的新要求，必须更早地了解或熟悉这种新思想。为此本书第13章剔除原来的一部分内容，增加面向服务的编程技术。

本书第3版以Visual Studio .NET 2017和C# 7.0为蓝本，深入介绍C#语言及其应用。全书共分14章，基本上覆盖了C#的主要领域，在讲解C#语法的基础上，不仅阐述面向对象、基于事件驱动和面向服务的3种不同的程序设计思想，还全面展现C#的具体应用技术，包括Windows程序设计、数据库应用编程、文件操作与编程、XML与LINQ高级数据访问、面向服务编程和多媒体处理编程技术等。

本书继续保持以下优点：第一，面向应用型本科院校学生，立足于把C#的语法讲透彻、讲清楚，文字叙述尽量简练；第二，重点围绕面向对象程序设计思想和可视化的Windows程序设计方法展开教学内容；第三，书中所有案例均精心设计，不仅代码完整，还贴近学生实际生活；第四，坚持零起点原则，学生可以在没有C/C++基础的情况下使用本书；第五，坚持应用为纲，全面展示C#在各应用领域的编程技巧。

本书可作为高等院校Visual C# .NET课程的教材或参考资料，也可供软件开发人员参考使用。

本书由四川大学锦城学院的罗福强老师主持修订。参与本书编写的还有杨剑、张敏辉、熊永福、陈虹君、李瑶、赵力衡等老师。本书长期以来获得清华大学出版社的各级领导的重视和支持,也获得了作者所在单位领导的大力支持。在此,我们对支持本书编写出版并提供过大量帮助的所有人员表示诚挚的感谢!

由于时间仓促,书中难免有不妥之处,我们殷切地期望读者提出宝贵的意见。

编　者

2018 年 4 月

目 录

第1章 C#概述

总体要求

- 了解C#语言的特点及其发展。
- 了解C#应用程序的结构及其特点。

相关知识点

- 了解计算机软件、计算机语言及分类的知识。
- 熟悉Windows系统基础知识及操作。

学习重点

- C#程序的结构与特点。

学习难点

- 控制台应用程序与Win32应用程序的区别。

1.1 .NET与C#概述

1.1.1 .NET概述

1. .NET技术体系结构

.NET平台是微软公司在20世纪末为了迎接互联网的挑战而推出的Windows应用程序运行平台。经过近20年的发展,它如今已经成为一个可以跨越任何硬件系统的开发平台,在这个平台上可以构建和运行Windows应用程序、Web应用程序、Azure云应用程序、移动App应用程序、Unity游戏等。.NET建立在开放体系结构基础之上,集Microsoft在软件领域的主要技术成就于一身,如图1-1所示。

.NET技术的核心是.NET Framework,它为.NET平台下应用程序的运行提供基本框架,如果把Windows操作系统比作一幢摩天大楼的地基,那么.NET Framework就是摩天大楼中由钢筋和混凝土搭成的框架。为了实现跨平台运行的目标,微软公司新推出了.NET Core,其核心.NET Core Framework是参考.NET Framework重新开发的,它支持Windows,Mac OS,Linux等操作系统,可以用于嵌入式或物联网解决方案之中。为了使.NET应用程序能在智能终端设备之上运行,微软启动了Mono项目,该项目可以看作是.NET Framework的开源实现。

.NET Framework以微软的Windows操作系统为基础,由不同的组件组成(如图1-1所示),能够与Windows的各种应用程序服务组件(如消息队列服务、COM+组件服务、Internet信息服务(IIS)、Windows管理工具等)整合,来开发各种应用程序。

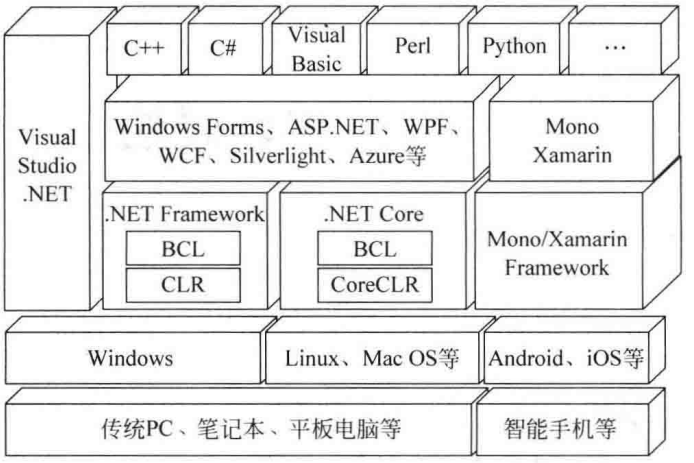

图 1-1 .NET 平台的体系结构

在.NET Framework 的最顶层是程序设计语言,.NET Framework 支持诸如 VB、C#、C++、F#、Perl、Python 等几十种高级程序设计语言。在 Visual Studio .NET 开发环境中,可直接使用 VB、C#、C++、F#、TypeScript、Python 等多种语言开发应用程序;利用新推出的移动应用跨平台开发插件 Xamarin[①],用户还可以直接开发 iOS、Android、Windows Phone 和 Mac App 等应用,而不需要转移到 Eclipse 或者额外购买 Mac 和使用 Xcode。

.NET Framework 具有两个主要组件:公共语言运行时(Common Language Runtime,CLR)和基础类库(Base Class Lib,BCL),除此之外还包括 ADO.NET、ASP.NET、WCF、Azure、Workflow 框架等。

CLR 是.NET Framework 的基础,是应用程序与操作系统之间的"中间人",它为应用程序提供内存管理、线程管理和远程处理等核心服务。在.NET 平台上,应用程序无论使用何种语言编写,在编译时都会被语言编译器编译成 MSIL(微软中间语言代码),在运行应用程序时 CLR 自动启用 JIT(Just in Time)编译器将 MSIL 再次编译成操作系统能够识别的本地机器语言代码(简称本地代码),然后运行并返回运行结果。因此,CLR 是所有.NET 应用程序的托管环境。这种运行在.NET 之上的应用程序被称为托管应用程序,而传统的直接在操作系统基础之上运行的应用程序则被称为非托管应用程序。

BCL 类库是一个综合性的面向对象的可重用类型集合,包括集合类、文件系统处理类、XML 处理类、网络通信接口类、异步 Task 类等,利用它可以开发多种应用程序,包括传统的命令行、图形用户界面(GUI)应用程序、Web 应用程序等。

ADO.NET 是.NET Framework 提供的微软新一代的面向对象数据处理技术,利用它可以简便、快捷地开发数据库应用程序。

ASP.NET 是.NET Framework 提供的全新的 Web 应用程序开发技术,利用它开发 Web 应用程序如同开发 Windows 应用程序一样简单。

WCF(Windows Communication Foundation)、WPF(Windows Presentation Foundation)以及 Silverlight 等技术是微软推出的全新.NET 技术。WCF 可以理解 Windows 通信接口,它整

① Xamarin 始创于 2011 年,2016 年 2 月被微软公司收购。如今,Xamarin 已经被微软内置到 Visual Studio .NET 2017 之中。此外,微软还开源了 Xamarin SDK,免费供用户使用。

合了 TCP/IP、XML、SOAP、JSON 等技术，因此简化了 XML Web 服务的设计与实现。WPF 为用户界面、2D/3D 图形、文档和媒体提供了统一的描述和操作方法。Silverlight 为开发具有专业图形、音频和视频处理的 Web 应用程序提供了全新的解决方案。

2. .NET Framework 的优点

.NET Framework 的目标是为应用程序开发人员提供了一个与平台无关的开发环境，具有以下优点。

(1) 基于 Web 的标准

.NET Framework 完全支持现有的 Internet 技术，包括 HTML(超文本标记语言)、HTTP(超文本传输协议)、XML(可扩展标记语言)、SOAP(简单对象访问协议)、XSLT(可扩展样式表语言转换)、XPath(XML 路径语言)、JSON(JavaScript 对象表示方法)和其他 Web 标准。

(2) 使用统一的应用程序模型

任何与.NET 兼容的语言都可以使用.NET Framework 类库。.NET Framework 为 Windows 应用程序、Web 应用程序、云计算服务、跨平台的智能手机应用提供了统一的应用程序模型，因此同一段代码可被这些应用程序无障碍地使用。

(3) 便于开发人员使用

在.NET Framework 中，代码被组织在不同的命名空间和类中，而命名空间采用树形结构，以便开发人员引用。当开发人员调用.NET Framework 类库的类时，只需将该类属性命名空间添加到引用解决方案中即可。

(4) 可扩展类

.NET Framework 提供了通用类型系统，它根据面向对象的思想把一个命名空间或类中代码的实现细节隐藏，开发人员可以通过继承来访问类库中的类，也可以扩展类库中的类，甚至构建自己的类库。

1.1.2 C# 语言的发展

在过去的 30 年里，C 和 C++已经成为在商业软件的开发领域中使用最广泛的语言。它们为程序员提供了十分灵活的操作，不过同时也牺牲了一定的效率。与 Visual Basic 等语言相比，同等级别的 C/C++应用程序往往需要更长时间来开发。由于 C/C++语言的复杂性，许多程序员都试图寻找一种新的语言，希望能在功能与效率之间找到一个更为理想的权衡点。

目前有些语言，以牺牲灵活性的代价来提高效率。可是这些灵活性正是 C/C++程序员所需要的。这些解决方案对编程人员的限制过多(如屏蔽一些底层代码控制的机制)，其所提供的功能难以令人满意。这些语言无法方便地同原来的系统交互，也无法与当前的网络编程很好地结合。

对于 C/C++用户来说，最理想的解决方案无疑是在快速开发的同时又可以调用底层平台的所有功能。他们想要一种和最新的网络标准保持同步并且能和已有的应用程序良好整合的环境。另外，一些 C/C++开发人员还需要在必要的时候进行一些底层的编程。

C# (读作 C Sharp)是微软对这一问题的解决方案。C#是一种最新的、面向对象的编程语言。它是一种简单但功能强大的编程语言，使程序员可以快速地编写各种基于

Microsoft .NET 平台的应用程序。

它从 C 和 C++语言演化而来。它在语句、表达式和运算符方面使用了许多 C++功能。它在类型安全性、版本转换、事件和垃圾回收等方面进行了相当大的改进和创新。它提供对常用 API(例如. NET Framework、COM+等)的访问。

C#自推出以来,已得到不断的改进和优化,通常同. NET Framework 一起,随新版的 Visual Studio .NET 的发布而更新。目前,C#最新的版本是 C# 7.0,该版本是 2017 年 3 月 8 日微软公司正式发布 Visual Studio .NET 2017 时发布的。

本书以. NET Framework 4.6.2 和 Visual Studio .NET 2017 为范本,所有案例均在 Visual Studio .NET 2017 中经过调试运行无误。

1.1.3 C#语言的特点

C#是一种简洁、类型安全的面向对象的语言,开发人员可以用它来构建运行在. NET Framework 上的各种安全、可靠的应用程序,包括控制台应用程序、Windows 窗体应用程序、Web 应用程序等。借助 Xamarin 插件,C#还可以用于开发 iOS、Android、Windows Phone 和 Mac App 等应用等。

作为一种面向对象的语言,C#支持封装、继承和多态性的概念。所有的变量和方法,包括 Main 方法(应用程序的入口点),都封装在类定义中。C#程序的生成过程比 C 和 C++简单,比 Java 更为灵活,没有单独的头文件,也不要求按照特定顺序声明方法和类型。C#源文件可以定义任意数量的类、结构、接口和事件。

相对其他计算机程序设计语言来说,C#具有如下优点。

(1) C#是一种精确、简单、类型安全、面向对象的语言。正是由于 C#面向对象的卓越设计,使它成为构建各种应用程序组件的理想之选——无论是高级的商业对象还是系统级的应用程序。

(2) C#具有生成持久系统级组件的能力,提供 COM+或其他技术平台支持以集成现有代码,提供垃圾回收和类型安全以实现应用程序的可靠性,提供内部代码信任机制以保证应用程序的安全性。

(3) C#利用. NET Framework 的通用类型系统能够与其他程序设计语言交互操作。C#应用程序能跨语言、跨平台互相调用。使用 C#语言可实现具有不同专业技术背景的人员协同工作,完成软件系统的设计和开发。

(4) C#支持 MSMQ(微软消息队列服务)、COM+组件服务、WCF 服务和. NET Framework。使用 C#语言,一方面实现组件之间的相互调用,也就实现了使用不同软件技术开发组件之间的集成应用。另一方面能够把传统的组件转化为 XML Web 服务,实现了组件之间的跨互联网调用。

(5) C#语言允许自定义数据类型,用来扩展元数据。这些元数据可以应用于任何对象。项目构建者可以定义领域特有的属性并把他们应用于任何语言元素——类、接口等。然后,开发人员可以编程检查每个元素的属性。这样,很多工作都变得方便多了,比如编写一个小工具,用来自动检查每个类或接口是否被正确定义为某个抽象商业对象的一部分,或者只是创建一份基于对象领域特有属性的报表。定制的元数据和程序代码之间的紧密对应有助于加强程序的预期行为和真正实现之间的对应关系。