

SAMS
**Teach
Yourself**

- 全球销量逾百万册的系列图书
- 连续十余年打造的经典品牌
- 直观、循序渐进的学习教程
- 掌握关键知识的最佳起点
- “Read Less, Do More”（精读多练）的教学理念
- 以示例引导读者完成最常见的任务

每章内容针对初学者精心设计，**1**小时轻松阅读学习，
24小时彻底掌握关键知识

每章**案例与练习题**助你轻松完成常见任务，
通过**实践**提高应用技能，巩固所学知识

C# 5.0

入门经典

[美] Scott J. Dorman 著
刘琦 袁国忠 译

C# 5.0

入门经典

[美] Scott J. Dorman 著
刘琦 袁国忠 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

C# 5.0入门经典 / (美) 多曼 (Dorman, S. J.) 著 ;
刘琦, 袁国忠 译. — 北京 : 人民邮电出版社, 2014. 4
ISBN 978-7-115-34423-6

I. ①C… II. ①多… ②刘… ③袁… III. ①C语言—
程序设计 IV. ①TP312

中国版本图书馆CIP数据核字 (2014) 第015814号

版 权 声 明

Scott J. Dorman: Sams Teach Yourself C# 5.0 in 24 Hours

ISBN: 0672336847

Copyright © 2013 by Pearson Education, Inc.

Authorized translation from the English languages edition published by Pearson Education, Inc.

All rights reserved.

本书中文简体字版由美国 **Pearson** 公司授权人民邮电出版社出版。未经出版者书面许可, 对本书任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

-
- ◆ 著 [美] Scott J.Dorman
 - 译 刘 琦 袁国忠
 - 责任编辑 傅道坤
 - 责任印制 程彦红 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 大厂聚鑫印刷有限责任公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 24.25
 - 字数: 598 千字 2014 年 4 月第 1 版
 - 印数: 1-3 500 册 2014 年 4 月河北第 1 次印刷
 - 著作权合同登记号 图字: 01-2012-8615 号
-

定价: 49.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

内容提要

C#融 C++的灵活和强大与 Visual Basic 的简单易用于一身，已成为在.NET 平台上进行开发的首选语言。本书简要而全面地介绍了C#和.NET Framework，可帮助程序员迅速进入 Visual C# 5.0 编程领域。

全书分 5 部分，共 24 章。第 1 部分介绍.NET Framework、Visual Studio、C#面向对象编程功能以及 C#类型和事件的基本知识；第 2 部分阐述基本的编程知识，包括循环、字符串、正则表达式、数组和集合以及异常、泛型、匿名函数（Lambda）和查询表达式（LINQ）；第 3 部分演示如何与文件系统交互、创建和查询 XML 文档以及使用数据库；第 4 部分阐述 WPF 应用程序、数据绑定和验证以及 Windows 应用商店应用和 Web 应用程序；第 5 部分简要地介绍了特性编程、动态类型、内存组织、垃圾收集、线程、任务并行库（TPL）、并行 LINQ（PLINQ）等。

本书是为刚涉足.NET 的程序员以及想学习 C#的.NET 程序员编写的，如果您刚开始学习编程，本书也会有所帮助。

序

千年之交，在微软 41 号楼二层的一间小型会议室，一个设计小组开发了一种全新的编程语言——C#。这些设计师希望这种语言具有如下特点：易于理解又不过于简化；C++和 Java 程序员熟悉但不完全抄袭；安全但限制性不过强；实用但不完全放弃设计原则。

经过长时间的思考、设计、开发、测试和文档编写，他们向公众发布了 C# 1.0。这是一种简单易懂的面向对象语言，对其设计的很多细节都做了仔细考量，确保对象可组织成独立的组件，但该语言的基本概念都是从 20 世纪 70 年代前的面向对象语言和过程性语言借鉴而来的。

还是在那间会议室，设计小组每周相聚三次，在 C# 1.0 打下的坚实基础上继续工作。通过与街对面的 Microsoft Research Cambridge 和 CLR 小组通力合作，拓展了类型系统，让泛型类型和泛型方法支持参数化多态。他们增添了迭代器块，让创建集合更容易。泛型和生成器最初出现在 20 世纪 70 年代和 80 年代的 CLU 和 Ada 等语言中。他们还增添了匿名方法，而在现有方法中嵌入匿名方法的理念可追溯到 20 世纪 50 年代的计算机科学。

相对于其前任，C# 2.0 有巨大进步，但设计小组并不满足。他们还是每周相聚三次，但考虑的是更加基础性的问题。设计小组认识到，为操作数据，程序员需要以复杂方式组合使用相对简单的操作，这些操作通常包括对数据集合进行排序、筛选、分组、连接和投影。句法模式 (syntactic pattern) 准确地描述了这些操作，这种概念可追溯到 Haskell 等函数式语言，但也适用于 C# 等更严格的语言。LINQ (语言集成查询) 应运而生。

设计小组每周都在最初的那间会议室相聚 6 小时，这项活动持续了 10 年。随后，为与不在现场的小组成员召开电话会议，会议地点移到了五层。回顾 10 年来的历程，设计小组找出了 C# 仍没有很好解决的问题、不尽如人意的地方。鉴于对与现代动态语言和遗留对象模型互操作的需求日益增加，开发小组在 C# 4.0 中引入了一些新功能，如动态类型。另外，他们还让一些泛型接口支持类型安全的协变。

可设计小组还不满意，他们始终关注开发人员，看看能否提供更卓越的语言工具，以解决开发人员面临的烦恼。他们发现，在很多现代应用程序开发人员看来，延迟都是一个棘手的问题。当今的世界，多核计算机通过网络相连，必须立即处理收到的请求，1 毫秒也不能

等待。然而，在高延迟环境中，要编写清晰、易于理解且效果良好的程序很难。有鉴于此，C# 5.0 增添了异步方法，让程序能够在等待高延迟结果的同时做些其他有益的工作。

在这里简要地回顾一下 C# 的发展历程是个不错的主意，但本书显然没有按编年史的方式介绍 C#，这很好！编写入门书籍时，作者通常按读者的学习顺序组织材料。在大多数情况下，这与功能被引入到语言中的顺序并不相同。对于本书，我特别欣赏的一点是，Scott 按合乎逻辑的方式介绍了各种概念，从最基本的算术运算到非常复杂的概念。另外，他选择的示例很真实，能激发读者去思考，同时又足够简单，只需几个段落就能说清楚。

这里重点介绍了 C# 的发展历程，但语言的发展历程远不是全部。语言只是一个工具，让您能够利用运行库和框架库的强大功能，而运行库和框架库本身就是复杂而庞大的主题。对于本书，我特别欣赏的另一点是，它不仅关注 C# 语言本身，还在介绍 C# 概念后，阐述如何使用基类库中常用的类型提供的强大功能。

从前面的 C# 简史可知，即使只考虑 C# 语言本身，需要学习的东西也很多。我使用 C# 有十多年，还是 C# 设计小组的一员，可每天仍能学到有关 C# 的新知识和新的编程技巧，但愿本书能带领您踏上持续 10 年的编程和学习之旅。至于 C# 设计小组，我们依然在努力找出未来的发展方向，但愿能够成功！

Eric Lippert

2012 年 9 月于华盛顿西雅图

关于作者

Scott J. Dorman 是 Microsoft C# MVP、演讲人、博主、网站 WP Requests 和 WinStore Request 的创始人。他打记事起就一直与计算机打交道，于 1993 年开始从事计算机方面的工作，并于 2001 年开始使用 .NET 和 C#。当前，他正致力于使用 Microsoft .NET 技术开发商业软件，负责一个关注软件架构的用户组，在各种会议上（包括 Microsoft TechEd 和社区资助的 Code Camp）发表演讲，定期在 The Code Project 和 StackOverflow 等在线社区发表文章，还就 .NET Framework 和 C# 技术撰写博客，地址为 <http://geekswithblogs.com/sdorman>。

献辞

谨将本书献给 Nathan，我希望他能追随我的脚步，有朝一日也能著书立说。谢谢你给了我独特的视角，并用你的童真来向我展示这个美好的世界。

致谢

本书上一版的交稿仿佛发生在昨天，但实际上已经过去两年了。在决定写作本书时，我没有对即将遭遇的困难做好准备。即使本书只是重点关注 .NET Framework 和 C# 编程语言的更新，我依然审查了上一版本的所有内容和评论，旨在寻找哪里可以进行改进和简化。一如既往，我虽然是本书内容的负责人，可如果没有他人的帮助和支持，单凭一己之力也无法胜任。首先，我要感谢 Brook，感谢他给我灌输了写作本书第一版的想法；感谢 Neil，感谢他让我接着写作本书的当前版本。感谢 Sams Publishing 的其他同仁，没有他们的帮助，本书不可能出版，与你们共事的感觉很棒。还要感谢我的技术编辑 Christopher。当然，如果没有整个 C#.NET Framework 和 Visual Studio 产品团队的贡献，我压根没有东西可写。

本书是为开发社区而写的，它们给了我太多帮助。没有它们的鼓励和支持，我根本坚持不下来。开发社区包含与 Microsoft MVP 项目相关的所有人，以及 Microsoft 领域布道者，尤其是 Joe “devfish” Healy、Jeff Barnes 和 Russ “ToolShed” Fustino。

最后，我还要感谢 Nathan，在我花费无数个不眠之夜和周末来写作本书时，他表现出了理解与宽容。尽管他现在还很小，不能理解我为什么在计算机上要花费这么多时间，而不是陪他玩耍，希望等他年龄大一些后能够理解。在我写作本书第一版时，就已经让他与计算机相识，尽管他那时年龄尚幼，但当他盯着我写作时，就已经培养了他的求知欲，并对他周围的技术世界有了更深的理解和掌握。无论如何，我知道，当我写作完本书，并且开始陪他游戏玩乐时，他将会很开心。

前 言

1998 年 12 月底, Microsoft 着手推出一种全新的开发平台, 旨在提供一种创建和运行下一代应用程序和 Web 服务的全新方式。这种新平台名为 .NET Framework, 于 2000 年 6 月正式发布。

.NET Framework 将原有 Windows 接口和服务融合到单个应用程序编程接口 (API) 中, 并将众多新出现的行业标准 (如简单对象访问协议[SOAP]) 和原有的 Microsoft 技术 (如 Microsoft 组件对象模型[COM 和 COM+]和 Active Server Pages[ASP]) 加入其中。除提供一致的开发体验外, .NET Framework 还提供了最大的类库之一, 让开发人员能够重点关注应用程序逻辑而不是常见的编程任务。最后, 应用程序可以在托管运行环境中运行, 而这种环境自动处理内存分配并提供了沙箱 (限制访问), 这减少甚至消除了众多常见的编程错误。

10 多年的时间, .NET Framework 在不断发展, 它缓解了开发人员每天都能感觉到的痛点, 以易于使用的方式支持新的技术和行业标准。最新的 .NET Framework 为使用异步方法提供了非并行支持, 让您能够轻松地创建 Windows 应用商店 (Windows Store) 应用。Visual Studio 开发环境和相关工具在不断改进, 让开发人员的工作越来越轻松。

2008 年举行的 Microsoft 专业开发人员大会 (PDC) 的主题之一是 “让简单的事情容易, 让艰难的事情可能”。.NET Framework 第 1 版秉承了这种宗旨, 而每个新版本都在不断向这个目标迈进。

C# 编程语言是 Anders Hejlsberg、Scott Wiltamuth 和 Peter Golde 为 .NET Framework 开发的, 于 2000 年 7 月首次发布。鉴于 C# 是专门为 .NET Framework 开发的, 很多人将其视为 .NET Framework 的御用语言。作为编程语言, C# 借鉴了 Delphi 5、C++ 和 Java 2 的语法和主要功能, 它是一种类型安全的面向对象通用语言, 可用于编写任何类型的应用程序。随着 .NET Framework 不断发展, C# 也在不断发展, 以适应 .NET Framework 的新变化并引入新功能, 从而让简单的事情容易, 让艰难的事情可能。

虽然 .NET Framework 支持的语言有 50 多种, 但 C# 始终是最受欢迎的现代通用语言之一。

针对的读者及组织结构

本书是为刚涉足.NET 的程序员和想学习 C#的.NET 程序员编写的。如果您刚开始学习编程，本书也会有所帮助，但它并非编程入门书。本书旨在让读者尽快学会如何使用 C#完成编程任务，进而提高编程效率；它全面概述了 C#语言，重点是其功能如何帮助您解决问题，旨在让您对 C#有最全面的认识。

本书分 5 部分，每个部分都关注 C#的不同方面：从简单的基本知识到高级主题，因此建议按顺序阅读。

- 第 1 部分，“C#基础”，介绍.NET Framework、C#的面向对象编程功能以及 C#类型系统和事件的基本知识；还将简要地介绍 Visual Studio，在此过程中，您将编写一个简单的应用程序，并学习应用程序调试基本知识。
- 第 2 部分，“C#编程”，阐述基本的编程知识。您将学习如何执行循环以及如何使用字符串、正则表达式和集合；接下来，将转向更高级的主题，如异常管理和泛型；最后，将介绍匿名函数（Lambda）、查询表达式（LINQ）以及如何与动态语言交互。
- 第 3 部分，“使用数据”，演示如何与文件系统和流交互、创建和查询 XML 文档以及操作数据库。
- 第 4 部分，“使用 Visual Studio 创建应用程序”，首先创建一个使用数据绑定和数据验证的 Windows 桌面应用程序；然后介绍如何使用异步模式创建 Windows 应用商店应用，包括异步编程；最后阐述如何创建 Web 应用程序。
- 第 5 部分，“深入探索”，首先简要地介绍特性编程、动态类型和语言互操作性等高级概念；接下来，您将学习.NET Framework 如何组织内存、垃圾收集器的工作原理、.NET Framework 提供的确定性终结机制等基本知识；最后，您将学习如何使用多个线程和并行处理。

本书提供了大量的示例，旨在演示现实世界的问题以及如何使用 C#和.NET Framework 解决问题。在第 4 部分，还利用前 3 部分介绍的技能，从头开始创建一些完整的应用程序。

本书使用的约定

本书使用了一些设计元素和约定，让读者能够充分利用和参考书中的信息。

**By the
Way**

注意：提供有用的信息，读者可以马上阅读，也可阅读完当前话题后再阅读。

**Watch
Out!**

警告：提醒读者注意可能在特定情况下发生的问题或副作用。

**Did you
know?**

提示：突出那些可让编程更有效的信息。

目 录

第 1 部分 C#基础

第 1 章 .NET Framework 和 C# 3

1.1 .NET Framework 3

1.1.1 公共语言运行时 4

1.1.2 Framework 类库 6

1.1.3 并行计算平台 8

1.1.4 动态语言运行时 8

1.2 C#语言 9

1.2.1 类型 10

1.2.2 语句和表达式 10

1.2.3 变量和常量 11

1.2.4 标识符和关键字 13

1.3 总结 14

1.4 问与答 15

1.5 作业 15

1.5.1 测验 15

1.5.2 答案 16

1.5.3 练习 17

第 2 章 Visual Studio 简介 18

2.1 Visual Studio 版本简介 19

2.2 编写第一个程序 20

2.2.1 新建项目 21

2.2.2 添加程序语句 22

2.2.3 运行应用程序 23

2.3 在 Visual Studio 中调试 24

2.3.1 编译错误和运行错误 24

2.3.2 调试代码 25

2.3.3 控制程序执行 29

2.4 可视化数据 31

2.5 注释代码 32

2.6 总结 34

2.7 问与答 34

2.8 作业 34

2.8.1 测验 34

2.8.2 答案 35

2.8.3 练习 35

第 3 章 理解 C#类型 36

3.1 类型概述 36

3.2 预定义类型 37

3.3 其他常用类型 40

3.3.1 日期和时间 40

3.3.2 全局唯一标识符 (GUID) 42

3.3.3 统一资源标识符 (URI) 42

3.3.4 大型整数和复数 43

3.4 使用运算符 44

3.4.1 算术运算符和赋值 运算符 45

3.4.2 关系运算符	46	5.6.2 答案	92
3.4.3 逻辑运算符	47	5.6.3 练习	92
3.4.4 条件运算符	48		
3.5 默认值	48	第 6 章 创建枚举类型和结构	93
3.6 null 和可以为 null 的类型	49	6.1 枚举类型	93
3.7 强制转换与转换	51	6.2 使用结构	98
3.8 总结	53	6.2.1 定义结构的方法	99
3.9 问与答	53	6.2.2 构造和初始化	102
3.10 作业	54	6.3 总结	104
3.10.1 测验	54	6.4 问与答	104
3.10.2 答案	54	6.5 作业	104
3.10.3 练习	55	6.5.1 测验	104
		6.5.2 答案	104
第 4 章 理解 C#类和对象	56	6.5.3 练习	105
4.1 面向对象编程	56		
4.2 面向组件编程	57	第 7 章 事件及其处理	106
4.3 C#类	57	7.1 理解事件	106
4.4 作用域和声明空间	58	7.2 订阅和取消订阅	107
4.4.1 访问性	59	7.3 发布事件	110
4.4.2 字段和常量	60	7.4 引发事件	112
4.4.3 属性	61	7.5 总结	113
4.4.4 方法	64	7.6 问与答	114
4.4.5 实例化类	70	7.7 作业	114
4.5 嵌套类	73	7.7.1 测验	114
4.6 分部类	73	7.7.2 答案	114
4.7 静态类	73	7.7.3 练习	115
4.8 对象初始值设定项	75		
4.9 总结	75		
4.10 问与答	76	第 2 部分 C#编程	
4.11 作业	76	第 8 章 控制程序流程	119
4.11.1 测验	76	8.1 理解选择语句	119
4.11.2 答案	77	8.1.1 if 语句	119
4.11.3 练习	77	8.1.2 switch 语句	121
第 5 章 继承、接口和抽象类	78	8.2 迭代语句	124
5.1 继承和多态	78	8.2.1 while 语句	124
5.2 抽象类和抽象成员	86	8.2.2 do 语句	125
5.3 使用接口	87	8.2.3 for 语句	126
5.4 总结	90	8.2.4 foreach 语句	127
5.5 问与答	91	8.3 跳转语句	128
5.6 作业	91	8.3.1 break 语句	129
5.6.1 测验	91	8.3.2 continue 语句	130
		8.3.3 return 语句	131

8.4	总结	131	10.5	集合接口	171
8.5	问与答	131	10.6	可枚举的对象和迭代器	172
8.6	作业	131	10.7	总结	174
8.6.1	测验	131	10.8	问与答	174
8.6.2	答案	132	10.9	作业	175
8.6.3	练习	133	10.9.1	测验	175
第 9 章	使用字符串和正则表达式	134	10.9.2	答案	176
9.1	字符串	134	10.9.3	练习	176
9.1.1	空字符串	136	第 11 章	使用异常处理错误	177
9.1.2	字符串操作	137	11.1	理解异常	177
9.2	使用 StringBuilder 创建 可变字符串	141	11.1.1	System.Exception	178
9.3	类型格式化	143	11.1.2	使用标准异常	178
9.3.1	标准格式字符串	143	11.2	引发异常	179
9.3.2	自定义格式字符串	145	11.3	处理异常	180
9.3.3	复合格式化	147	11.4	重新引发捕获的异常	186
9.4	正则表达式	148	11.5	溢出和整型算术运算	187
9.4.1	C#中的正则表达式类	149	11.6	异常、代码契约和 参数验证	188
9.4.2	使用正则表达式验证 字符串	150	11.7	总结	192
9.4.3	使用正则表达式 搜索子串	150	11.8	问与答	192
9.5	总结	150	11.9	作业	193
9.6	问与答	151	11.9.1	测验	193
9.7	作业	151	11.9.2	答案	193
9.7.1	测验	151	11.9.3	练习	193
9.7.2	答案	152	第 12 章	理解泛型	195
9.7.3	练习	153	12.1	为何应使用泛型	195
第 10 章	使用数组和集合	154	12.1.1	理解泛型类型参数	199
10.1	一维数组和多维数组	154	12.1.2	泛型类型的默认值	200
10.1.1	数组索引	156	12.2	使用泛型方法	201
10.1.2	数组初始化	157	12.3	创建泛型类	202
10.1.3	System.Array 类	158	12.4	结合使用泛型和数组	204
10.2	索引器	158	12.5	泛型接口的可变性	205
10.3	泛型集合	160	12.6	使用元组	207
10.3.1	列表	160	12.7	总结	208
10.3.2	集合	162	12.8	问与答	209
10.3.3	字典	164	12.9	作业	209
10.3.4	集 (set)	166	12.9.1	测验	209
10.3.5	栈和队列	169	12.9.2	答案	209
10.4	集合初始值设定项	170	12.9.3	练习	210

第 13 章 理解查询表达式	211
13.1 LINQ 简介	211
13.2 使用 LINQ 操作数据	212
13.2.1 选择数据	213
13.2.2 筛选数据	215
13.2.3 对数据进行分组和排序	216
13.2.4 联接数据	218
13.2.5 数据平坦化	220
13.3 标准查询运算符方法	221
13.3.1 Lambda	222
13.3.2 预定义的委托	224
13.4 延迟执行	225
13.5 总结	225
13.6 问与答	225
13.7 作业	226
13.7.1 测验	226
13.7.2 答案	226
13.7.3 练习	226

第 3 部分 使用数据

第 14 章 使用文件和流	229
14.1 文件和目录	229
14.1.1 使用路径	229
14.1.2 与 Windows 特殊目录交互	230
14.1.3 DirectoryInfo 和 FileInfo 类	231
14.2 读写数据	237
14.2.1 二进制文件	238
14.2.2 缓冲流	239
14.2.3 文本文件	240
14.2.4 使用 File 类读写数据	241
14.3 总结	243
14.4 问与答	243
14.5 作业	243
14.5.1 测验	243
14.5.2 答案	244
14.5.3 练习	244

第 15 章 使用 XML	245
15.1 理解 XML DOM	246

15.2 使用 LINQ to XML	246
15.2.1 XDocument	247
15.2.2 XElement 和 XAttribute	247
15.2.3 XName 和 XNamespace	250
15.3 选择和查询 XML	252
15.4 修改 XML	255
15.5 总结	257
15.6 问与答	257
15.7 作业	258
15.7.1 测验	258
15.7.2 答案	258
15.7.3 练习	258

第 16 章 使用数据库	259
16.1 先决条件	259
16.2 理解 ADO.NET	261
16.3 理解 LINQ to ADO.NET	265
16.3.1 使用 LINQ to DataSet	265
16.3.2 使用 LINQ to SQL	266
16.3.3 使用 LINQ to Entities	271
16.4 总结	273
16.5 问与答	273
16.6 作业	274
16.6.1 测验	274
16.6.2 答案	274
16.6.3 练习	274

第 4 部分 使用 Visual Studio 创建应用程序

第 17 章 创建 Windows 应用程序	277
17.1 理解 WPF	277
17.1.1 使用标记和代码隐藏	278
17.1.2 应用程序	280
17.1.3 理解 WPF 布局	280
17.1.4 使用图形、动画和媒体	282
17.1.5 理解文本、版式和文档	282
17.1.6 数据绑定	282
17.2 创建 WPF 应用程序	282

17.3 设置布局的样式	289	20.4 理解数据验证	328
17.4 总结	291	20.5 总结	330
17.5 问与答	291	20.6 问与答	330
17.6 作业	291	20.7 作业	330
17.6.1 测验	291	20.7.1 测验	330
17.6.2 答案	291	20.7.2 答案	330
17.6.3 练习	291	20.7.3 练习	330
第 18 章 使用数据绑定和验证	292	第 5 部分 深入探索	
18.1 理解数据绑定	292	第 21 章 特性编程	333
18.2 转换数据	298	21.1 理解特性	333
18.3 验证数据	301	21.2 使用调用方信息特性	335
18.4 使用数据模板	303	21.3 使用常见特性	337
18.5 总结	303	21.3.1 特性 Obsolete	337
18.6 问与答	304	21.3.2 特性 Conditional	338
18.7 作业	304	21.3.3 全局特性	339
18.7.1 测验	304	21.4 使用自定义特性	339
18.7.2 答案	305	21.5 在运行阶段访问特性	340
18.7.3 练习	305	21.6 总结	341
第 19 章 创建 Windows 应用商店		21.7 问与答	342
应用	306	21.8 作业	342
19.1 理解 Windows Runtime	307	21.8.1 测验	342
19.2 Windows 应用商店		21.8.2 答案	342
应用简介	307	21.8.3 练习	342
19.3 异步地工作	309	第 22 章 动态类型和语言互操作性	343
19.4 创建 Windows 应用商店		22.1 使用动态类型	343
应用	312	22.1.1 转换	345
19.5 总结	316	22.1.2 动态重载解析	345
19.6 问与答	316	22.2 理解 DLR	346
19.7 作业	316	22.3 与 COM 互操作	349
19.7.1 测验	316	22.4 使用反射进行互操作	350
19.7.2 答案	317	22.5 总结	351
19.7.3 练习	317	22.6 问与答	351
第 20 章 创建 Web 应用程序	318	22.7 作业	351
20.1 理解 Web 应用程序架构	318	22.7.1 测验	351
20.2 使用 ASP.NET	319	22.7.2 答案	352
20.3 创建 Web 应用程序	321	22.7.3 练习	352
20.3.1 创建布局	323	第 23 章 内存组织和垃圾收集	353
20.3.2 设置布局的样式	325	23.1 内存组织	353
20.3.3 理解数据绑定	326		

23.2	垃圾收集	354	24.2.2	使用通知	363
23.3	理解 IDisposable	355	24.2.3	互锁操作	364
23.4	使用释放模式	356	24.2.4	其他同步原语	364
23.5	声明和使用终结器	357	24.3	理解任务并行库	365
23.6	总结	358	24.3.1	数据并行	365
23.7	问与答	358	24.3.2	任务并行	367
23.8	作业	359	24.4	使用并行 LINQ (PLINQ)	369
23.8.1	测验	359	24.5	潜在的陷阱	370
23.8.2	答案	359	24.6	总结	370
23.8.3	练习	359	24.7	问与答	371
第 24 章	理解线程、并发和并行	360	24.8	作业	371
24.1	理解线程和线程处理	360	24.8.1	测验	371
24.2	并发和同步	361	24.8.2	答案	372
24.2.1	使用锁	362	24.8.3	练习	372