

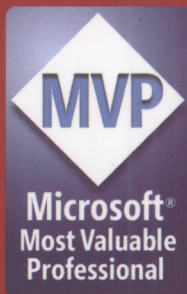
PROGRAMMER TO PROGRAMMER™



Visual Basic 2008 Programmer's Reference

Visual Basic 2008

编程参考手册



(美) Rod Stephens
徐燕华

著
译



清华大学出版社

Visual Basic 2008

编程参考手册

(美) Rod Stephens 著
徐燕华 译

清华大学出版社

北 京

Rod Stephens
Visual Basic 2008 Programmer's Reference
EISBN: 978-0-470-18262-8
Copyright © 2008 by Wiley Publishing, Inc.
All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2008-1933

本书封面贴有 Wiley 公司防伪标签, 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 Basic 2008 编程参考手册/(美)斯蒂芬斯(Stephens, R.)著; 徐燕华 译. —北京: 清华大学出版社, 2009.9

书名原文: Visual Basic 2008 Programmer's Reference

ISBN 978-7-302-20766-5

I. V… II. ①斯… ②徐… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 151706 号

责任编辑: 王 军 李 阳

装帧设计: 孔祥丰

责任校对: 成凤进

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 63.25 字 数: 1618 千字

版 次: 2009 年 9 月第 1 版 印 次: 2009 年 9 月第 1 次印刷

印 数: 1~3000

定 价: 128.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 028730-01

前 言

被誉为“最后一位全才”的艾萨克·牛顿爵士是一位颇有造诣的物理学家(他的三大运动定律是经典力学的基础,而经典力学定义天体物理学长达 3 个世纪)、数学家(他是微积分学的发明者,提出了求方程根的牛顿法)、天文学家、自然哲学家和炼金术士。他发明了反射式望远镜、色彩理论和冷却定律并研究了音速。

同样重要的是,他是在相对论、量子力学、基因序列、热力学、并行计算以及大量其他极其复杂的科学分支出现之前出生的。

如果您曾用过 Visual Basic 3,那么一定了解其一切。Visual Basic 3 是相对较小但功能强大的语言。Visual Basic 4 向该语言中添加了类,使 Visual Basic 变得更为复杂。版本 4、5、6 添加了对数据库编程和其他主题(如自定义控件)的更多支持,但 Visual Basic 仍是较容易理解的语言,如果您花些时间,就可成为其专家。

Visual Basic .NET 极大地加速了 Visual Basic 的发展。.NET Framework 向 Visual Basic 中添加了强大的新工具,但同时也加大了复杂性。相关的技术以不断增加的速度添加到 Visual Basic 语言中,因此,要让某个人成为有关 Visual Basic 的每个主题的专家是不可能的。

为了触及 Visual Basic 的各个方面,您需要深入了解数据库技术、自定义控件、自定义属性编辑器、XML、加密技术、串行化、二维/三维制图、多线程、反射、代码文档对象模型(DOM)、诊断技术、全球化、Web 服务、进程间通信、工作流、Office、ASP 以及更多内容。

本书并不试图涉及所有方面,而是提供对基本的 Visual Basic 主题的广泛而扎实的理解。介绍了使 Visual Basic 成为高效率语言的强大开发环境。讲述了 Visual Basic 语言本身并解释了如何使用它来执行大量重要的开发任务。

本书还介绍了窗体、控件和其他 Visual Basic 提供的用于在现代窗口环境中构建应用程序的对象。

本书没有涉及 Visual Basic 相关的每个主题,但介绍了开发人员构建复杂应用程序所需的大部分技术。

0.1 使用 Visual Basic 2008 的原因

至今为止,软件工程师已接触到 5 代语言。第一代语言(1GL)是机器语言:0 和 1。

第二代语言(2GL)是汇编语言,它提供了机器指令的简单记忆方法,还简单提供了编写机器代码的少量额外工具。

第三代语言(3GL)是更高级的语言,如 Pascal 和 FORTRAN。它们提供了更复杂的语言元素,如子例程、循环和数据结构。

第三代语言(4GL)是“自然语言”,如 SQL。它们让开发人员使用某种类似于人类语言的

语言来执行编程任务。例如，SQL 语句“SELECT*FROM Customers WHERE Balance>50”让数据库返回有关欠款大于\$50 的客户的信息。

第五代语言(5GL)提供了强大的、高度图形化的开发环境，允许开发人员以更复杂的方式使用底层语言。Visual Studio 开发环境极其强大。它提供了图形化编辑器，使得构建窗体和编辑属性变得非常简单；提供了 IntelliSense 功能帮助开发人员进行输入；提供了自动完成功能，使开发人员可以使用有意义的变量名称，而不必浪费时间完全靠手工输入；提供了断点、监视器和其他高级调试工具，让应用程序的构建更容易。

Visual Basic 使用了一个最为强大的开发环境——Visual Studio，但它不是使用该环境的唯一语言。C#语言也使用 Visual Studio 开发环境。因此，问题是应使用 Visual Basic 还是 C#？

Visual Basic 程序员经常会开这样的玩笑：“Visual Basic .NET 和 C#之间的区别是什么？大概 3 个月！”这个玩笑暗示 Visual Basic .NET 语法更容易理解，并且使用它构建应用程序更为快速。同样，C#程序员也有自己的关于 Visual Basic .NET 的玩笑，也是暗示 C#的功能更为强大。

实际上，Visual Basic .NET 并不比 C#更易用，而 C#也并不比 Visual Basic .NET 功能更为强大。两种语言的基本形式非常类似。除了一些风格上的区别(Visual Basic 面向代码行，而 C#则使用大量大括号和分号)外，这两种语言在很多方面都非常相似。两种语言都使用 Visual Studio 开发环境，都能访问支持类和工具的 .NET Framework，并且都提供了执行基本编程任务的类似语法。

这两种语言的主要区别在于风格。如果您曾经使用过以前版本的 Visual Basic，或许会发现 Visual Basic 2008 更易于上手。如果您曾经使用过 C++或 Java，或许会发现 C#(或者是 Visual C++或 Visual J#)更容易学习。

Visual Basic 与其他 Microsoft 产品有一些联系。例如，ASP 使用 Visual Basic 创建交互式 Web 页面。Microsoft Office 应用程序(Word、Excel 和 PowerPoint 等)和许多第三方的工具使用 Visual Basic for Application(VBA)作为宏编程语言。如果了解 Visual Basic，就可以在使用其他语言时处于优势地位。活动服务器页面(Active Server Pages, ASP)和 VBA 基于 .NET 之前的 Visual Basic 版本，虽然您不会立刻就知道如何使用它们，但是如果需要学习 ASP 或 VBA，掌握它们将会占有很大的优势。

如果您是编程新手，那么 Visual Basic 2008 和 C#都是很好的选择。笔者个人认为 Visual Basic .NET 可能更容易学习一些，但是这种观点可能稍微带有一些偏见，因为笔者在 C#出现之前一直都在使用 Visual Basic。选择哪种语言并不是大问题，如果需要的话，您可以在后面很容易地改为使用另一种语言。

0.2 本书的读者对象

本书的读者对象是所有层次的程序员。本书从头开始讲述 Visual Basic 2008 语言，因此读者不需要学习该语言的以前版本。本书也介绍了许多中、高级主题。本书对于各个主题的介绍都非常深入，即使是有经验的开发人员也会发现新的提示、技巧及语言细节。已经掌握了该语言的读者，仍然可以通过本书发现一些有用的信息，并且本书的参考附录将帮助您非

常容易地查找已经遗忘的细节。

本书中一些章节快速讲述了较为基础的内容。如果您之前从来没有进行过编程并且不了解计算机，就可能需要先阅读一些基础的书籍。如果您基本了解计算机，就可以轻松地通过本书学习 Visual Basic 2008。

如果已经使用其他任何语言进行过编程，就应该熟悉一些基础内容，如变量声明、数据类型和数组，因此应该可以毫无困难地学习本书。本书的参考附录可以很好地帮助您将已经了解的语言转换为对应的 Visual Basic 语法。

0.3 本书的结构

本书共分为 6 大部分。下面简要描述每一部分中的章节。如果您是有经验的程序员，可以参阅这些说明来决定跳过哪些章节及详细学习哪些章节。

第 I 部分 IDE

这一部分的章节从 Visual Basic 开发人员的视角介绍了 Visual Studio 开发环境(IDE)。这个 IDE 对于 C#和其他人员来说也是大体相同的，但也有一些区别，如哪个键盘快捷键执行哪个任务。

第 1 章“IDE 简介”介绍了如何从 Visual Studio 集成开发环境入手。讲述了如何为不同的开发配置 IDE。定义并描述了 Visual Basic 项目和解决方案，演示了如何创建、运行和保存新项目。

第 2 章“菜单、工具栏和窗口”讲述了 IDE 的菜单和工具栏中最有用和最重要的命令。IDE 的菜单和工具栏包括上百个命令，因此该章仅重点介绍最为有用的命令。

第 3 章“定制”介绍了如何定制 IDE，讲述了如何创建、隐藏和重新安排菜单和工具栏，使得易于使用您认为最为有用的工具。

第 4 章“Windows Form Designer”讲述了可用于构建 Windows 窗体的设计器。介绍了如何创建、移动和复制控件以及调整其大小。说明了如何设置控件属性和添加代码响应控件事件。还介绍了如何使用便捷的设计工具，如智能标记和命令动词。

第 5 章“WPF 设计器”介绍了如何使用 WPF(Windows Presentation Foundation)窗体设计器。该章类似于第 4 章，但介绍的是 WPF 窗体而非 Windows 窗体。

第 6 章“Visual Basic 代码编辑器”讲述了开发人员使用的一个最为重要的窗口：代码窗口。解释了如何编写代码、设置断点、使用代码段和最大程度地利用 IntelliSense。

第 7 章“调试”介绍了 Visual Studio 提供的调试工具。讲述了调试窗口，介绍了一些诸如设置复杂断点来定位故障之类的技术。

第 II 部分 开始学习

这一部分介绍了 Visual Basic 语言的大部分内容和支持它的对象。解释了窗体、控件和程序用于构建用户界面的其他对象，介绍了如何放置后台代码来实现程序的功能。

第 8 章“选择 Windows 窗体控件”提供了可放置在窗体上的 Windows 窗体控件的一个

概览。它按类别组织这些控件，以帮助您找到可用于特定目的的控件。

第 9 章“使用 Windows 窗体控件”较详细地介绍了如何使用 Windows 窗体控件。解释了如何在设计时或运行时构建控件、如何设置复杂属性值以及如何使用许多不同类型控件常用的有用属性。还介绍了如何添加事件处理程序处理控件事件以及如何验证用户输入的数据。

第 10 章“Windows 窗体”讲述了用于 Windows 窗体应用程序中的窗体。窗体只是另一种控件，但它们在应用程序体系结构中的唯一地位意味着它们有一些特殊的属性，该章就将讲述这些属性。

第 11 章“选择 WPF 控件”概述了 WPF 控件。本章中按类别组织这些控件，以帮助您找到可用于特定目的的控件。该章类似于第 8 章，但它介绍的是 WPF 控件而不是 Windows 窗体控件。

第 12 章“使用 WPF 控件”较详细地介绍了如何使用 WPF 控件。该章类似于第 9 章，但它介绍的是 WPF 控件而不是 Windows 窗体控件。

第 13 章“WPF 窗口”介绍了 WPF 应用程序使用的取代 Windows 窗体的窗口。该章类似于第 10 章，但它介绍的是 WPF 控件而非 Windows 窗体控件。

第 14 章“程序和模块结构”介绍了构成 Visual Basic 项目的最重要的文件。讲述了项目可能包含的隐藏文件，解释了一些可提供给模块中代码的结构，如代码区域和条件编译代码。

第 15 章“数据类型、变量和常量”解释了 Visual Basic 提供的标准数据类型。介绍了如何声明和初始化变量和常量，并且介绍了变量作用域。讨论了值类型和引用类型、按值或按引用传递参数及动态创建参数变量。还介绍了如何创建数组、枚举类型和结构体。

第 16 章“运算符”描述了程序用于执行计算的运算符。这些运算符包括算术运算符(+、*、\、)、字符串运算符(&)及 Boolean 运算符(And、Or)。介绍了在表达式结合多种类型的运算符(如算术运算符和 Boolean 运算符)时引发的运算符优先级和类型转换问题。

第 17 章“子例程和函数”解释了如何使用子例程和函数将程序分解为多个可管理的部分。同时也描述了子例程重载和作用域。另外，还介绍了 lambda 函数和非严格委托(这两个功能都是 Visual Basic 2008 中的新功能)。

第 18 章“程序控制语句”描述了 Visual Basic 程序用于控制代码执行的语句。这些语句包括判定语句(如 If Then Else、Select Case、IIF 和 Choose 等)及循环语句(如 For Next、For Each、Do While、While Do 和 Repeat Until 等)。

第 19 章“错误处理”解释了错误处理和调试技术。描述了 Try Catch 结构化错误处理程序，以及从早期版本的 Visual Basic 继承而来的较老的 On Error 语句。还讨论了程序在捕获到错误时可能执行的常见操作，并且描述了防止错误发生以及使实际发生的错误显而易见的技术。

第 20 章“数据库控件和对象”解释了如何使用 Visual Basic 的标准数据库控件。这些数据库控件包括处理数据库连接的数据库连接组件、保存应用程序数据的 DataSet 组件，以及将数据在数据连接和 DataSet 之间移动的数据适配器控件。

第 21 章“LINQ”讲述了语言集成查询(LINQ)功能。解释了如何编写类似于 SQL 的查询读写对象、XML 或数据库对象。LINQ 是 Visual Basic 2008 中的新功能。

第 22 章“自定义控件”介绍了如何构建自定义控件，可以在后面将这些控件用于其他应用程序中。还介绍了创建自定义控件的三种主要方法：派生、复合及从头开始构建。该章

也提供了一些示例，可以将这些示例用作创建自己控件的起点。

第 23 章“拖动、释放和剪贴板”解释了 Visual Basic 程序如何支持拖放操作。说明了 Visual Basic 如何开始到另一个应用程序的拖动操作，如何响应由另一个应用程序开始的拖动操作，以及如何接收来自于另一个应用程序的释放操作。同时也解释了如何将数据复制到剪贴板以及如何从剪贴板中复制数据。使用剪贴板类似于执行某种类型的拖放操作，因此这些主题很自然地放在一章中进行介绍。

第 24 章“UAC 安全功能”讲述了 Vista 操作系统使用的 User Access 安全模型。有了 UAC 安全功能，所有用户都以降低的“普通”用户权限运行。如果程序必须执行要求管理员权限的任务，UAC 对话框就允许提升应用程序的权限级别。该章介绍了 UAC 安全功能，解释了如何标记程序进行权限提升。

第 III 部分 面向对象编程

这一部分的章节解释使用 Visual Basic 进行面向对象编程(OOP)的基本概念，并且描述了一些较为重要的类和对象，可以在构建应用程序时使用这些类和对象。

第 25 章“OOP 概念”介绍了隐藏在面向对象编程之后的基本思想。描述了 OOP 的三个主要特性：封装、多态和继承，解释了这些特性的优点，并且说明了如何在 Visual Basic 中利用它们。

第 26 章“类和结构体”解释了如何声明和使用类和结构体。介绍了类和结构体的概念，并且描述了它们之间的区别。还展示了基本的声明语法，并且说明了如何创建类和结构体的实例。该章也介绍了一些较为复杂的类问题(如私有类的作用域、声明事件及共享的变量和方法)。

第 27 章“名称空间”介绍了名称空间。说明了 Visual Studio 如何使用名称空间分类代码并防止名称冲突，描述了项目的根名称空间，并说明了 Visual Basic 如何使用名称空间解析名称(如函数名和类名)，以及如何将名称空间添加到应用程序中。

第 28 章“集合类”解释了 Visual Studio 中的类，可以使用这些类保存对象组。该章描述了各种集合、字典、队列及栈类，说明了如何建立这些类的强类型版本，并且提供了关于在不同的情况下决定使用哪个类的指导。

第 29 章“泛型”解释了模板，可以使用模板构建新的类以操作特定的数据类型。例如，可以构建泛型二元树，在后面使用这个泛型二元树构建类，以表示顾客订单、雇员或工作项的二元树。

第 IV 部分 绘图

这一部分描述了 Visual Basic .NET 中的绘图。解释了图形设备接口(GDI+)例程，程序使用这些例程在 Visual Basic 中绘制图像。解释了如何绘制直线和文本，如何绘制和填充圆和其他形状，以及如何加载、操作和保存位图图像。同时也解释了如何生成打印输出以及如何将报表发送给屏幕或打印机。

第 30 章“绘图基础”解释了在 Visual Basic .NET 中绘制图形的基本原理。描述了图形名称空间及它们包含的类，详细介绍了其中最重要的类 Graphics。还描述了 Paint 事件处理程序及其他事件，程序应该使用这些事件使图形最新。

第 31 章“画刷、画笔和路径”解释了除了 Graphics 之外最重要的图形类: Pen 和 Brush。说明了如何使用 Pen 绘制实线、虚线、带有自定义虚线样式的直线, 以及带有自定义纵向条纹样式的直线。同时展示了如何使用 Brush 在区域中填充颜色、阴影线样式、线性渐变色、遵循路径的其他渐变色及平铺的图像。该章也描述了 GraphicsPath 类, 该类表示一系列直线、形状、曲线和文本。

第 32 章“文本”解释了如何绘制文本字符串, 如何创建不同类型的字体, 精确地确定以特定的字体绘制大文本时该字体的外观, 以及使用 GDI+函数简化文本的定位。描述了如何使用 StringFormat 对象对文本进行排列、换行和剪裁操作, 以及如何读取和定义制表位。

第 33 章“图像处理”解释了如何加载、修改和保存图像文件, 如何读写图像中的像素, 以及如何以不同的文件格式(如 BMP、GIF 和 JPEG 等)保存结果。说明了如何使用图像提供自动重绘特性, 以及逐个像素地操作图像, 这两个操作都使用了 Bitmap 的 GetPixel 和 SetPixel 方法, 并且都使用了“不安全的”访问技术, 这就使像素操作比使用普通的 GDI+方法更为快速。

第 34 章“打印”介绍了程序发送输出给打印机的不同方法。展示了如何使用 PrintDocument 对象生成打印输出的数据, 然后可以使用 PrintDocument 立刻打印数据, 使用 PrintDialog 控件让用户选择打印机并设置它的特征, 或者使用 PrintPreviewDialog 控件让用户在打印之前预览结果。

第 35 章“报表”简述了 Crystal Reports, 使用该工具可以相对容易地在 Visual Basic 中生成报表。该章介绍了 Crystal Reports 的基础知识, 并且逐步介绍了构建简单报表的示例。

第 V 部分 与环境交互

这一部分描述了应用程序如何与它的环境交互, 程序如何在外部源(如系统注册表、资源文件和文本文件等)中保存和加载数据, 如何操作计算机的屏幕、键盘和鼠标, 以及如何通过标准对话框控件与用户交互。

第 36 章“配置和资源”描述了 Visual Basic 程序存储在运行时使用的配置和资源值的一些方法。其中一些最有用的方法包括环境变量、注册表、配置文件和资源文件。

第 37 章“流”解释了 Visual Basic 应用程序用于操作流数据的类, 包括 FileStream、MemoryStream、BufferedStream、TextReader 和 TextWriter 等。

第 38 章“文件系统对象”描述了 Visual Basic 应用程序与文件系统交互的类, 这些类包括 Directory、DirectoryInfo、File 和 FileInfo, 它们简化了创建、检查、移动、重命名, 以及删除目录和文件的操作。

第 39 章“WCF”介绍了 Windows Communication Foundation, 这是使构建面向服务的应用程序更为简单的一个库和工具集。该章讲述了如何使用新的 WCF 特性简单定义服务、如何使用配置文件配置服务, 以及如何使用 WCF 工具使用服务。

第 40 章“有用的名称空间”描述了 .NET Framework 定义的一些最常见的名称空间。概述了一些最重要的 System 名称空间, 并且给出了演示正则表达式、XML、加密、反射和 Direct3D 的较为详细的示例。

第VI部分 附录

本书的附录提供了 Visual Basic 2008 语言的分类参考。通过这些附录可以快速回顾特定命令的语法、从某个例程的一些重载版本中进行选择或者重新回忆特定类可以执行的操作。本书前面的章节提供了更多的上下文，并且解释了如何执行特定的任务，以及一种方法比另一种方法更可行的原因。

附录 A “有用的控件属性、方法和事件” 描述了对于许多不同类型的控件非常有用的属性、方法和事件。

附录 B “变量声明和数据类型” 总结了声明变量的语法。该附录也给出了基本数据类型所允许值的大小和范围。

附录 C “运算符” 总结了标准运算符，如+、<<、Or Else 和 Like 等。该附录也提供了运算符重载的语法。

附录 D “子例程和函数声明” 总结了子例程、函数和属性过程声明的语法。

附录 E “控制语句” 总结了控制程序流的语句，如 If Then、Select Case 和循环语句。

附录 F “错误处理” 总结了结构化和“经典的”错误处理。该附录描述了一些有用的异常类，并且提供了如何构建自定义异常类的示例。

附录 G “Windows 窗体控件和组件” 描述了 Visual Basic 2008 提供的 Windows 窗体控件和组件。该附录解释了在操作这些组件时最有用的属性、方法和事件。

附录 H “WPF 控件” 介绍了在构建 WPF 应用程序时工具箱中默认出现的控件。这是个相对简短的列表，给出了可用的控件，而不是像附录 G 中对标准的 Windows 窗体控件那样提供详细描述。

附录 I “Visual Basic Power Packs” 列出了一些可下载的额外工具，它们可使 Visual Basic 开发更为容易。该附录描述了 Microsoft 提供的一些 Visual Basic 6 兼容工具，以及一些内置在 Visual Basic 3 中的有用控件的 GotDotNet Power Pack。

附录 J “窗体对象” 描述了窗体。从非常实际的意义来说，窗体是另一种类型的组件。然而，窗体在 Visual Basic 应用程序中扮演了关键的角色，因此值得专门通过一个附录来进行介绍。

附录 K “类和结构体” 总结了声明类和结构体的语法，以及定义它们的构造函数和事件的语法。

附录 L “LINQ” 总结了 LINQ 语法。

附录 M “泛型” 概述了声明泛型类的语法。

附录 N “图形” 概述了用于在 Visual Basic 2008 中生成图形的对象。该附录介绍了最有用的图形名称空间。

附录 O “有用的异常类” 列出了 Visual Basic 定义的一些最有用的异常类。我们可能需要在自己的代码中抛出这些异常。

附录 P “日期和时间格式指示符” 概述了指示符字符，可以使用它们格式化日期和时间。例如，这些字符可以以 12 小时时钟或 24 小时时钟的方式显示时间。

附录 Q “其他格式指示符” 概述了数字和枚举类型的格式化。

附录 R “Application 类” 概述了 Application 类，该类提供了控制当前应用程序的属性和

方法。

附录 S “My 名称空间” 描述了 My 名称空间，该名称空间提供了遍布在 .NET Framework 其他部分的有用特性的快捷方式。该附录提供了操作应用程序、计算机硬件、应用程序窗体、资源和当前用户的快捷方式。

附录 T “流” 概述了 Visual Basic 的流类，如 Stream、FileStream、MemoryStream、TextReader 和 CryptoStream 等。

附录 U “文件系统类” 概述了应用程序可用于了解和操作文件系统的方法。该附录解释了传统的 Visual Basic 方法，如 FreeFile、WriteLine 和 ChDir，以及较新的 .NET Framework 类，如 FileSystem、Directory 和 File 等。

附录 V “示例索引” 简述了可从本书的合作网站 www.wrox.com 或 www.tupwk.com.cn/downpage，以及本书的作者网站 www.vb-helper.com/vb_prog_ref.htm 上下下载的 435 个示例程序。可以通过这一列表查看某个程序演示的特定技术。

0.4 如何使用本书

如果您是有经验的 Visual Basic .NET 程序员，则可以跳过本书第 I 部分中介绍的语言基础。不过，您或许会发现 Visual Basic 2008 中的一些新特性，因此不应该完全跳过这些章节。但是，这里介绍的大多数基本语言特性都与前面版本的相同。初步了解 Visual Basic 2008 的程序员和对 Visual Basic 2008 具有较少经验的程序员应该更加仔细地阅读这些章节。

第 III 部分 “面向对象编程” 介绍了一些特别复杂的主题。在您学习关于继承和接口的所有变体时可能会很容易产生混淆。

初学者应该花费较多的时间学习这些基础章节，因为这些章节为后面将要介绍的内容打好了基础。如果已经清楚地了解了错误处理代码的讨论，则可以较为容易地理解关于文件管理和正则表达式的讨论。

编程是一种最好通过实践来学习的技能。您可以根据喜好选择本书并快速阅读它，但是如果打开开发环境并亲自实验一些程序，则更有可能牢记本书中介绍的信息。

注意：

通常情况下，当我阅读新的编程书籍时，都会亲自完成每个示例，修改代码以查看在尝试作者没有介绍到的内容时所发生的情况。应尝试新的变体并特别注意产生的错误，但在本书中很难全面地介绍所有的错误。了解强类型的集合是一回事，使用对您有意义的数构建自己的强类型集合又是另一回事。

通过实践来学习这种观点可能会促使您跳过本书中的一些部分。例如，第 1 章详细介绍了 IDE。在阅读本章一段时间后，您可能希望跳过一些部分并亲自开始体验该环境。我非常鼓励您这样做，通过实践来学习总是好于通过阅读来学习。在对该开发环境有一些经验之后，您可以回过头并更为详细地研究第 1 章，查看是否有遗漏的内容。

本书的最后一部分是 Visual Basic 2008 参考。这些附录给出了关于该语言的较为简明的分类信息。您可以使用这些附录回顾特定操作的细节。例如，可以阅读第 8 章以学习控件用

于哪些特定的目标，然后使用附录 G 了解特定控件的属性、方法和事件。

在工作过程中，您也可以参考这些附录以获得关于特定的类、控件和语法的信息。例如，可以在附录 M 中快速查找声明泛型类的语法。如果需要关于泛型的信息，可以在第 29 章或联机帮助中进行查找。然而，如果只是为了重新回忆基本语法，则浏览附录 M 将更为快速。

0.5 必要的配备

为了阅读本书并理解其中的示例，需要一些特殊的配备。为了使用 Visual Basic 2008 并运行可在本书相关网页上找到的示例，您需要足以运行 Visual Basic 2008 的任何计算机。这意味着要有一台相当现代化的、快速的计算机，该计算机配有大量内存。可以查看 Visual Basic 2008 文档以了解 Microsoft 提出的具体需求和推荐信息。

为了构建 Visual Basic 2008 程序，您也需要一份 Visual Basic 2008 的副本。如果您有 .NET 之前的 Visual Basic 版本(如 Visual Basic 6)，则不要尝试运行本书中介绍的示例。Visual Basic 6 和 Visual Basic .NET 之间的改动非常大，许多 Visual Basic .NET 概念并不能很好地转换为 Visual Basic 6 中的概念。如果对 C# 有一些经验，则可以较为容易地将程序转换为该语言。

注意：

您甚至可以使用免费的 Visual Basic 2008 Express Edition。要了解更多信息，可访问 msdn2.microsoft.com/en-us/express/default.aspx。

然而，许多 Visual Basic 2008 发布版本都与 Visual Basic 2005 和 Visual Basic .NET 的早期版本兼容，因此可以在 Visual Basic .NET 的早期版本中使用许多示例。不过，您将不能够加载从本书的网站上下载的示例程序，而是需要将重要的代码部分复制并粘贴到您的 Visual Basic 版本中。

要使用 UAC 安全功能，必须在您的计算机上安装它。在 Windows Vista 操作系统中默认安装和激活了 UAC。

0.6 本书支持网站

编程环境(如 Visual Basic 和 Windows 操作系统)不是静态的，因此介绍它们的书籍不应是单一的。现代编程书籍都与配套网站相结合，这些网站提供示例程序、提问和解答的论坛、纠正和变更的列表以及其他补充资料。

本书也不例外。您只有访问了本书网站，才能获得物超所值的享受。

要访问本书的 Wrox 网站，请浏览到 www.wrox.com 并搜索本书的英文版的书名或书号，或者作者名 Rod Stephens。该网站包括了作者信息、摘要、可下载的示例程序、必要的纠错和论坛。另外，您也可以通过站点 www.tupwk.com.cn/downpage 下载本书的所有源代码。

注意：

请访问本书网站，并下载示例程序，这样可以尝试本书中的代码。我会密切关注 Wrox

论坛并尽快给予回复。

本书的作者网站 www.vb-helper.com/vb_prog_ref.htm 包含了类似资料,外加一些额外示例。主要的 VB Helper 网站还包含数以千计的提示、技巧和用 Visual Basic 各种版本编写的示例。

要获得有关本书或我的其他书籍的更改通知,可在 www.vb-helper.com/newsletter.html 上注册我的某个新闻通讯。这个新闻通讯(大约每周发送一次)包括了 Visual Basic 提示、技巧和示例,以及本书的更新信息和有关 Visual Basic 开发的其他内容。

如果您发现有错误或意见,可通过 RodStephens@vb-helper.com 将这些信息发送给我,我将尽可能及时更新。

0.7 勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误,但是错误总是难免的,如果您在本书中找到了错误,例如拼写错误或代码错误,请告诉我们,我们将非常感激。通过勘误表,可以让其他读者避免受挫,当然,这还有助于提供更高质量的信息。

请给 wkservice@vip.163.com 发电子邮件,我们就会检查您的反馈信息,如果是正确的,我们将在本书的后续版本中采用。

要在网站上找到本书英文版的勘误表,可以登录 <http://www.wrox.com>,通过 Search 工具或书名列表查找本书,然后在本书的细目页面上,单击 Book Errata 链接。在这个页面上可以查看到 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表,网址是 www.wrox.com/misc-pages/booklist.shtml。

0.8 P2P.WROX.COM

要与作者和同行讨论,请加入 p2p.wrox.com 上的 P2P 论坛。这个论坛是一个基于 Web 的系统,便于您张贴与 Wrox 图书相关的消息和相关技术,与其他读者和技术用户交流心得。该论坛提供了订阅功能,当论坛上有新的消息时,它可以给您传送感兴趣的论题。Wrox 作者、编辑和其他业界专家和读者都会到这个论坛上探讨问题。

在 <http://p2p.wrox.com> 上,有许多不同的论坛,它们不仅有助于阅读本书,还有助于开发自己的应用程序。要加入论坛,可以遵循下面的步骤:

- (1) 进入 p2p.wrox.com,单击 Register 链接。
- (2) 阅读使用协议,并单击 Agree 按钮。
- (3) 填写加入该论坛所需要的信息和自己希望提供的其他信息,单击 Submit 按钮。
- (4) 您会收到一封电子邮件,其中的信息描述了如何验证账户,完成加入过程。

注释:

不加入 P2P 也可以阅读论坛上的消息,但要张贴自己的消息,就必须加入该论坛。

加入论坛后,就可以张贴新消息,响应其他用户张贴的消息。可以随时在 Web 上阅读消

息。如果要让该网站给自己发送特定论坛中的消息，可以单击论坛列表中该论坛名旁边的 **Subscribe to this Forum** 图标。

关于使用 Wrox P2P 的更多信息，可阅读 P2P FAQ，了解论坛软件的工作情况以及 P2P 和 Wrox 图书的许多常见问题。要阅读 FAQ，可以在任意 P2P 页面上单击 FAQ 链接。

0.9 Visual Basic 2008 第 1 版的警告

由于装载 Visual Basic 2008 第 1 版时的配置错误，在创建新的 WPF 项目时，结果中会包含一个错误。模板项目试图从应返回 `WpfApplication1.Application` 对象的函数中返回一个 `System.Windows.Application` 对象。错误是：

```
Option Strict On disallows implicit conversions from  
'System.Windows.Application' to 'WpfApplication1.Application'.
```

错误发生在下列代码的 `Return` 语句中：

```
<Global.System.Diagnostics.CodeAnalysis.SuppressMessageAttribute( _  
"Microsoft.Performance", "CA1811:AvoidUncalledPrivateCode")> _  
Friend ReadOnly Property Application() As Application  
    Get  
        Return Global.System.Windows.Application.Current  
    End Get  
End Property
```

要修改这一错误，可悬停在 Visual Basic 显示的红色方框上，以显示 **Error Correction Options** 下拉列表。接受错误修正建议来进行修改：

```
Return CType(Global.System.Windows.Application.Current, _  
    WpfApplication1.Application)
```

这个错误应在 Visual Basic 2008 的第一个服务补丁和后续发布中修正。

在阅读第 5、11、12 和 13 章时可了解这一问题。要了解有关此问题的更多信息，可访问 support.microsoft.com/default.aspx/kb/945756。

目 录

第 I 部分 IDE

第 1 章 IDE 简介	3
1.1 不同的 IDE 外观	4
1.2 IDE 配置	5
1.3 项目和解决方案	6
1.4 启动 IDE	6
1.5 创建项目	7
1.6 保存项目	10
1.7 小结	11
第 2 章 菜单、工具栏和窗口	13
2.1 菜单	13
2.1.1 File 菜单	14
2.1.2 Edit 菜单	16
2.1.3 View 菜单	17
2.1.4 Project 菜单	19
2.1.5 Build 菜单	23
2.1.6 Debug 菜单	24
2.1.7 Data 菜单	24
2.1.8 Format 菜单	24
2.1.9 Tools 菜单	25
2.1.10 Test 菜单	29
2.1.11 Window 菜单	29
2.1.12 Community 菜单	30
2.1.13 Help 菜单	30
2.2 工具栏	31
2.3 辅助窗口	31
2.3.1 Toolbox 窗口	32
2.3.2 Properties 窗口	34
2.4 小结	35

第 3 章 定制	37
3.1 添加命令	37
3.2 删除命令	38
3.3 修改命令	39
3.4 创建键盘快捷键	40
3.5 小结	41
第 4 章 Windows Form Designer	43
4.1 设置设计器的选项	43
4.2 添加控件	44
4.3 选择控件	45
4.4 复制控件	45
4.5 控件的移动和大小设置	46
4.6 安排控件	46
4.7 设置属性	47
4.7.1 设置组属性	47
4.7.2 设置几个控件的属性	47
4.7.3 使用智能标记	48
4.7.4 使用命令动词	48
4.8 给控件添加代码	49
4.9 小结	50
第 5 章 WPF 设计器	51
5.1 早期版本的警告	51
5.2 设计器窗口	52
5.3 添加控件	53
5.4 选择控件	54
5.5 复制控件	55
5.6 控件的移动和大小设置	55
5.7 设置属性	56
5.8 设置组属性	57
5.9 给控件添加代码	57

5.10	小结	58
第 6 章	Visual Basic 代码编辑器	59
6.1	边空图标	60
6.2	大纲视图	61
6.3	工具提示	62
6.4	IntelliSense	63
6.5	代码着色和突出显示	65
6.6	代码段	67
6.6.1	使用代码段	67
6.6.2	创建代码段	68
6.7	运行时的代码编辑器	70
6.8	小结	71
第 7 章	调试	73
7.1	Debug 菜单	73
7.2	Debug→Windows 子菜单	75
7.3	Breakpoint 窗口	79
7.4	Command 窗口和 Immediate 窗口	80
7.5	小结	82

第 II 部分 开始学习

第 8 章	选择 Windows 窗体控件	85
8.1	控件概述	85
8.2	选择控件	89
8.2.1	包含控件和安排控件	89
8.2.2	进行选择	91
8.2.3	输入数据	92
8.2.4	显示数据	92
8.2.5	提供反馈	93
8.2.6	初始化动作	94
8.2.7	显示图形	95
8.2.8	显示对话框	95
8.2.9	支持其他控件	96
8.3	第三方控件	96
8.4	小结	97
第 9 章	使用 Windows 窗体控件	99
9.1	控件和组件	99

9.2	创建控件	100
9.2.1	在设计时创建控件	101
9.2.2	添加控件到容器	101
9.2.3	在运行时创建控件	102
9.3	属性	104
9.3.1	设计时的属性	104
9.3.2	运行时的属性	108
9.3.3	有用的控件属性	109
9.3.4	位置和尺寸属性	112
9.4	方法	113
9.5	事件	114
9.5.1	在设计时创建事件处理程序	114
9.5.2	WithEvents 事件处理程序	115
9.5.3	在运行时设置事件处理程序	116
9.5.4	控件数组事件	117
9.5.5	验证事件	117
9.6	小结	122

第 10 章	Windows 窗体	123
10.1	透明性	123
10.2	About 对话框、启动画面和登录窗体	126
10.3	鼠标光标	127
10.4	图标	129
10.4.1	应用程序图标	130
10.4.2	通知图标	130
10.5	子控件采用的属性	131
10.6	属性复位方法	132
10.7	重写 WndProc	132
10.8	SDI 和 MDI	133
10.8.1	MDI 的功能	134
10.8.2	MDI 事件	137
10.8.3	MDI 与 SDI 的对比	138
10.9	MRU 列表	139
10.10	对话框	141
10.11	向导	143
10.12	小结	144

第 11 章	选择 WPF 控件	145
11.1	控件概述	146

11.2	包含和安排控件	146	13.2.4	向导应用程序	194
11.3	选择控件	148	13.3	小结	198
11.4	输入数据	148	第 14 章	程序和模块结构	199
11.5	显示数据	149	14.1	隐藏文件	199
11.6	提供反馈	149	14.2	代码文件结构	203
11.7	执行动作	150	14.2.1	代码区域	204
11.8	表示图形和媒体	151	14.2.2	条件编译	205
11.9	提供导航	152	14.2.3	名称空间	211
11.10	管理文档	152	14.3	印刷代码元素	213
11.11	数字墨水	152	14.3.1	注释	213
11.12	小结	153	14.3.2	XML 注释	214
第 12 章	使用 WPF 控件	155	14.3.3	续行	217
12.1	WPF 概念	155	14.3.4	行连接	218
12.1.1	用户界面和代码分离	156	14.3.5	行标签	218
12.1.2	WPF 控件层次结构	156	14.4	小结	218
12.2	IDE 中的 WPF	157	第 15 章	数据类型、变量和常量	221
12.2.1	编辑 XAML	158	15.1	数据类型	222
12.2.2	编辑 Visual Basic 代码	161	15.2	类型字符	223
12.3	XAML 功能	163	15.3	数据类型转换	225
12.3.1	对象	164	15.3.1	收缩转换	226
12.3.2	资源	166	15.3.2	数据类型解析方法	228
12.3.3	样式	167	15.3.3	扩大转换	228
12.3.4	模板	168	15.4	变量声明	228
12.3.5	变换	170	15.4.1	attribute_list	229
12.3.6	动画	171	15.4.2	accessibility 子句	229
12.3.7	绘图对象	173	15.4.3	Shared	230
12.4	过程化 WPF	177	15.4.4	Shadows	231
12.5	文档	181	15.4.5	ReadOnly	233
12.5.1	流文档	182	15.4.6	Dim	233
12.5.2	固定文档	183	15.4.7	WithEvents	234
12.5.3	XPS 文档	184	15.4.8	name	236
12.6	小结	184	15.4.9	bounds_list	236
第 13 章	WPF 窗口	187	15.4.10	New	237
13.1	Window 应用程序	187	15.4.11	as 子句和推断类型	238
13.2	Page 应用程序	189	15.4.12	initialization_expression	239
13.2.1	浏览器应用程序	189	15.4.13	多个变量的声明	242
13.2.2	框架应用程序	191	15.5	Option Explicit 和	
13.2.3	PageFunction 应用程序	192		Option Strict	243