

*Eighth Edition*

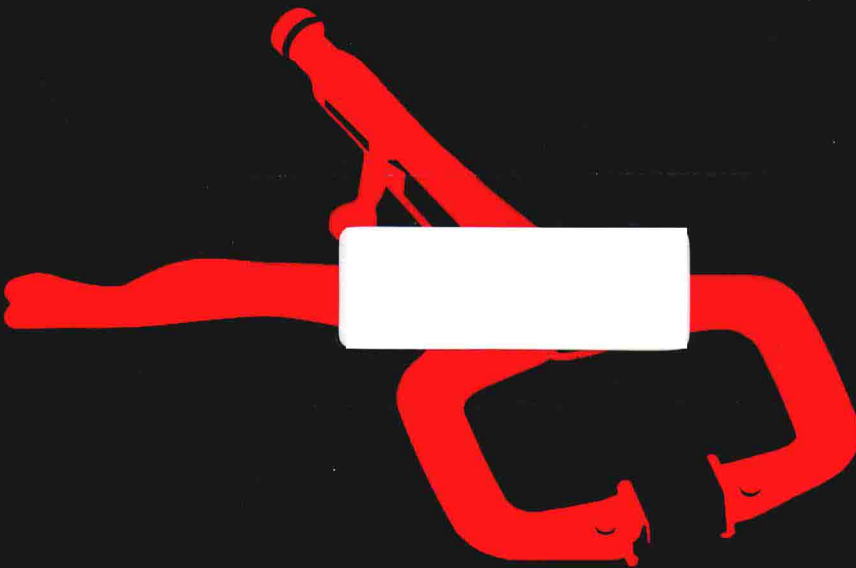
- 承袭Step-by-Step风格，完美演绎C#精髓
- 久经考验、深受国内外读者喜爱的C#入门经典

# Visual C#

## 从入门到精通 (第8版)

John Sharp 著  
周靖 译

Step by step



清华大学出版社

# Visual C#

## 从入门到精通 (第8版)

John Sharp 著  
周靖 译

清华大学出版社  
北京

## 内 容 简 介

C#作为微软的旗舰编程语言,深受程序员喜爱,是编写高效应用程序的首选语言。Visual C# 2015 提供了大量新功能,本书围绕语言的基础知识和这些新功能全面介绍了如何利用 Visual Studio 2015 和 .NET Framework 4.6 编写 C#应用程序。本书沿袭深受读者欢迎的 Step by Step 风格,通过合理的练习引导读者逐步构建在 Windows 10 上运行的应用程序、访问 SQL Server 数据库以及开发多线程应用等。

全书共 27 章,结构清晰,叙述清楚。所有练习均在 Visual Studio 2015 简体中文版上进行过全面演练。无论是刚开始接触面向对象编程的新手,还是打算迁移到 C#的 C、C++或 Java 程序员,都可以从本书汲取到新的知识。迅速掌握 C#编程技术。

Authorized translation from the English language edition, entitled MICROSOFT VISUAL C# STEP BY STEP, 8th Edition, 9781509301041 by SHARP, JOHN, published by Pearson Education, Inc, publishing as Microsoft Press, Copyright © 2015 by John Sharp.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc. CHINESE SIMPLIFIED language edition published by TSINGHUA UNIVERSITY PRESS LIMITED Copyright © 2015.

本书中文简体翻译版由 Pearson Education 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2016-3540

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

Visual C#从入门到精通(第 8 版)/(英)夏普(Sharp, J.)著. 周靖译. —北京:清华大学出版社, 2016  
书名原文: Microsoft Visual C# Step by Step, 8th Edition  
ISBN 978-7-302-43527-3

I. ①V… II. ①夏… ②周… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2016)第 081011 号

责任编辑: 文开琪

装帧设计: 杨玉兰

责任校对: 李玉萍

责任印制: 杨 艳

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](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 42 字 数: 899 千字

版 次: 2016 年 6 月第 1 版 印 次: 2016 年 6 月第 1 次印刷

印 数: 1~3000

定 价: 129.00 元

产品编号: 068360-01

# 前 言

Microsoft Visual C#是一种功能强大、使用简单的语言，主要面向需要使用 Microsoft .NET Framework 来创建应用程序的开发者。它在 C++和 Microsoft Visual Basic 的基础上去芜存菁，最终形成一种更加清晰、更富有逻辑的语言。C# 1.0 于 2001 年亮相。几年后随着 C# 2.0 和 Visual Studio 2005 的问世，语言新增了几个重要功能，包括泛型、迭代器和匿名方法等。随同 Visual Studio 2008 发布的 C# 3.0 添加了更多功能，包括扩展方法、Lambda 表达式以及语言集成查询(Language Integrated Query, LINQ)。2010 年发布的 C# 4.0 继续增强，改善了与其他语言和技术的互操作性。新增功能包括具名参数和可选参数，另外还有 `dynamic` 类型(指示语言的“运行时”对对象进行晚期绑定)。在随同 C# 4.0 发布的 .NET Framework 中，最重要的新功能就是“任务并行库”(Task Parallel Library, TPL)。可用 TPL 构建具有良好伸缩性的应用程序，从而快速和简单地发挥出多核处理器的潜力。C# 5.0 则通过 `async` 方法修饰符和 `await` 操作符提供了对异步任务的原生支持。C# 6.0 是一次增量式升级，提供了许多有利于简化开发的功能，包括字符串插值(再也不需要 `String.Format` 了)，改进的属性实现方式，表达式主体方法等。本书将一一阐述。

Microsoft 的另一个里程碑事件是 Windows 10 的发布。新的操作系统集成了以前版本的 Windows 最好(和最受人喜爱)的元素，支持高交互性的应用。这些应用实现了数据共享和协作，还能连接在云端运行的服务。Windows 10 最引人注目的是对(通用 Windows 平台, Universal Windows Platform, UWP)应用的支持。这种应用设计在任何 Windows 10 设备上运行，无论是全功能的桌面系统、笔记本和平板，还是资源有限的智能手机和物联网设备。熟悉 C#的核心功能后，下一步就是掌握如何开发能在所有这些平台上运行的应用。

语音激活是另一个值得关注的功能。Windows 10 提供了 Cortana(小娜)——你的个人语音数字助理。可将自己的应用和 Cortana 集成来实现数据搜索和其他操作。虽然自然语音分析天生就很复杂，但让应用响应 Cortana 的请求却令人惊讶地简单。详情将在第 26 章描述。另外，从大规模企业应用程序到手机上运行的移动应用，云已成为许多系统架构的重要元素，本书最后一章会讲解如何开发基于云的应用。

Visual Studio 2015 开发环境使这些强大功能变得容易使用，大量新向导和增强显著提高了开发效率。写作这本书给我带来了许多乐趣，希望你的阅读亦是如此！

下面列出和本书上一版《Visual C# 2013 从入门到精通》相比的主要变化：

- 代码字体改用 `Consolas`，可以更好地保护程序员的视力
- 针对 Windows 10 上的 UWP 应用开发进行了全面重写，不需要针对不同 Windows 平台维护不同代码库
- 字符串插值，所有代码都修改为可以利用该技术，参见 2.5.1 节



- 表达式主体方法，参见 3.1.3 节
- 异常过滤器，参见 6.2.3 节
- using static 语句，参见 7.5.4 节
- 可以创建只读的自动属性了，参见 15.5 节
- 索引器风格的对象初始化列表，声明并初始化字典时特别好用，参见 18.2 节
- 利用可视状态触发器大幅简化视图大小自适应代码，参见 25.2.1 节
- nameof 操作符，参见 26.1.2 节
- 和小娜一起玩语音搜索，参见 26.2 节
- 与时俱进使用 Microsoft Azure 创建云端数据库，参见第 27 章

## 本书面向的读者

本书假定你要使用 Visual Studio 2015 和 .NET Framework 4.6 学习基础的 C# 编程知识。学完本书后，会对 C# 有一个全面和透彻的理解，会用它开发响应灵敏、易于伸缩的 Windows 10 应用程序。

## 本书不面向的读者

本书面向刚开始用 C# 进行开发的人士，重点放在 C# 语言上面。本书不涉及企业级 Windows 应用程序的开发技术，比如 ADO.NET、ASP.NET、Windows Communication Foundation 或者 Workflow Foundation。要了解这些方面的知识，可参考微软出版社的其他书籍。

## 本书的组织

全书分为四部分。

- 第 I 部分“Visual C# 和 Visual Studio 2015 概述”介绍 C# 语言的核心语法，还演示了 Visual Studio 编程环境。
- 第 II 部分“理解 C# 对象模型”深入探讨如何用 C# 创建和管理新类型，如何管理这些类型引用的资源。
- 第 III 部分“用 C# 定义可扩展类型”全面讨论如何利用 C# 语言元素来构建能在多个应用程序中重用的类型。

- 第 IV 部分“用 C#构建 UWP 应用”描述通用 Windows 10 编程模型，以及如何用 C#为新模型构建交互式应用程序。

## 导读

本书帮助你掌握多个基本领域的开发技能。无论刚开始学习编程，还是从另一种语言(C、C++、Java 或 Visual Basic)转向 C#，本书都能提供帮助。参考下表找到最佳起点。

读者类型	步骤
面向对象编程的新手	<ol style="list-style-type: none"> <li>1. 按照“示例代码”一节的步骤安装练习文件</li> <li>2. 顺序阅读第 I 部分、第 II 部分和第 III 部分</li> <li>3. 有了一定经验后，如果有兴趣，继续完成第 IV 部分的学习</li> </ol>
熟悉 C 语言等过程编程语言，但新涉足 C#	<ol style="list-style-type: none"> <li>1. 按照“示例代码”一节的步骤安装练习文件</li> <li>2. 略读前 5 章来获得对 C#和 Visual Studio 2015 的大致印象，重点阅读第 6 章~第 22 章</li> <li>3. 阅读第 IV 部分了解如何构建可伸缩的 Windows 8.1 应用程序</li> </ol>
从面向对象语言 C++或 Java 等迁移到 C#	<ol style="list-style-type: none"> <li>1. 按照“示例代码”一节的步骤安装练习文件</li> <li>2. 略读前 7 章，获得对 C#语言和 Visual Studio 2015 的大致印象，重点阅读第 8~22 章</li> <li>3. 阅读第 IV 部分了解如何构建 UWP 应用</li> </ol>
从 Visual Basic 迁移到 C#	<ol style="list-style-type: none"> <li>1. 按照“示例代码”一节的步骤安装练习文件</li> <li>2. 顺序阅读第 I 部分、第 II 部分和第 III 部分</li> <li>3. UWP 应用的开发请阅读第 IV 部分</li> <li>4. 阅读每章末尾的“快速参考”小节，了解 C#和 Visual Studio 2015 特有的构造</li> </ol>
做完所有练习后再将本书用作为参考书	<ol style="list-style-type: none"> <li>1. 按目录查主题</li> <li>2. 阅读章末“快速参考”，查看语法和技术要点归纳</li> </ol>

本书大多数章节都通过实例方便读者巩固刚学到的知识。无论感兴趣的是哪个主题，都注意先下载并安装好示例代码。

## 本书的约定和特色

本书通过一些约定来增强内容的可读性，以便于读者理解。

- 每个练习都用编号的操作步骤来完成。

- “注意”等特色段落提供了成功完成一个步骤需要了解的额外信息或替代方案。
- 要求读者输入的文本加粗显示。
- 两个键名之间的加号(+)意味着必须同时按下这两个键。例如，“按组合键 Alt+Tab”意味着按住 Alt 键，再按 Tab 键。
- 描述菜单操作时，采用“文件”|“打开”的形式，意思是从“文件”菜单中选择“打开”命令。

## 系统需求

为了完成本书的练习，需准备以下硬件和软件：

- Windows 10 专业版或更高版本
- Visual Studio 2015 社区版、专业版或企业版

---

 **重要提示** 需要在 Visual Studio 2015 中安装 Windows 10 开发工具。

---

- 1.6 GHz 或更快的处理器(推荐 2 GHz 以上)
- 32 位操作系统至少 1 GB RAM, 64 位至少 2 GB RAM。在虚拟机中运行再加 512 MB
- 10 GB 可用硬盘空间
- 支持 DirectX 9 的显示卡, 1024 × 768 或更高分辨率
- 光盘驱动器(如果从光盘安装 Visual Studio)
- 下载软件和示例代码需要 Internet 连接

取决于 Windows 配置，可能需要以管理员身份安装和配置 Visual Studio 2015。

计算机需启用开发人员模式以创建和运行 UWP 应用。详情参考“启用设备进行开发”(<https://msdn.microsoft.com/library/windows/apps/dn706236.aspx>)

## 示例代码

本书大多数章节都包含互动练习供你练手。从以下网址下载所有示例项目(包括练习前后两种格式)：

<http://aka.ms/sharp8e/companioncontent>


<http://pan.baidu.com/s/1yd43W>

<http://transbot.ys168.com>

安装示例代码很简单，将下载的 CSharpSBS.zip 文件解压到“文档”文件夹即可。<sup>①</sup>

## 使用示例代码

本书每一章都解释了在什么时候以及如何使用练习文件。需要练习文件时，书中会给出相应的指示，帮助你打开正确文件。

 **重要提示** 许多例子都依赖 NuGet 包。这些包在首次生成项目时自动下载。所以如果首次打开一个项目而且不进行生成，Visual Studio 可能报告大量引用无法解析的错误。生成一次项目即可完成引用的解析，错误会消失不见。

下表总结了本书用到的所有 Visual Studio 2015 项目和解决方案，它们以文件夹的形式进行分组以便查找。练习通常会为同一个项目提供初始文件和完成之后的版本。已完成的项目存储在带有 - Complete 后缀的文件夹中。

项目/解决方案	说明
<b>第 1 章</b>	
TextHello	作为第一个项目，它指导你创建一个简单程序来显示欢迎文本
Hello	打开一个窗口，提示用户输入姓名并显示个性化的欢迎辞
<b>第 2 章</b>	
PrimitiveDataTypes	演示如何使用基元类型声明变量，如何向变量赋值，如何在窗口中显示值
MathsOperators	演示算术操作符(+、-、*、/、%)
<b>第 3 章</b>	
Methods	改进上个项目的代码，体会如何使用方法来建立代码的结构
DailyRate	指导你写自己的方法，执行方法，使用 Visual Studio 2015 调试器来单步执行方法
DailyRate Using Optional Parameters	演示如何让方法获取可选参数，如何使用具名参数来调用方法
<b>第 4 章</b>	
Selection	演示如何用嵌套 if 语句实现复杂逻辑，例如比较两个日期的相等性
SwitchStatement	这个简单的程序使用一个 switch 语句将字符转换成相应的 XML 形式
<b>第 5 章</b>	
WhileStatement	用 while 语句逐行读取源文件，在窗体上的文本框中显示每一行
DoStatement	使用 do 语句将十进制数转换成八进制
<b>第 6 章</b>	
MathsOperators	改进第 2 章的 MathsOperators 项目，试验会造成程序执行失败的各种未处理异常。然后用 try 和 catch 关键字使应用程序更健壮，防止因为错误输入或操作而失败

<sup>①</sup> 译注：本书将路径“C:\Users\YourName\Documents”简称为“文档”文件夹。可在 Windows 文件资源管理器的地址栏输入环境变量%UserProfile%\Documents 打开该文件夹。



续表

项目/解决方案	说明
第 7 章	
Classes	演示如何定义自己的类, 为它添加公共构造器、方法和私有字段; 还演示如何用 <code>new</code> 关键字创建类的实例, 如何定义静态方法和字段
第 8 章	
Parameters	演示值类型和引用类型的参数的区别, 还演示如何使用 <code>ref</code> 和 <code>out</code> 关键字
第 9 章	
StructsAndEnums	定义结构来表示日期
第 10 章	
Cards	使用数组建模纸牌游戏中的一手牌
第 11 章	
ParamsArrays	演示如何使用 <code>params</code> 关键字使方法能接受任意数量的实参
第 12 章	
Vehicles	用继承创建简单交通工具类, 还演示如何定义虚方法
ExtensionMethod	演示如何为 <code>int</code> 类型创建扩展方法, 允许将整数从十进制转换成其他进制
第 13 章	
Drawing	实现图形绘图包的一部分。用接口定义要由几何图形对象公开并实现的方法
Drawing Using Interfaces	扩展 <code>Drawing</code> 项目, 将几何图形对象的常用功能集成到抽象类中
第 14 章	
GarbageCollectionDemo	演示如何使用 <code>Dispose</code> 模式实现异常安全的资源清理
第 15 章	
Drawing Using Properties	扩展第 13 章的 <code>Drawing</code> 项目, 用属性封装类的数据
AutomaticProperties	演示如何为类创建自动属性, 如何用它们初始化类的实例
第 16 章	
Indexers	该项目使用了两个索引器, 一个根据姓名查找电话号码, 另一个根据电话号码查找姓名
第 17 章	
BinaryTree	演示如何使用泛型生成类型安全的结构, 可包含任何类型的元素
BuildTree	演示如何使用泛型实现类型安全的方法, 可获取任何类型的参数
第 18 章	
Cards	升级第 10 章的代码, 演示如何用集合建模一手牌
第 19 章	
BinaryTree	演示如何实现泛型 <code>IEnumerator&lt;T&gt;</code> 接口, 为泛型 <code>Tree</code> 类创建枚举器
IteratorBinaryTree	用迭代器为泛型 <code>Tree</code> 类生成枚举器

续表

项目/解决方案	说明
第 20 章	
Delegates	演示如何通过委托调用方法，将方法的逻辑和调用方法的应用程序分开。然后对项目进行扩展，演示如何用事件提醒对象发生了某事，以及如何捕捉事件并执行必要的处理
第 21 章	
QueryBinaryTree	演示如何通过 LINQ 查询从二叉树对象获取数据
第 22 章	
ComplexNumbers	定义新类型来建模复数，并为这种类型实现常用的操作符
第 23 章	
GraphDemo	生成并在 UWP 窗体上显示复杂图表。用单线程执行计算
Parallel GraphDemo	使用 Parallel 类对创建和管理任务的过程进行抽象
GraphDemo With Cancellation	中途得体地取消任务
ParallelLoop	演示何时不该使用 Parallel 类创建和运行任务
第 24 章	
GraphDemo	修改第 23 章的同名项目，使用 async 关键字和 await 操作符异步计算图表数据
PLINQ	使用并行任务，用 PLINQ 查询数据
CalculatePI	使用统计学采样计算 PI 的近似值。使用了并行任务
第 25 章	
Customers	实现能自动适应不同屏幕分辨率和设备大小的 UI。UI 使用 XAML 样式更改字体和背景图片
第 26 章	
DataBinding	修改上一章的 Customers 项目，使用数据绑定在 UI 中显示从数据源获取的客户资料；还演示了如何实现 INotifyPropertyChanged 接口，从而允许 UI 更新客户资料，并将改动发送回数据源
ViewModel	这个版本的 Customers 项目通过实现 Model-View-ViewModel 模式，将 UI 同数据源访问逻辑分开
Cortana	将 Customers 应用和 Cortana 集成。用户可发出语音指令来搜索客户
第 27 章	
Web Service	该解决方案包含一个 Web 应用程序来提供 ASP.NET Web API Web 服务，使 Customers 应用能从 SQL Server 数据库获取客户数据。Web 服务通过由实体框架创建的实体模型来访问数据库

# 简 明 目 录

## 第 I 部分 Visual C#和 Visual Studio 2015 概述

第 1 章 欢迎进入 C#编程世界.....	3
第 2 章 使用变量、操作符和表达式.....	27
第 3 章 方法和作用域.....	48
第 4 章 使用判断语句.....	72
第 5 章 使用复合赋值和循环语句.....	89
第 6 章 管理错误和异常.....	105

## 第 II 部分 理解 C#对象模型

第 7 章 创建并管理类和对象.....	127
第 8 章 理解值和引用.....	146
第 9 章 使用枚举和结构创建值类型.....	167
第 10 章 使用数组.....	185
第 11 章 理解参数数组.....	204
第 12 章 使用继承.....	215
第 13 章 创建接口和定义抽象类.....	236
第 14 章 使用垃圾回收和资源管理.....	260

## 第 III 部分 用 C#定义可扩展类型

第 15 章 实现属性以访问字段.....	281
第 16 章 使用索引器.....	302
第 17 章 泛型概述.....	317
第 18 章 使用集合.....	344
第 19 章 枚举集合.....	366

第 20 章	分离应用程序逻辑并处理事件.....	379
第 21 章	使用查询表达式来查询内存中的数据.....	406
第 22 章	操作符重载 .....	428

### 第 IV 部分 用 C#构建 UWP 应用

第 23 章	使用任务提高吞吐量 .....	451
第 24 章	通过异步操作提高响应速度 .....	489
第 25 章	实现 UWP 应用的用户界面 .....	525
第 26 章	在 UWP 应用中显示和搜索数据 .....	567
第 27 章	在 UWP 应用中访问远程数据库 .....	608

# 目 录

## 第 I 部分 Visual C#和 Visual Studio 2015 概述

<b>第 1 章 欢迎进入 C#编程世界</b> ..... 3	<b>第 3 章 方法和作用域</b> ..... 48
1.1 开始在 Visual Studio 2015 环境中编程..... 3	3.1 创建方法..... 48
1.2 写第一个程序..... 7	3.1.1 声明方法..... 48
1.3 使用命名空间..... 11	3.1.2 从方法返回数据..... 49
1.4 创建图形应用程序..... 14	3.1.3 使用表达式主体方法..... 50
1.4.1 探索通用 Windows 平台应用程序..... 20	3.1.4 调用方法..... 51
1.4.2 向图形应用程序添加代码..... 23	3.2 使用作用域..... 53
小结..... 25	3.2.1 定义局部作用域..... 54
第 1 章快速参考..... 25	3.2.2 定义类作用域..... 54
<b>第 2 章 使用变量、操作符和表达式</b> ..... 27	3.2.3 重载方法..... 55
2.1 理解语句..... 27	3.3 编写方法..... 56
2.2 使用标识符..... 27	3.4 使用可选参数和具名参数..... 63
2.3 使用变量..... 29	3.4.1 定义可选参数..... 64
2.3.1 命名变量..... 29	3.4.2 传递具名参数..... 64
2.3.2 声明变量..... 29	3.4.3 消除可选参数和具名参数的歧义..... 65
2.4 使用基元数据类型..... 30	小结..... 69
2.4.1 未赋值的局部变量..... 31	第 3 章快速参考..... 70
2.4.2 显示基元数据类型的值..... 31	<b>第 4 章 使用判断语句</b> ..... 72
2.5 使用算术操作符..... 36	4.1 声明布尔变量..... 72
2.5.1 操作符和类型..... 36	4.2 使用布尔操作符..... 72
2.5.2 深入了解算术操作符..... 38	4.2.1 理解相等和关系操作符..... 73
2.5.3 控制优先级..... 42	4.2.2 理解条件逻辑操作符..... 73
2.5.4 使用结合性对表达式进行求值..... 43	4.2.3 短路求值..... 74
2.5.5 结合性和赋值操作符..... 43	4.2.4 操作符的优先级和结合性总结..... 74
2.6 变量递增和递减..... 44	4.3 使用 if 语句做出判断..... 75
2.7 声明隐式类型的局部变量..... 45	4.3.1 理解 if 语句的语法..... 75
小结..... 46	4.3.2 使用代码块分组语句..... 76
第 2 章快速参考..... 47	4.3.3 嵌套 if 语句..... 77
	4.4 使用 switch 语句..... 82
	4.4.1 理解 switch 语句的语法..... 83



4.4.2 遵守 switch 语句的规则.....	84	6.2 尝试执行代码和捕捉异常.....	105
小结.....	87	6.2.1 未处理的异常.....	107
第4章快速参考.....	87	6.2.2 使用多个 catch 处理程序.....	108
<b>第5章 使用复合赋值和循环语句.....</b>	<b>89</b>	6.2.3 捕捉多个异常.....	108
5.1 使用复合赋值操作符.....	89	6.2.4 传播异常.....	113
5.2 使用 while 语句.....	90	6.3 使用 checked 和 unchecked 整数	
5.3 编写 for 语句.....	94	运算.....	115
5.4 编写 do 语句.....	96	6.3.1 编写 checked 语句.....	115
小结.....	103	6.3.2 编写 checked 表达式.....	116
第5章快速参考.....	104	6.4 抛出异常.....	118
<b>第6章 管理错误和异常.....</b>	<b>105</b>	6.5 使用 finally 块.....	122
6.1 处理错误.....	105	小结.....	123
		第6章快速参考.....	124

## 第 II 部分 理解 C#对象模型

<b>第7章 创建并管理类 and 对象.....</b>	<b>127</b>	8.3.1 创建 ref 参数.....	155
7.1 理解分类.....	127	8.3.2 创建 out 参数.....	156
7.2 封装的目的.....	127	8.4 计算机内存的组织方式.....	158
7.3 定义并使用类.....	128	8.5 System.Object 类.....	160
7.4 控制可访问性.....	129	8.6 装箱.....	160
7.4.1 使用构造器.....	130	8.7 拆箱.....	161
7.4.2 重载构造器.....	131	8.8 数据的安全转型.....	162
7.5 理解静态方法和数据.....	138	8.8.1 is 操作符.....	163
7.5.1 创建共享字段.....	139	8.8.2 as 操作符.....	163
7.5.2 使用 const 关键字创建静态		小结.....	165
字段.....	140	第8章快速参考.....	165
7.5.3 理解静态类.....	140	<b>第9章 使用枚举和结构创建值类型.....</b>	<b>167</b>
7.5.4 静态 using 语句.....	140	9.1 使用枚举.....	167
7.5.5 匿名类.....	143	9.1.1 声明枚举.....	167
小结.....	144	9.1.2 使用枚举.....	167
第7章快速参考.....	144	9.1.3 选择枚举字面值.....	168
<b>第8章 理解值和引用.....</b>	<b>146</b>	9.1.4 选择枚举的基础类型.....	169
8.1 复制值类型的变量和类.....	146	9.2 使用结构.....	171
8.2 理解 null 值和可空类型.....	151	9.2.1 声明结构.....	173
8.2.1 使用可空类型.....	153	9.2.2 理解结构和类的区别.....	174
8.2.2 理解可空类型的属性.....	154	9.2.3 声明结构变量.....	175
8.3 使用 ref 和 out 参数.....	154	9.2.4 理解结构的初始化.....	175

9.2.5 复制结构变量 .....	179	12.2.6 声明重写方法 .....	222
小结 .....	183	12.2.7 理解受保护的访问 .....	225
第 9 章快速参考 .....	183	12.3 理解扩展方法 .....	230
<b>第 10 章 使用数组</b> .....	<b>185</b>	小结 .....	234
10.1 声明和创建数组 .....	185	第 12 章快速参考 .....	234
10.1.1 声明数组变量 .....	185	<b>第 13 章 创建接口和定义抽象类</b> .....	<b>236</b>
10.1.2 创建数组实例 .....	186	13.1 理解接口 .....	236
10.1.3 填充和使用数组 .....	186	13.1.1 定义接口 .....	237
10.1.4 创建隐式类型的数组 .....	187	13.1.2 实现接口 .....	237
10.1.5 访问单独的数组元素 .....	188	13.1.3 通过接口来引用类 .....	239
10.1.6 遍历数组 .....	189	13.1.4 使用多个接口 .....	239
10.1.7 数组作为方法参数和返回值 传递 .....	190	13.1.5 显式实现接口 .....	240
10.1.8 复制数组 .....	191	13.1.6 接口的限制 .....	242
10.1.9 使用多维数组 .....	192	13.1.7 定义和使用接口 .....	242
10.1.10 创建交错数组 .....	193	13.2 抽象类 .....	250
小结 .....	202	13.3 密封类 .....	252
第 10 章快速参考 .....	202	13.3.1 密封方法 .....	252
<b>第 11 章 理解参数数组</b> .....	<b>204</b>	13.3.2 实现并使用抽象类 .....	253
11.1 回顾重载 .....	204	小结 .....	258
11.2 使用数组参数 .....	205	第 13 章快速参考 .....	259
11.2.1 声明参数数组 .....	206	<b>第 14 章 使用垃圾回收和资源管理</b> .....	<b>260</b>
11.2.2 使用 params object[] .....	208	14.1 对象的生存期 .....	260
11.2.3 使用参数数组 .....	209	14.1.1 编写析构器 .....	261
11.3 比较参数数组和可选参数 .....	211	14.1.2 为什么要使用垃圾回收器 .....	262
小结 .....	213	14.1.3 垃圾回收器的工作原理 .....	264
第 11 章快速参考 .....	214	14.1.4 慎用析构器 .....	264
<b>第 12 章 使用继承</b> .....	<b>215</b>	14.2 资源管理 .....	265
12.1 什么是继承 .....	215	14.2.1 资源清理方法 .....	265
12.2 使用继承 .....	216	14.2.2 异常安全的资源清理 .....	266
12.2.1 复习 System.Object 类 .....	217	14.2.3 using 语句和 IDisposable 接口 .....	266
12.2.2 调用基类构造器 .....	218	14.2.4 从析构器中调用 Dispose 方法 .....	268
12.2.3 类的赋值 .....	219	14.3 实现异常安全的资源清理 .....	270
12.2.4 声明新方法 .....	220	小结 .....	276
12.2.5 声明虚方法 .....	221	第 14 章快速参考 .....	277

## 第 III 部分 用 C#定义可扩展类型

<b>第 15 章 实现属性以访问字段</b> .....	281	17.5.2 逆变接口 .....	340
15.1 使用方法实现封装 .....	281	小结 .....	342
15.2 什么是属性 .....	283	第 17 章快速参考 .....	342
15.2.1 使用属性 .....	285	<b>第 18 章 使用集合</b> .....	344
15.2.2 只读属性 .....	285	18.1 什么是集合类 .....	344
15.2.3 只写属性 .....	286	18.1.1 List<T>集合类 .....	345
15.2.4 属性的可访问性 .....	286	18.1.2 LinkedList<T>集合类 .....	347
15.3 理解属性的局限性 .....	287	18.1.3 Queue<T>集合类 .....	349
15.4 在接口中声明属性 .....	288	18.1.4 Stack<T>集合类 .....	350
15.5 生成自动属性 .....	294	18.1.5 Dictionary<TKey, TValue> 集合类 .....	351
15.6 用属性初始化对象 .....	296	18.1.6 SortedList<TKey, TValue> 集合类 .....	352
小结 .....	299	18.1.7 HashSet<T>集合类 .....	353
第 15 章快速参考 .....	300	18.2 使用集合初始化器 .....	355
<b>第 16 章 使用索引器</b> .....	302	18.3 Find 方法、谓词和 Lambda 表达式 .....	356
16.1 什么是索引器 .....	302	18.4 比较数组和集合 .....	359
16.1.1 不用索引器的例子 .....	302	小结 .....	363
16.1.2 同一个例子改用索引器 .....	304	第 18 章快速参考 .....	363
16.1.3 理解索引器的访问器 .....	306	<b>第 19 章 枚举集合</b> .....	366
16.1.4 对比索引器和数组 .....	306	19.1 枚举集合中的元素 .....	366
16.2 接口中的索引器 .....	308	19.1.1 手动实现枚举器 .....	367
16.3 在 Windows 应用程序中使用 索引器 .....	309	19.1.2 实现 IEnumerable 接口 .....	371
小结 .....	315	19.2 用迭代器实现枚举器 .....	373
第 16 章快速参考 .....	315	19.2.1 一个简单的迭代器 .....	373
<b>第 17 章 泛型概述</b> .....	317	19.2.2 使用迭代器为 Tree<TItem>类 定义枚举器 .....	375
17.1 object 的问题 .....	317	小结 .....	377
17.2 泛型解决方案 .....	320	第 19 章快速参考 .....	378
17.2.1 对比泛型类和常规类 .....	322	<b>第 20 章 分离应用程序逻辑并处理         事件</b> .....	379
17.2.2 泛型和约束 .....	323	20.1 理解委托 .....	379
17.3 创建泛型类 .....	323	20.1.1 .NET Framework 类库的委托 例子 .....	380
17.3.1 二叉树理论 .....	323		
17.3.2 使用泛型构造二叉树类 .....	326		
17.4 创建泛型方法 .....	334		
17.5 可变性和泛型接口 .....	337		
17.5.1 协变接口 .....	338		

20.1.2 自动化工厂的例子 .....	382	21.2.6 查询 Tree<TItem>对象中的 数据 .....	417
20.1.3 不用委托实现工厂控制 系统 .....	382	21.2.7 LINQ 和推迟求值 .....	422
20.1.4 用委托实现工厂控制 系统 .....	383	小结 .....	425
20.1.5 声明和使用委托 .....	385	第 21 章快速参考 .....	425
20.2 Lambda 表达式和委托 .....	392	<b>第 22 章 操作符重载</b> .....	428
20.3 启用事件通知 .....	393	22.1 理解操作符 .....	428
20.3.1 声明事件 .....	394	22.1.1 操作符的限制 .....	428
20.3.2 订阅事件 .....	395	22.1.2 重载的操作符 .....	429
20.3.3 取消订阅事件 .....	395	22.1.3 创建对称操作符 .....	430
20.3.4 引发事件 .....	395	22.2 理解复合赋值 .....	432
20.4 理解用户界面事件 .....	396	22.3 声明递增和递减操作符 .....	433
小结 .....	403	22.4 比较结构和类中的操作符 .....	434
第 20 章快速参考 .....	403	22.5 定义成对的操作符 .....	434
<b>第 21 章 使用查询表达式来查询内存 中的数据</b> .....	406	22.6 实现操作符 .....	435
21.1 什么是 LINQ .....	406	22.7 理解转换操作符 .....	441
21.2 在 C#应用程序中使用 LINQ .....	407	22.7.1 提供内建转换 .....	441
21.2.1 选择数据 .....	408	22.7.2 实现用户自定义的转换 操作符 .....	442
21.2.2 筛选数据 .....	411	22.7.3 再论创建对称操作符 .....	443
21.2.3 排序、分组和聚合数据 .....	411	22.7.4 添加隐式转换操作符 .....	444
21.2.4 联接数据 .....	413	小结 .....	446
21.2.5 使用查询操作符 .....	415	第 22 章快速参考 .....	446
<b>第 IV 部分 用 C#构建 UWP 应用</b>			
<b>第 23 章 使用任务提高吞吐量</b> .....	451	23.3 取消任务和处理异常 .....	473
23.1 使用并行处理执行多任务处理 .....	451	23.3.1 协作式取消的原理 .....	473
23.2 用 .NET Framework 实现多任务 处理 .....	452	23.3.2 为 Canceled 和 Faulted 任务 使用延续 .....	485
23.2.1 任务、线程和线程池 .....	453	小结 .....	485
23.2.2 创建、运行和控制任务 .....	454	第 23 章快速参考 .....	486
23.2.3 使用 Task 类实现并行处理 .....	457	<b>第 24 章 通过异步操作提高响应速度</b> .....	489
23.2.4 使用 Parallel 类对任务进行 抽象 .....	467	24.1 实现异步方法 .....	490
23.2.5 什么时候不要使用 Parallel 类 .....	471	24.1.1 定义异步方法：问题 .....	490
		24.1.2 定义异步方法：解决方案 .....	492
		24.1.3 定义返回值的异步方法 .....	497