

全面论述C# 5.0的入门经典著作，附赠**160**个实例源代码及**500**分钟视频教程！  
微软（中国）有限公司技术顾问总监**夏鹏**作序！

清华

开发者书库



Proficient in C# 5.0 Programming

# 精通C# 5.0

周家安◎编著

Zhou Jia'an



清华大学出版社

开发者书库



Proficient in C# 5.0 Programming

# 精通C# 5.0

周家安◎编著

Zhōu Jiā'an

清华大学出版社

## 内 容 简 介

本书系统论述了 C# 5.0 的开发方法、语法知识、程序结构、编程实例及编程技巧。全书共分 20 章,详细介绍了如下内容: C# 与 .NET 基础知识, Visual Studio 开发环境的使用, C# 语法基础与应用程序的一般结构, 面向对象编程基础知识, 开发控制台应用程序, 数据类型转换与字符串的格式化处理, 简单的数学运算, 泛型与集合, 变体的应用、调试、纠错与单元测试, Lambda 表达式与 LINQ, 开发 Windows 窗体应用程序, WPF 基础知识与常用技术, 异步编程, 系统信息管理, 操作目录、文件、流以及序列化技术, 访问数据库, 网络编程, 反射技术入门以及互操作性。本书适合作为 C# 入门读者的自学图书, 并适合作为相关高校及培训机构开设 C# 及 .NET 技术课程的教材。

本书同时配套制作了 160 个实例源代码及 500 分钟视频教程, 有助于读者快速动手实践, 提高学习效率。

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

精通 C# 5.0/周家安编著. —北京: 清华大学出版社, 2014

(清华开发者书库)

ISBN 978-7-302-38024-5

I. ①精… II. ①周… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 219830 号

责任编辑: 盛东亮

封面设计: 李召霞

责任校对: 白 蕾

责任印制: 刘海龙

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)

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

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: 186mm×240mm 印 张: 45.25

字 数: 1013 千字

版 次: 2014 年 11 月第 1 版

印 次: 2014 年 11 月第 1 次印刷

印 数: 1~2500

定 价: 89.00 元

产品编号: 058339-01



# 序

## FOREWORD

微软公司 1975 年成立,微软的童年可谓光芒四射,BASIC 语言、Dos、Windows 3.1 等不断地惊艳当时高速发展的信息时代。在他成长到 20 岁时(也就是 1995 年),发布了 Windows 95,随后的几年,他达到一个无人可及的顶峰,那些年他几乎统治了整个 IT 界和几乎每个人的生活。又过了 19 年之后,2014 年他迎来了新的掌门人——纳德拉(Satya Nadella),面对世界的新技术、新公司、新生活方式的挑战,感受着来自各方面的压力,他为公司提出了全新的策略,简言之就是“移动为先,云为先”。他同时指出:“我坚信,在未来十年,计算将无处不在,智能将触手可及。软件的进化与新式硬件的普及会在其中起到媒介作用,目前我们在工作和生活中从事和体验的很多内容都将实现数字化,甚至整个世界也是如此。可联网设备的数量快速增长、云环境所能提供的海量计算资源,大数据的洞察力,以及机器学习所获得的智能,诸多因素让这一切变为可能。”

接近不惑之年的微软,正在不断地调整以改变自己。从内部人员到产品线,进而到产品设计理念。现在,微软的产品线不仅软件产品异常丰富,而且在硬件领域不断出击,从常用的键盘、鼠标到家用游戏机 Xbox、业界最好的体感设备 Kinect 及随后推出的 Surface RT/Surface Pro。2014 年,微软更是完成了对著名移动厂商 Nokia 的收购,从而使公司变成了“软硬”兼备的公司。微软目前拥有数十个著名的产品品牌、数百个优秀的产品、数以千计的先进技术、数万名业界著名人才、数百万个行业技术解决方案以及数百亿美金的现金储备,这些资源在一个敢于面对变革的新 CEO 领导下,微软像一位围棋高手一样不断变换布局迎接全新的 21 世纪,这个布局的核心就是“移动为先,云为先”,换言之就是“服务+设备”。

笔者从小就是一个非常“Geek”的人,从装收音机、电视机到给科技杂志投稿,整天畅想着如科幻小说般的未来,这一切伴随着我的少年时代。后来逐步学习各种计算机语言和各種 IT 技术,希望自己能够修炼成 IT 界的“绝世高手”。但是我天赋平凡,面对发展迅猛的 IT 产业,我依然像个无知的孩子,只有不断地学习新的知识。另一方面,一直以来,在我的内心深处都认为传道授业、教书育人是一件无上光荣的事情。1996 年春天,当 Windows 95 中文版在中国发布后不久,我加入了微软公司,我那时的头衔是“布道师”(Evangelist),虽然不是“老师”,但是我找到了“装老师”的感觉。从主办 TechEd、PDC(Build),到在微软研究院和最聪明的科学家一起工作……我在微软经历了人生最美好的时光。2000 年,我加入了另外一家伟大的“水果”公司……直到 2012 年,当 Windows 8.0 即将发布时,我回到了微软

公司,我的职业生涯和这家伟大的公司重新绑定,我相信我选择的未来之路!

清华大学出版社是令人敬仰的出版社,选题精准,作风严谨。小时候,它就是寻找计算机和技术“武功秘籍”的地方。随着移动互联网的飞速发展,人们的时间被无情的“碎片化”——微信、微博、短信、邮件、网页,等等;但是我认为要想在技术方面有所作为,踏踏实实地读书并积极地实践是最有效的方式。很荣幸受邀为此微软技术系列图书撰写序言,当我看到这些选题和主要内容时,我迫不及待地恳请编辑务必“赐予”我一套图书,我一定会仔细拜读,我也会推荐给我的业界好友。

北京的雾霾好像越来越严重了,而周末在一个安静的地方阅读一本好书,整个人的“小宇宙”会被提升到另一个维度,大有醍醐灌顶、大彻大悟的感觉。希望您也能和我一样在阅读这套图书时找到这样的美妙感觉……

夏 鹏

微软(中国)有限公司

# 前言

## PREFACE

C# (C Sharp, “#”读作 Sharp) 是微软公司推出的一种完全面向对象、简单易学、现代化的新型编程语言。经过几个版本的演化, C# 的语法已然变成更加丰富和成熟, 并且具有许多其他高级程序设计语言所不具备的特性, 如 LINQ 语法等。

C# 与 .NET Framework 平台高度集成, 也就是说, C# 编写的应用程序必须依赖于 .NET 框架, 并可以与如 Visual Basic .NET、Visual C++ .NET 等语言编写的组件进行交互。 .NET 框架向开发者公开庞大的 API 库, 以帮助开发人员快速构建强大的应用程序。

### 本书包含了哪些内容

本书的内容基于最新的 C# 版本 (目前为 5.0) 而展开叙述的, 主要涉及以下方面的内容:

- C# 与 .NET 简介
- 开发环境 Visual Studio 的使用方法与技巧
- C# 语法基础与应用程序的一般结构剖析
- 面向对象编程基础
- 开发控制台应用程序
- 数据类型转换与字符串的格式化处理
- 简单的数学运算
- 泛型与集合
- 变体的应用
- 调试、纠错与单元测试
- Lambda 表达式与 LINQ
- 开发 Windows 窗体应用程序
- WPF 基础知识与常用技术
- 异步编程
- 系统信息管理
- 操作对目录、文件、流以及序列化技术
- 访问数据库
- 网络编程

- 反射技术入门
- 互操作性

## 如何使用本书

本书采用通俗易懂的讲解风格,尽量避免晦涩难懂的描述,降低初学者的理解难度。同时,书中针对每个知识点都附有简单示例程序。读者在阅读本书的时候,可以先阅读有关每个知识点的说明,然后将附带的示例程序动手操作一遍。在完成书中示例代码后,读者应该运行一下应用程序,并观察程序代码的执行结果,然后思考一下为什么会得到这样的结果,不要做完示例就抛在脑后,应当学会举一反三。

当读者了解书中的示例后,也可以进行“二次创作”,即适当地将示例代码进行修改,使其变成一个新的示例。如此一来,一个示例可以变成两个、三个,甚至更多的示例,读者从中也学会了如何灵活地运用学到的知识。当然,这对于从未进行过任何程序开发的入门者而言,会有些困难。刚开始的时候,读者可以抱着尝试的心态去做,不要在乎付出的劳动是否马上得到回报,“欲速则不达”,只要坚持不懈,一点一滴地积累和总结,相信每个有志于步入编程殿堂的人都会收获颇丰。

在学习编写代码的过程中,读者不妨学会多问几个“为什么”。例如,这段代码为什么要先执行第1行,才能去执行第2行?如果把第1行代码与第2行代码调换后会发生什么?这个程序为什么要这样写?如果不这样写又会遇到哪些问题?……这种学习方法有助于初学者形成清晰的思路,养成良好的编程习惯。因为程序代码都是人写的,它是事先安排好的一系列指令,交给计算机去执行,程序代码体现了人的主观意志,所以在编写代码的时候,思路非常重要。有了好的思路,并且在满足技术条件的情况下,才能开发出优秀的应用程序。若思路不清晰,写出来的代码会非常混乱,维护起来也会十分困难。

## 本书适合我吗

本书内容侧重基础知识,适合以下读者对象:

- 希望通过自学走上编程之路却不具备基础知识的读者
- 希望从其他编程语言转向.NET方向的读者
- 对C#以及.NET相关技术感兴趣的读者
- 培训机构或高等院校学习.NET相关课程的学员

## 学习本书内容,我能做什么

C#与.NET框架紧密结合,除了可以开发常见的控制台、Windows应用程序外,还可以用来开发Web应用程序(如网站)、运行在移动平台上(如Windows Phone)的应用程序。另外,通过Mono及相关技术,还可以使用C#.NET来开发运行在Android、iOS等移动终端上的应用程序。

## 勘误与支持

由于作者水平有限,编写时间仓促,书中难免会有一些不妥之处。望读者不吝指正,可以通过以下途径与作者联系:

电子邮件: [csdev2012@foxmail.com](mailto:csdev2012@foxmail.com)

博客: <http://www.cnblogs.com/tcjiaan>

微博: <http://weibo.com/tcjiaan>

## 致谢

本书能够顺利完成,离不开亲朋好友的鼓励和支持。感谢众多网友所给予的肯定,也感谢清华大学出版社给了我一个写作的机会,感谢盛东亮编辑,在书稿的写作过程中给予了我极大的帮助和支持。

作 者

2014 年 10 月

# 目录

## CONTENTS

---

|                            |    |
|----------------------------|----|
| 第 1 章 准备工作                 | 1  |
| 1.1 为什么要编写程序               | 1  |
| 1.2 C# 与 .NET 框架的关系        | 2  |
| 1.3 开发工具——Visual Studio    | 3  |
| 1.3.1 下载和安装                | 3  |
| 1.3.2 认识 Visual Studio     | 3  |
| 1.3.3 使用快捷键                | 4  |
| 1.3.4 放置子窗口                | 4  |
| 1.3.5 自动隐藏窗口               | 5  |
| 1.3.6 快速启动                 | 6  |
| 1.3.7 个性化设置                | 7  |
| 1.4 创建第一个应用程序              | 12 |
| 1.5 解决方案与项目的关系             | 15 |
| 第 2 章 语法基础                 | 17 |
| 2.1 代码的基本结构                | 17 |
| 2.2 命名空间                   | 18 |
| 2.2.1 为什么要使用命名空间           | 18 |
| 2.2.2 如何访问命名空间中的成员         | 19 |
| 2.2.3 引入命名空间               | 19 |
| 2.3 Main 入口点               | 20 |
| 2.3.1 程序代码真的从 Main 方法开始执行吗 | 21 |
| 2.3.2 如何选择入口点              | 21 |
| 2.4 变量与常量                  | 23 |
| 2.4.1 变量                   | 23 |
| 2.4.2 变量的生命周期              | 24 |
| 2.4.3 常量                   | 24 |

|       |                |    |
|-------|----------------|----|
| 2.5   | 命名规则           | 25 |
| 2.6   | 运算符            | 26 |
| 2.6.1 | 示例：简单运算        | 26 |
| 2.6.2 | 示例：自增和自减运算     | 27 |
| 2.6.3 | 示例：位运算         | 28 |
| 2.6.4 | 示例：比较运算符       | 29 |
| 2.7   | 流程控制           | 30 |
| 2.7.1 | 顺序执行           | 31 |
| 2.7.2 | 有选择地执行         | 31 |
| 2.7.3 | 循环往复           | 34 |
| 2.8   | 写注释是个好习惯       | 39 |
| 2.9   | 基本的数据类型        | 40 |
| 2.10  | 不要错过《C#语言规范》文档 | 42 |
| 第3章   | 面向对象编程         | 44 |
| 3.1   | 类              | 44 |
| 3.1.1 | 字段             | 45 |
| 3.1.2 | 属性             | 45 |
| 3.1.3 | 方法             | 47 |
| 3.1.4 | 构造函数与析构函数      | 49 |
| 3.2   | 结构             | 53 |
| 3.3   | 引用类型与值类型       | 55 |
| 3.4   | ref 参数与 out 参数 | 57 |
| 3.5   | 方法重载           | 62 |
| 3.6   | 静态类与静态成员       | 63 |
| 3.7   | 继承与多态          | 64 |
| 3.7.1 | 可访问性           | 65 |
| 3.7.2 | 继承             | 67 |
| 3.7.3 | 注意可访问性要一致      | 69 |
| 3.7.4 | 隐藏基类的成员        | 70 |
| 3.7.5 | 覆写基类成员         | 71 |
| 3.7.6 | 如何阻止类被继承       | 73 |
| 3.8   | 抽象类            | 74 |
| 3.9   | 接口             | 77 |
| 3.9.1 | 定义接口           | 78 |
| 3.9.2 | 接口与多继承         | 78 |

|        |                        |     |
|--------|------------------------|-----|
| 3.9.3  | 实现接口 .....             | 78  |
| 3.9.4  | 显式实现接口 .....           | 80  |
| 3.10   | 扩展方法 .....             | 82  |
| 3.11   | 委托与事件 .....            | 83  |
| 3.11.1 | 定义和使用委托 .....          | 83  |
| 3.11.2 | 将方法作为参数传递 .....        | 85  |
| 3.11.3 | 使用事件 .....             | 86  |
| 3.12   | 枚举 .....               | 91  |
| 3.12.1 | 使用枚举类型 .....           | 92  |
| 3.12.2 | 如何获取枚举的值列表 .....       | 94  |
| 3.12.3 | 如何取得枚举各成员的名字 .....     | 95  |
| 3.12.4 | 枚举的位运算 .....           | 96  |
| 3.13   | 特性 .....               | 99  |
| 3.13.1 | 自定义特性 .....            | 100 |
| 3.13.2 | 如何把特性应用到方法的返回值 .....   | 101 |
| 3.13.3 | 通过反射技术检索特性 .....       | 102 |
| 3.14   | 数组 .....               | 103 |
| 3.14.1 | 定义数组的几种方法 .....        | 103 |
| 3.14.2 | 多维数组 .....             | 105 |
| 3.14.3 | 嵌套数组 .....             | 107 |
| 3.14.4 | 复制数组 .....             | 109 |
| 3.14.5 | 反转数组 .....             | 111 |
| 3.14.6 | 更改数组的大小 .....          | 111 |
| 3.14.7 | 在数组中查找元素 .....         | 112 |
| 3.14.8 | 灵活使用 ArrayList 类 ..... | 116 |
| 第 4 章  | 控制台应用程序 .....          | 118 |
| 4.1    | 输出文本信息 .....           | 118 |
| 4.2    | 获取键盘输入 .....           | 120 |
| 4.2.1  | Read 方法 .....          | 120 |
| 4.2.2  | ReadKey 方法 .....       | 121 |
| 4.2.3  | ReadLine 方法 .....      | 122 |
| 4.3    | 命令行参数 .....            | 123 |
| 4.4    | 设置控制台窗口的外观 .....       | 125 |
| 4.5    | 控制台窗口的大小和位置 .....      | 126 |

|                                      |     |
|--------------------------------------|-----|
| 第5章 字符串处理与数学运算 .....                 | 131 |
| 5.1 对象的字符串表示形式 .....                 | 131 |
| 5.1.1 重写 ToString 方法 .....           | 131 |
| 5.1.2 扩展方法 .....                     | 133 |
| 5.2 字符串的串联与拆分 .....                  | 135 |
| 5.2.1 使用+运算符来串联字符串 .....             | 135 |
| 5.2.2 使用 Join 方法拼接字符串数组 .....        | 136 |
| 5.2.3 使用 Concat 方法创建字符串 .....        | 136 |
| 5.2.4 使用 StringBuilder 类来创建字符串 ..... | 137 |
| 5.2.5 使用 Split 方法来拆分字符串 .....        | 140 |
| 5.3 转义字符 .....                       | 141 |
| 5.4 英文字母的大小写转换 .....                 | 143 |
| 5.5 Parse 和 TryParse 方法 .....        | 143 |
| 5.6 查找和修改字符串 .....                   | 145 |
| 5.6.1 查找索引 .....                     | 146 |
| 5.6.2 存在性查找 .....                    | 147 |
| 5.6.3 字符串的增删改 .....                  | 148 |
| 5.6.4 填充字符串 .....                    | 150 |
| 5.7 格式化字符串 .....                     | 151 |
| 5.7.1 IFormatProvider 接口 .....       | 151 |
| 5.7.2 区域性相关的信息 .....                 | 152 |
| 5.7.3 实现字符串的格式化 .....                | 153 |
| 5.8 字符串的复合格式化 .....                  | 160 |
| 5.9 生成随机数 .....                      | 161 |
| 5.10 Math 类与常用的数学运算 .....            | 162 |
| 5.11 与日期时间有关的计算 .....                | 163 |
| 第6章 类型转换 .....                       | 165 |
| 6.1 隐式转换 .....                       | 165 |
| 6.2 显式转换 .....                       | 167 |
| 6.3 可以为 null 的类型 .....               | 169 |
| 6.4 引用类型的兼容性转换 .....                 | 170 |
| 6.5 类型转换帮助器 .....                    | 172 |
| 6.6 自定义转换 .....                      | 175 |

|                              |     |
|------------------------------|-----|
| 第 7 章 泛型、集合与变体 .....         | 178 |
| 7.1 泛型类型 .....               | 178 |
| 7.1.1 泛型与类 .....             | 179 |
| 7.1.2 泛型与接口 .....            | 180 |
| 7.1.3 泛型与类型成员 .....          | 182 |
| 7.1.4 泛型与委托 .....            | 183 |
| 7.1.5 泛型约束 .....             | 186 |
| 7.2 集合 .....                 | 189 |
| 7.2.1 普通集合 .....             | 190 |
| 7.2.2 泛型集合 .....             | 192 |
| 7.2.3 字典 .....               | 200 |
| 7.3 自定义排序 .....              | 205 |
| 7.4 变体 .....                 | 208 |
| 7.4.1 协变与逆变 .....            | 209 |
| 7.4.2 类型参数的输入与输出 .....       | 211 |
| 7.4.3 必不可少的技巧 .....          | 214 |
| 第 8 章 纠错与单元测试 .....          | 216 |
| 8.1 实时纠正语法错误 .....           | 216 |
| 8.2 代码调试 .....               | 218 |
| 8.2.1 断点法 .....              | 218 |
| 8.2.2 输出信息法 .....            | 220 |
| 8.3 断言 .....                 | 222 |
| 8.4 使用日志文件 .....             | 225 |
| 8.5 异常处理 .....               | 229 |
| 8.6 单元测试 .....               | 232 |
| 第 9 章 匿名方法与 Lambda 表达式 ..... | 237 |
| 9.1 匿名方法 .....               | 237 |
| 9.2 Lambda 表达式 .....         | 238 |
| 9.2.1 用于赋值 .....             | 239 |
| 9.2.2 用于参数传递 .....           | 240 |
| 第 10 章 集成化查询 .....           | 245 |
| 10.1 LINQ 基本语法 .....         | 245 |

|                                    |                                 |            |
|------------------------------------|---------------------------------|------------|
| 10.1.1                             | 对查询结果类型的巧妙处理 .....              | 246        |
| 10.1.2                             | 延迟执行与强制立即执行 .....               | 248        |
| 10.2                               | 筛选 .....                        | 250        |
| 10.3                               | 排序 .....                        | 251        |
| 10.4                               | 分组 .....                        | 253        |
| 10.5                               | 更复杂的查询 .....                    | 257        |
| 10.5.1                             | 动态创建类型 .....                    | 258        |
| 10.5.2                             | 联合查询 .....                      | 259        |
| 10.5.3                             | 嵌套查询 .....                      | 265        |
| <b>第 11 章 Windows 窗体应用程序 .....</b> |                                 | <b>269</b> |
| 11.1                               | Windows 窗体应用程序的基本结构 .....       | 269        |
| 11.1.1                             | 客户区域 .....                      | 270        |
| 11.1.2                             | 控件 .....                        | 270        |
| 11.1.3                             | 应用程序项目 .....                    | 270        |
| 11.1.4                             | Windows 消息循环 .....              | 272        |
| 11.1.5                             | 演练：手动创建一个 Windows 应用程序 .....    | 275        |
| 11.2                               | ApplicationContext 类 .....      | 278        |
| 11.3                               | 窗体设计器 .....                     | 280        |
| 11.4                               | 控件的基类——Control .....            | 285        |
| 11.4.1                             | 示例：设置控件的位置和大小 .....             | 286        |
| 11.4.2                             | 示例：设置控件的前景色和背景色 .....           | 289        |
| 11.4.3                             | 示例：调整控件的 Z 顺序 .....             | 291        |
| 11.5                               | 常规控件 .....                      | 295        |
| 11.5.1                             | 共享事件处理程序 .....                  | 295        |
| 11.5.2                             | 制作“用户登录”窗口 .....                | 296        |
| 11.5.3                             | CheckBox 与 RadioButton 示例 ..... | 299        |
| 11.5.4                             | 使用 ListBox 控件呈现对象列表 .....       | 302        |
| 11.5.5                             | 自行绘制列表控件的项 .....                | 308        |
| 11.5.6                             | 组合框 .....                       | 311        |
| 11.5.7                             | TextBox 的自动完成功能 .....           | 313        |
| 11.5.8                             | 多视图列表 .....                     | 314        |
| 11.5.9                             | 图像呈现控件 .....                    | 320        |
| 11.5.10                            | 实时报告进度 .....                    | 327        |
| 11.5.11                            | 选择日期和时间 .....                   | 333        |
| 11.5.12                            | 规范用户输入 .....                    | 335        |

|               |                        |            |
|---------------|------------------------|------------|
| 11.6          | 容器控件 .....             | 340        |
| 11.6.1        | 在容器内部显示滚动条 .....       | 340        |
| 11.6.2        | 分组面板 .....             | 341        |
| 11.6.3        | 网格布局面板 .....           | 342        |
| 11.6.4        | 选项卡 .....              | 345        |
| 11.6.5        | 自动排列内容的面板 .....        | 349        |
| 11.7          | 工具栏和菜单 .....           | 351        |
| 11.7.1        | 工具栏 .....              | 351        |
| 11.7.2        | 菜单栏 .....              | 355        |
| 11.7.3        | 上下文菜单 .....            | 358        |
| 11.7.4        | 自定义承载控件 .....          | 359        |
| 11.8          | 对话框 .....              | 363        |
| 11.8.1        | “打开文件”和“保存文件”对话框 ..... | 363        |
| 11.8.2        | 字体选择对话框 .....          | 367        |
| 11.8.3        | 颜色选择对话框 .....          | 369        |
| 11.8.4        | 自定义对话框 .....           | 370        |
| 11.9          | 用户控件 .....             | 375        |
| 11.10         | 自定义控件 .....            | 376        |
| <b>第 12 章</b> | <b>WPF 应用程序 .....</b>  | <b>382</b> |
| 12.1          | WPF 应用程序项目的结构 .....    | 382        |
| 12.2          | XAML 基础 .....          | 386        |
| 12.2.1        | XAML 命名空间 .....        | 386        |
| 12.2.2        | 代码隐藏 .....             | 387        |
| 12.2.3        | 标记扩展 .....             | 387        |
| 12.2.4        | 内容属性 .....             | 389        |
| 12.3          | 依赖项属性 .....            | 391        |
| 12.3.1        | 定义和使用依赖项属性 .....       | 392        |
| 12.3.2        | 使用元数据 .....            | 395        |
| 12.3.3        | 使用已注册的依赖项属性 .....      | 400        |
| 12.3.4        | 只读依赖项属性 .....          | 403        |
| 12.3.5        | 附加属性 .....             | 405        |
| 12.4          | 路由事件 .....             | 409        |
| 12.4.1        | 路由策略 .....             | 412        |
| 12.4.2        | 注册路由事件的类处理程序 .....     | 414        |
| 12.5          | 认识 WPF 控件 .....        | 417        |

|         |                              |     |
|---------|------------------------------|-----|
| 12.5.1  | Control 类                    | 417 |
| 12.5.2  | 内容控件                         | 421 |
| 12.5.3  | 输入控件                         | 429 |
| 12.6    | 数据绑定与视图呈现                    | 431 |
| 12.6.1  | 用于数据绑定的 XAML 扩展标记            | 432 |
| 12.6.2  | 使用 InotifyPropertyChanged 接口 | 438 |
| 12.6.3  | 上下文绑定                        | 443 |
| 12.6.4  | 绑定转换器                        | 449 |
| 12.6.5  | 数据集合控件                       | 451 |
| 12.6.6  | 数据模板                         | 456 |
| 12.6.7  | 数据视图                         | 465 |
| 12.6.8  | ObservableCollection<T>集合    | 476 |
| 12.7    | 布局                           | 479 |
| 12.7.1  | Panel 类                      | 479 |
| 12.7.2  | 网格布局(Grid)                   | 481 |
| 12.7.3  | 栈布局(StackPanel)              | 485 |
| 12.7.4  | 停靠面板(DockPanel)              | 486 |
| 12.7.5  | 绝对定位(Canvas)                 | 487 |
| 12.7.6  | 自动换行(WrapPanel)              | 489 |
| 12.7.7  | Z 顺序                         | 490 |
| 12.8    | 用户控件与自定义控件                   | 491 |
| 12.9    | 样式与资源                        | 500 |
| 12.9.1  | 样式中的触发器                      | 501 |
| 12.9.2  | 资源的有效范围                      | 503 |
| 12.9.3  | 合并资源字典                       | 505 |
| 12.9.4  | 静态资源与动态资源                    | 508 |
| 12.10   | 图形                           | 509 |
| 12.10.1 | Shape 类                      | 509 |
| 12.10.2 | 基本图形                         | 514 |
| 12.11   | 动画                           | 516 |
| 12.11.1 | 演示图板                         | 516 |
| 12.11.2 | 简单动画与关键帧动画                   | 518 |
| 第 13 章  | 异步编程                         | 522 |
| 13.1    | 为什么要使用异步编程                   | 522 |
| 13.2    | 使用 Thread 类进行异步编程            | 525 |

|               |                             |            |
|---------------|-----------------------------|------------|
| 13.3          | 线程锁 .....                   | 527        |
| 13.4          | 通过委托执行异步操作 .....            | 530        |
| 13.5          | 并行任务 .....                  | 533        |
| 13.5.1        | 启动简单的并行任务 .....             | 533        |
| 13.5.2        | Task 与 Task<TResult>类 ..... | 538        |
| 13.5.3        | 创建可取消的 Task .....           | 540        |
| 13.6          | async 和 await 关键字 .....     | 542        |
| <b>第 14 章</b> | <b>系统信息管理 .....</b>         | <b>548</b> |
| 14.1          | 管理进程 .....                  | 548        |
| 14.1.1        | 管理现有进程 .....                | 548        |
| 14.1.2        | 启动新进程 .....                 | 552        |
| 14.1.3        | 重定向输入/输出流 .....             | 554        |
| 14.2          | 管理服务 .....                  | 556        |
| 14.3          | WMI 查询 .....                | 563        |
| 14.4          | 读取系统参数 .....                | 567        |
| 14.4.1        | ComputerInfo 类 .....        | 568        |
| 14.4.2        | Environment 类 .....         | 569        |
| 14.4.3        | SystemInformation 类 .....   | 570        |
| 14.4.4        | 用于 WPF 的系统信息类 .....         | 571        |
| 14.5          | 写入事件日志 .....                | 572        |
| <b>第 15 章</b> | <b>文件与流 .....</b>           | <b>575</b> |
| 15.1          | 创建和删除目录 .....               | 575        |
| 15.2          | 创建和删除文件 .....               | 577        |
| 15.3          | 流 .....                     | 578        |
| 15.3.1        | 读写文件 .....                  | 579        |
| 15.3.2        | 内存流 .....                   | 584        |
| 15.4          | 内存映射文件 .....                | 587        |
| 15.5          | 监控文件系统的变化 .....             | 590        |
| 15.6          | 独立存储 .....                  | 592        |
| 15.7          | 压缩与解压缩 .....                | 595        |
| <b>第 16 章</b> | <b>序列化与反序列化 .....</b>       | <b>600</b> |
| 16.1          | 二进制序列化 .....                | 600        |
| 16.2          | XML 序列化 .....               | 603        |