

114

TP312BA
014

“用实例学编程”丛书

用实例学 Visual Basic .NET

Visual Basic .NET by Example

[美] Gabriel Oancea 著
Bob Donald

魏迎梅 等译

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书主要介绍了如何使用 Visual Basic .NET 工具开发各类应用程序,书中囊括了一些开发 Visual Basic .NET 应用程序所必备的知识,如开发环境概述、Visual Basic 语言本身、面向对象的基本概念、数据库的基本概念和 XML 等。本书首先介绍了有关 Visual Basic .NET 语言的一些基本知识,具体包括数据类型、变量和语句等。接着循序渐进地讲解了有关面向对象编程的一些基本概念,并结合实例阐述了使用 Visual Basic .NET 开发用户接口、数据访问、SQL,以及使用 Web 表单开发 Web 应用程序的各项内容。

在本书的最后还提供了一个关于 UML 速成教材的附录。本书 Web 站点中还包括许多将 Visual Basic 6.0 移植到 Visual Basic .NET 应用程序,XML (eXtensible Markup Language) 以及关于 Visual Basic .NET 语言的一些参考资料。

本书全文通俗易懂,又不失严谨,适合于有意学习 Visual Basic .NET 的读者使用。

Authorized translation from the English language edition published by Que Corporation. Copyright © 2002. 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 the Publisher. Simplified Chinese language edition published by Publishing House of Electronics Industry, Copyright © 2002.

本书中文简体版专有翻译出版版权由 Pearson 教育集团所属的 Que Corporation 授予电子工业出版社。其原文版权及中文翻译出版版权受法律保护。未经许可,不得以任何形式或手段复制或抄袭本书内容。

版权贸易合同登记号:图字:01-2001-4960

图书在版编目(CIP)数据

用实例学 Visual Basic .NET / (美)欧塞(Oancea, G.)著;魏迎梅等译.-北京:电子工业出版社,2002.7
(“用实例学编程”丛书)

书名原文:Visual Basic .NET by Example

ISBN 7-5053-7808-2

I. 用... II. ①欧... ②魏... III. BASIC 语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字(2002)第051754号

责任编辑:徐 申

印刷者:北京市增富印刷有限责任公司

出版发行:电子工业出版社 www.phei.com.cn

北京市海淀区万寿路173信箱 邮编:100036

经 销:各地新华书店

开 本:787 × 1092 1/16 印张:42.75 字数:1067千字

版 次:2002年7月第1版 2002年7月第1次印刷

定 价:59.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。

联系电话:(010)68279077

前 言

“用实例学编程”系列

“用实例学编程”系列怎样将你变成更出色的程序开发人员？“用实例学编程”系列用尽可能好的方法讲解软件开发技术。在引入一个概念之后，你会看到一个或多个使用这个概念的实例。正文作为指导者，形象地提醒你所学概念要注意的地方，并且说明新的使用方法。实例的数量很大；在讲解新材料的时候，你会看到一个接一个用来说明如何使用新学概念的实例。

本系列的原则很简单，即讲解软件开发技术最好的方法是通过实例。向初学者讲解一门编程语言，仅仅使用语言特征、语法特征和语言参考（language reference）还不够。初学者只有通过阅读很多实例，其中用到刚学过的新特性，并且实际运行实例程序，才能对这种语言有一种感性的认识。

本书读者对象

本书适合于有意学习 Visual Basic .NET 的各类读者对象。学习本书之前首先需要对 Microsoft Windows 有一个基本的了解。只要基本掌握诸如文件、目录、Internet、应用和程序之类的一些简单概念，学习本书就不会有任何困难。

本书包含了一些开发 Visual Basic .NET 应用程序所必备的基本背景知识，如开发环境概述、Visual Basic 语言、面向对象的基本概念、数据库的基本概念和 XML 等。

本书适合于以下读者：

- 知道如何使用 Microsoft Windows 95/98/NT/2000，并且希望学习编程
- C++ 和 / 或 Java 的开发人员，想学习新的 Visual Basic .NET 语言
- 想学习这门新语言的 Visual Basic 6.0 的开发人员
- 希望学习有助于进行企业应用软件设计和开发的一种语言

本书的组织结构

本书分为六个部分。第一部分简单介绍了 Microsoft .NET 和 Visual Studio .NET，为读者展现了开发 Visual Basic .NET 应用程序的大环境。本部分首先提供了一个非常简单的应用实例，通过编译和执行这个实例，使读者能够尽快熟悉如何使用 Visual Basic .NET 构造一个应用程序。

第二部分包括 Visual Basic .NET 语言的一些基本知识，这是了解这门语言本身的第一步，其中有数据类型、变量和语句等。

第三部分转而讲述面向对象的一些基本概念。由于 Visual Basic 的早期版本中并不需要任何面向对象的知识，而 Visual Basic .NET 对这种语言做了一些改进，因此对于新版本的 Visual Basic .NET，要想成为一名优秀的开发人员，就必须掌握面向对象这部分知识，因而本书提供了这些内容。

第四部分讲述如何使用 Visual Basic .NET 开发用户接口，包括如何开发作为应用程序一部分的表单和控件等。

第五部分介绍了数据访问。当前的大多数应用程序都需要处理某些格式的数据，ADO.NET 和 XML 是最常用的数据处理方法，因而本部分包含了这方面的内容，使得读者能够掌握如何在 Visual Basic .NET 中使用这项技术。此外，由于在 ADO.NET 提供的数据库访问中会用到 SQL，本部分还包括了有关 SQL 的基本概述。

第六部分介绍了如何使用 Web 表单开发 Web 应用程序。这部分主要探讨了在开发 Web 应用程序的过程中所涉及的一些问题，以及如何使用 Visual Basic .NET 解决这些问题。

在本书的最后还提供了一个关于 UML (Unified Modeling Language) 速成教材的附录。Web 站点包括将 Visual Basic 6.0 移植到 Visual Basic .NET 应用程序，XML (eXtensible Markup Language) 以及关于 Visual Basic .NET 语言的一些参考资料。

本书约定

本书中的例子都用下段中左边的图标标注。



源代码都用等宽字体给出，例如：

```
Module Module1
Sub Main()
System.Console.Out.WriteLine("An Example")
End Sub
End Module
```

在很多例子的代码中经常会包含一些用黑体表示的代码行，这主要是为了引起读者的注意，因为这个例子的重点和相关的说明都在这些代码行上。当读者阅读本书及书中的例子时，为了更好地理解和领悟每个例子的意途，需要特别注意这些突出显示的代码行。



每个例子的输出用左边的图标表示。

说明：说明是对刚才所提供资料的补充，它们通常用于阐明一些概念和过程。

提示：提示给出了关于某个特定问题的快捷操作或工作区。

注意：注意对一些需要特别预防的事情提出警告，使读者避免浪费时间和并减小出现错误的可能性。

与 VB6 的兼容性：这些信息给出了 Visual Basic 语言从 Visual Basic 6.0 到 Visual Basic .NET 的一些变化。

Visual Basic .NET 中的新内容：这些信息给出了 Visual Basic .NET 中的新特征。

Web 站点

本书中的代码例子、每章最后“练习”、“复习”和“实践”中的问题答案以及附加参考资源都可以在 www.quepublishing.com 中找到。

下文预告

本书从 Microsoft .NET 概述开始，然后对 Visual Studio .NET 环境做一个初步的介绍，这样读者就会逐渐熟悉本书中所用到的开发环境。

目 录

第一部分 初步熟悉 Visual Studio .NET 环境

第 1 章 环境概述	2
1.1 引言	2
1.2 Visual Studio .NET 简介	6
1.3 下文预告	13
第 2 章 创建第一个应用程序	16
2.1 引言	16
2.2 创建一个新项目	17
2.3 查看生成的项目	18
2.4 下文预告	24
第 3 章 编译和运行应用程序	26
3.1 编连项目	26
3.2 编连信息	27
3.3 编连和重新编连	28
3.4 运行应用程序	28
3.5 编连过程的细节	30
3.6 下文预告	30
第 4 章 Visual Basic .NET 项目类型	33
4.1 Windows 应用程序	33
4.2 类库	34
4.3 Windows 控件库	35
4.4 ASP.NET Web 应用程序	36
4.5 ASP.NET Web 服务	37
4.6 Web 控件库	38
4.7 控制台应用程序	38
4.8 Windows 服务程序	38
4.9 空项目	39
4.10 空 Web 项目	39
4.11 下文预告	40

第二部分 语言基础

第 5 章 语言描述	44
5.1 程序设计简介	44
5.2 编程语言	45
5.3 Visual Basic 项目的结构	46
5.4 标识符、文字、关键字和空白	49
5.5 说明	50
5.6 下文预告	51
第 6 章 基本数据类型	54
6.1 Numeric 数据类型	54
6.2 Boolean 数据类型	58
6.3 String 和 Char 数据类型	58
6.4 Date 数据类型	59
6.5 其他数据类型	60
6.6 下文预告	60
第 7 章 数据转换与表达	63
7.1 隐式数据转换	63
7.2 显式数据转换	65
7.3 in-place 运算符	68
7.4 比较运算符	70
7.5 布尔运算符	70
7.6 表达式和运算符的优先级	71
7.7 下文预告	72
第 8 章 数组、枚举和结构	76
8.1 数组	76
8.2 枚举	80
8.3 结构声明	83
8.4 使用结构化数据类型	85
8.5 下文预告	87
第 9 章 控制语句	91
9.1 If 语句	91
9.2 Select Case 语句	94
9.3 For 循环	98
9.4 Do Loop 循环	101

9.5 While 循环	104
9.6 下文预告	105
第 10 章 子程序和函数	109
10.1 结构化程序设计简介	109
10.2 子程序	112
10.3 函数	116
10.4 作用域和生存期	119
10.5 下文预告	125
第 11 章 调试和异常处理简介	129
11.1 准备调试一个程序	129
11.2 在代码中步进执行	130
11.3 结构化异常处理 (Structured Exception Handling, SEH)	136
11.4 非结构化异常处理	139
11.5 下文预告	140
第 12 章 Microsoft .NET 框架——组合体和名字空间	143
12.1 软件开发生命周期	143
12.2 组合体	144
12.3 名字空间	148
12.4 有用的名字空间	153
12.5 下文预告	154

第三部分 使用对象

第 13 章 对象和类——面向对象程序设计简介	158
13.1 对象和类简介	158
13.2 在 Visual Basic .NET 中定义类	159
13.3 类域	162
13.4 类的构造函数	164
13.5 下文预告	169
第 14 章 属性	172
14.1 属性的定义	172
14.2 Property 语句	173
14.3 使用属性	175
14.4 参数化的属性	177
14.5 域和属性	181
14.6 下文预告	181

第 15 章 方法	184
15.1 类方法的定义	184
15.2 方法重载	186
15.3 使用方法	187
15.4 下文预告	194
第 16 章 共享类成员	197
16.1 共享成员的定义	197
16.2 共享方法	200
16.3 关于共享成员的提示	201
16.4 下文预告	201
第 17 章 继承	204
17.1 继承的定义	204
17.2 简单继承	208
17.3 继承的含义	212
17.4 多级继承	215
17.5 多态性	217
17.6 抽象类	219
17.7 .NET 框架中的继承	223
17.8 下文预告	224
第 18 章 接口	227
18.1 接口的定义	227
18.2 使用接口	228
18.3 接口继承	232
18.4 .NET 框架中的接口	233
18.5 下文预告	234
第 19 章 集合	237
19.1 集合的基础知识	237
19.2 ArrayList	240
19.3 Hashtable	242
19.4 SortedList	244
19.5 Queue 和 Stack	247
19.6 NamedObjectCollectionBase —— 创建自己的集合	249
19.7 关于使用集合的提示	252
19.8 下文预告	252

第 20 章 面向对象程序设计高级概念	255
20.1 关系	255
20.2 对象的持久性	259
20.3 开发方法	260
20.4 下文预告	262

第四部分 使用窗体和控件

第 21 章 用户界面编程的基本概念	266
21.1 简介	266
21.2 应用程序代码的正确划分	267
21.3 单文档界面和多文档界面	268
21.4 控件	270
21.5 菜单	270
21.6 事件	271
21.7 多语言支持	271
21.8 对键盘和鼠标的支持	273
21.9 字体和颜色	275
21.10 下文预告	276

第 22 章 窗体	279
22.1 引言	279
22.2 大小和位置	281
22.3 颜色和背景	286
22.4 边框类型	288
22.5 窗体设计器	289
22.6 坞排和锚定	291
22.7 事件	292
22.8 滚动	296
22.9 消息框	298
22.10 对话框	306
22.11 单文档界面和多文档界面	307
22.12 下文预告	308

第 23 章 控件	311
23.1 简介	311
23.2 Control 类	313
23.3 使用控件	316
23.4 使用.NET 中通用控件的例子	327
23.5 创建自己的控件	347

23.6	鼠标事件	353
23.7	键盘事件	354
23.8	下文预告	354
第 24 章	菜单	357
24.1	引言	357
24.2	菜单和菜单项	358
24.3	菜单项和事件处理程序	361
24.4	菜单分隔条和隔断	367
24.5	允许、禁止和可见性	370
24.6	记忆键和快捷方式	371
24.7	复选菜单项	374
24.8	菜单中的单选按钮	375
24.9	嵌套菜单	377
24.10	动态菜单	380
24.11	上下文菜单	382
24.12	自绘制菜单项	383
24.13	SDI、MDI 和菜单	390
24.14	下文预告	391
第 25 章	用户界面中的程序流控制	394
25.1	应用程序的开始与结束	394
25.2	窗体初始化和终止	399
25.3	窗体关闭	401
25.4	事件驱动的程序设计	402
25.5	下文预告	403
第 26 章	图形程序设计	406
26.1	.NET 中的图形简介	406
26.2	文字	407
26.3	字体	409
26.4	页面单位	411
26.5	位置和大小	412
26.6	字符串的对齐与回绕	413
26.7	画笔、线和矩形	415
26.8	多边形	424
26.9	弧线和椭圆	426
26.10	开放曲线和闭合曲线	428

26.11 图像和图标	429
26.12 填充	432
26.13 剪辑	434
26.14 高级特征	435
26.15 下文预告	435
第 27 章 打印	438
27.1 .NET 中的打印简介	438
27.2 打印一个窗体	439
27.3 PrintPage 事件和 PrintPageEventArgs	444
27.4 多页打印	446
27.5 打印预览	449
27.6 创建定制的 PrintDocument 类	451
27.7 下文预告	455
第 28 章 为用户界面开发帮助系统	458
28.1 .NET 中的 Help 简介	458
28.2 Help 类	459
28.3 HelpProvider 组件	462
28.4 上下文相关帮助	463
28.5 弹出式帮助	469
28.6 ToolTip 帮助	474
28.7 下文预告	478

第五部分 使用 ADO.NET 和 XML 操纵数据

第 29 章 使用 ADO.NET 访问数据库	482
29.1 数据库基础	482
29.2 .NET 框架数据名字空间	484
29.3 .NET 数据提供者	485
29.4 服务器管理器	489
29.5 下文预告	492
第 30 章 数据集	495
30.1 使用数据集来读取和修改数据	495
30.2 排序和过滤数据集	500
30.3 把数据集绑定到控件上	503
30.4 多表数据集和关系	505
30.5 有类型和无类型的数据集	509
30.6 下文预告	511

第 31 章 SQL 命令	514
31.1 SQL 入门	514
31.2 使用 SqlCommand 和 SqlParameter	516
31.3 下文预告	523
第 32 章 XML 数据访问	526
32.1 从数据集中展示 XML 和 XML 方案	526
32.2 从数据集中编写 XML 和 XML 方案	534
32.3 把 XML 和 XML 方案读到数据集中	535
32.4 XML 和数据集的不同步	536
32.5 下文预告	542
第 33 章 读取 XML	545
33.1 概述	545
33.2 加载 XML 文件	545
33.3 遍历 XML	554
33.4 查找指定的结点和元素	559
33.5 读取内容	562
33.6 下文预告	564
第 34 章 生成 XML 文档	567
34.1 创建元素和属性	567
34.2 创建说明	570
34.3 创建文本	572
34.4 写结点	575
34.5 保存 XML 文档	576
34.6 XSL 转换	578
34.7 下文预告	585

第六部分 开发 Web 应用程序

第 35 章 Web 窗体介绍	590
35.1 基本概念	590
35.2 第一个 Web 应用程序	591
35.3 Web 窗体的属性及方法	594
35.4 在 Web 窗体上使用的控件	597
35.5 Web 窗体的事件模型	601
35.6 下文预告	613
第 36 章 开发 Web 应用程序	616
36.1 使用数据绑定控件访问数据	616

36.2 状态管理	626
36.3 开发 Web 用户控件	633
36.4 下文预告	636
第 37 章 Web 服务	639
37.1 Web 服务的简介	639
37.2 开发 Web 服务器示例	641
37.3 作为语言扩展机制的属性	646
37.4 在客户端应用程序中使用 Web 服务	649
37.5 下文预告	654

第七部分 附 录

附录 A UML 速成教程	658
---------------------	-----

第一部分

初步熟悉 Visual Studio .NET 环境

- 第 1 章 环境概述
- 第 2 章 创建第一个应用程序
- 第 3 章 编辑和运行应用程序
- 第 4 章 Visual Studio .NET 项目类型

第1章 环境概述

本章通过对开发环境的概述来介绍 Visual Studio .NET，其目的是使读者在正式学习 Visual Studio .NET 语言本身之前，先熟悉该语言的开发环境。

本章包括以下内容：

- 关于 Microsoft .NET、Visual Studio .NET 和 Visual Basic .NET 的概述
- Visual Studio .NET 简介

1.1 引言

Visual Basic 6 的用户将会认识到，Visual Basic .NET 不仅仅是对 Visual Basic 6 的简单升级，而是将 Visual Basic 语言封装到一个更宽广的舞台，从而使得开发人员可以创建更多类型的应用程序，包括那些用旧版 Visual Basic 非常难以实现的应用程序。欢迎第一次接触 Visual Basic 的读者阅读本书，你们将会发现 Visual Basic .NET 是一种功能非常强大的语言，可以使用户编写出一流的应用程序。

Visual Basic 从诞生到今天已经走过了漫长的发展历程。作为一种传统的编程语言，Visual Basic（简称 VB）一直致力于简化 Microsoft Windows 环境中的应用软件开发工作，包括 Windows 3.1、Windows 95、Windows 98、Windows NT 以及 Windows 2000。用 Visual Basic 所编写的最常见的应用程序类型是运行于单机上的独立应用程序，其中一个典型的例子是计算器程序，它是附带在 Microsoft Windows 操作系统中的，如图 1.1 所示。通过使用 VB，可以在非常短的时间里开发出应用程序；而在 VB 出现之前，用户需要用一种更复杂的编程语言来编写应用程序，如 C 或 C++，学习这些语言需要很长的时间，用它们来开发应用程序同样需要很长的时间。要编写一个简单的应用程序，VB 的初级开发人员比 VC 的中级开发人员所需要的时间还要短。VB 这种高生产率的优点，使得它成为一种最受欢迎的语言。

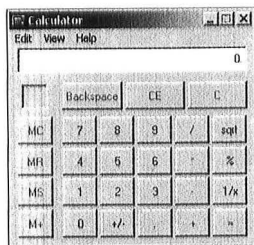


图 1.1 一个简单的应用程序

接下来开发人员开始使用 VB 编写需要把数据存储到数据库中的应用程序。通过这种方法，应用程序可以按一种结构化的格式存储数据，从而使得在关闭计算机之后，这些数据仍然保留在

计算机中。当下一次打开计算机并运行这个应用程序时,可以再次看到这些数据,并且可以添加新的数据。开始时,数据库必须与应用程序共存于同一台计算机中,后来,可以将数据库迁移到网络的中心位置,从而使同一个网络中的所有计算机都可以与同一个数据库对话,这样一个数据库就可以供多个用户使用,也就是说这些可以共享数据。这种类型的应用程序称为客户/服务器应用程序(client server application),程序本身是客户,数据库是服务器。图1.2给出了一个客户/服务器应用的例子,运行于一台计算机上的应用程序与运行于另一台计算机上的数据库进行通信。

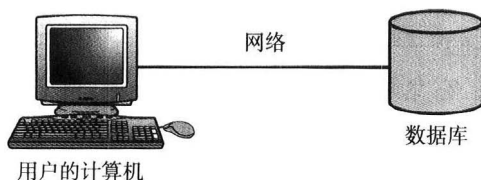


图 1.2 一个客户/服务器应用程序

一直到 Visual Basic 6,许多开发人员仍然使用客户/服务器模式。客户/服务器模式进一步发展成为n级应用程序(n-tiered application,如图1.3所示),其基本思想是按层次开发应用程序。当前比较典型的应用程序通常都使用三个不同的层:表述层、应用逻辑层和数据服务层。每一层都完成一个特定的功能,并且被它的上一层所使用。例如一个在线的电子商务应用程序,其表现层将显示对用户有用的信息,如记账信息和定货单。表述层与应用逻辑层通信,应用逻辑层为应用程序提供具体的功能,例如根据购买的产品计算应该从账户中减去的数值。为了读写永久数据,例如把定货单信息存储到数据库中,应用逻辑层需要与数据服务层进行通信。对于一个n级应用程序,应用程序的每一层都可以在一台单独的计算机上运行。如果所有层都在同一台计算机中运行,则这个应用程序被认为是一个独立应用程序,或者单级应用程序;如果应用程序在三个独立的计算机上运行(每一层都在一台独立的机器上运行),那么该应用程序就被认为是一个三级应用程序,以此类推。另一种对层和级的解释是:“层”(layer)体现了对应用程序中代码的逻辑划分,“级”(tier)体现了代码实际上在哪里运行。

为了与运行在不同计算机中的其他层进行对话,这些层需要一种在网络中进行通信的方法。许多Web应用程序都遵循n级模式,并把它当作自身体系结构的一部分。Web应用的例子有www.cnn.com、www.yahoo.com和www.amazon.com。在Microsoft Windows平台上的通信机制基于COM(Component Object Model)技术。

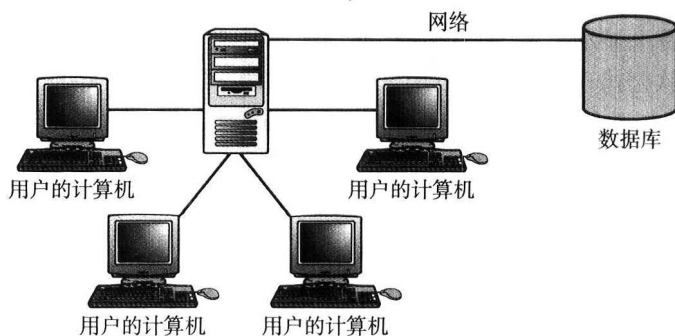


图 1.3 一个n级应用程序