

实用编程百例丛书

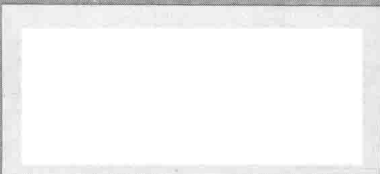
Visual Basic.NET

实用编程100例

付军 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



TP312VB

5/

Visual Basic.NET 实用编程 100 例

付 军 编著

中国铁道出版社

2003·北京

(京)新登字 063 号

内 容 简 介

本书共分 10 部分,通过对 100 个实例的讲解,让读者掌握使用 VB.NET 进行应用程序开发的各个方面。通过每一部分实例的重点讲述,可以让读者更加清楚地了解 Microsoft Visual Studio.NET 的功能及所用的控件、函数和方法的作用。

本书针对初学者 Microsoft Visual Studio.NET 的基础比较弱的特点,特意在第一部分讲述了较多的简单实例,使读者通过对实例的学习来拓宽编程思路,同时也让对 VB6 很熟悉的读者可以迅速掌握 Microsoft Visual Studio.NET 和 VB6.0 的不同之处。

希望通过本书实例的讲解,使读者很快地对 Microsoft Visual Studio.NET 有一些基本的了解,对拓展 Microsoft Visual Studio.NET 的编程思路有一定的帮助。

本书附有配套光盘,它提供了全部实例的源程序,以方便读者查看、调用和修改。本书适合于初、中级读者学习,对于高级读者也有一定的参考价值。

图书在版编目(CIP)数据

Visual Basic.NET 实用编程 100 例/付军编著. —北京:中国铁道出版社, 2003. 4

(实用编程百例丛书)

ISBN 7-113-05203-7

I. V II. 付... III. BASIC 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2003)第 025122 号

书 名: Visual Basic.NET 实用编程 100 例

作 者: 付 军

出版发行: 中国铁道出版社(100054,北京市宣武区右安门西街8号)

策划编辑: 严晓舟 魏 春

责任编辑: 苏 茜 霍志国

封面设计: 孙天昭 陈 帅

印 刷: 河北省遵化市胶印厂

开 本: 787×1092 1/16 印张: 23 字数: 539 千

版 本: 2003 年 5 月第 1 版 2003 年 5 月第 1 次印刷

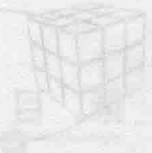
印 数: 1~5000 册

书 号: ISBN 7-113-05203-7/TP·916

定 价: 39.00 元

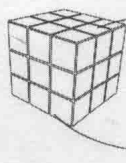
版权所有 侵权必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。



目 录

第一部分 基本组件与基础知识	1
实例 1 建立主菜单和弹出菜单.....	3
实例 2 会变背景图片的按钮.....	5
实例 3 获取计算机名称.....	7
实例 4 查看磁盘序列号程序.....	9
实例 5 防止多个文件运行.....	11
实例 6 建立计时器.....	13
实例 7 单选按钮与复选框.....	16
实例 8 多文本编辑器.....	19
实例 9 查找函数.....	24
实例 10 查找字符串.....	26
实例 11 拖动无标题窗口.....	29
实例 12 闪烁标题栏.....	31
实例 13 光驱开启.....	33
实例 14 注销.....	35
实例 15 系统情况.....	37
实例 16 系统日期.....	39
实例 17 系统目录.....	41
实例 18 任务管理.....	43
实例 19 磁盘空间获取.....	47
实例 20 控件的移动.....	49
实例 21 任务栏的隐藏与显示.....	51
实例 22 Windows 版本演示.....	53
实例 23 格式化磁盘.....	56
实例 24 监控程序.....	58
实例 25 删除文件夹.....	60
实例 26 光标速率.....	63
实例 27 数值输入.....	65
实例 28 最大最小化按钮.....	68
实例 29 屏幕保护程序.....	71
实例 30 浏览日历.....	73

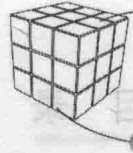


Visual Basic.NET

实用编程100例

第二部分 图形、图像及多媒体	81
实例 31 播放 CD.....	83
实例 32 播放 Wave、Midi 和 Avi 文件.....	86
实例 33 窗体插图.....	91
实例 34 绘制分析图形.....	93
实例 35 模拟画笔方位显示.....	100
实例 36 媒体播放.....	103
第三部分 使用对话框	105
实例 37 文件属性对话框.....	107
实例 38 颜色对话框.....	111
实例 39 打开及保存文件对话框.....	113
实例 40 输入及查找对话框.....	115
实例 41 消息提示框.....	119
第四部分 建立数据报表	121
实例 42 建立简单的数据表.....	123
实例 43 建立一个数据查询表格.....	127
实例 44 建立简单的报表.....	129
实例 45 建立一个订单入口报表.....	133
实例 46 建立一个窗体数据库表.....	135
第五部分 建立数据库应用程序	137
实例 47 职工记录编辑.....	139
实例 48 使用 SQL 计数.....	152
实例 49 SQL 查询.....	155
实例 50 建立 ADO 数据库连接.....	157
实例 51 个人通讯录.....	161
第六部分 字符串处理	171
实例 52 查找首字符串.....	173
实例 53 查找和替代字符串程序.....	175
实例 54 反转字符串.....	178
实例 55 分解字符串.....	180
实例 56 快速排序和冒泡排序.....	182
实例 57 以文本方式显示字符串提示信息.....	184
实例 58 找到字符串的最右边部分.....	186

第七部分 文件的处理	189
实例 59 自动使用 HTML 格式生成 VB 项目的项目报告	191
实例 60 文件比较.....	214
实例 61 读写注册表代替把设置保存到 INI 文本文件	219
实例 62 获取文件信息.....	222
实例 63 检测文件属性.....	226
实例 64 将文件分块读取.....	229
实例 65 历史文件的演示.....	231
实例 66 目录浏览.....	237
实例 67 系统监控程序.....	240
实例 68 使用 GetLogicalDriveStrings API 检索驱动器字母	249
第八部分 建立 Internet 应用程序	253
实例 69 简单的邮件发送.....	255
实例 70 建立和断开网络映射驱动器.....	258
实例 71 利用 FTP 下载.....	261
实例 72 批量转换 TXT 文件为 HTML 文件.....	264
实例 73 在不同程序间拖放获得文字.....	268
实例 74 提取收藏.....	270
实例 75 电话拨号上网.....	272
实例 76 给 ICQ 发送信息.....	274
实例 77 连续批量 ping 测试.....	277
实例 78 使用 UDP 协议发送信息.....	285
实例 79 TFTP 服务-客户端程序	287
实例 80 TFTP 服务-服务器程序	290
实例 81 获取电子邮件号码.....	292
实例 82 快速扫描计算机端口 PortScanner 2000	295
实例 83 一个 2 合 1 的网络交谈 (聊天) 程序.....	299
实例 84 多 WINSOCK C/S 结构的高效编程示例——客户端程序	301
实例 85 多 WINSOCK C/S 结构的高效编程示例——服务器程序	303
第九部分 数字应用程序	305
实例 86 简单的计算程序.....	307
实例 87 计算器.....	312
实例 88 随机数字应用程序.....	318
实例 89 模仿 LINUX 上面的那个二进制时钟	320
实例 90 一个有关一元二次方程的实例.....	323




Visual Basic.NET

实用编程100例

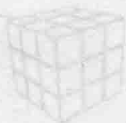
第十部分 高级应用及其他 327

实例 91	模拟鼠标单双击.....	329
实例 92	加密——对文本文件的加密.....	333
实例 93	用新的方法强制你的程序退出.....	336
实例 94	演示斐波纳契数列.....	338
实例 95	SHA-1 的密码检验示例.....	340
实例 96	建立文件关联.....	350
实例 97	在海水里面游动的美人鱼.....	353
实例 98	扑克光标.....	355
实例 99	帮助精灵.....	357
实例 100	百叶窗.....	359



第一部分 基本组件与基础知识

本部分是介绍用 Microsoft Visual Studio.NET 进行程序设计的基本内容，即 VB.NET 的常用组件和 VB.NET 的基本函数的应用。在这一部分中，概述了一些 VB.NET 基本组件的使用和一些简单程序的制作和例举部分程序代码。其中包括对 VB 函数的查找、如何建立主菜单和弹出菜单、如何制做多文本编辑器、如何获取计算机名称、如何制作单选按钮与复选按钮、如何编辑单选框与复选框、如何制作日历、如何进行数值的输入、如何制作监控程序、如何制作计时器、如何制作最大最小化按钮、如何设置屏幕保护及密码、如何查找文件、如何设置会变背景图片的按钮、如何获取磁盘序列号、如何进行文件夹的删除、如何制作闪烁标题栏、如何防止多个文件运行等一些简单程序和经典例子。如果用户对 VB6.0 有一定的了解，相信对此部分将会很快能够理解。



实例 1

建立主菜单和弹出菜单

实例说明

本实例主要说明如何建立下拉式菜单。运行程序，然后单击“文件”按钮，将会弹出菜单选项，如图 1-1 所示。

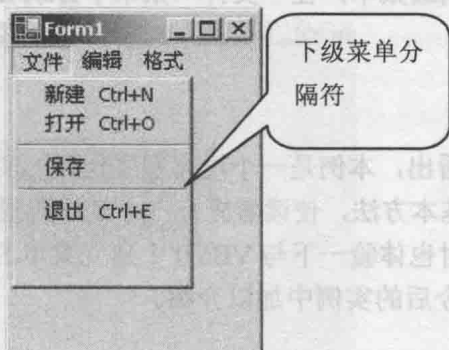


图 1-1 建立主菜单和弹出菜单

技术要点

1. 使下级菜单间有分隔，用鼠标在两个所需要分开的下级菜单中的第 2 个下级菜单上单击右键，在弹出的菜单选项中选择“插入分隔符”，如图 1-2 所示。

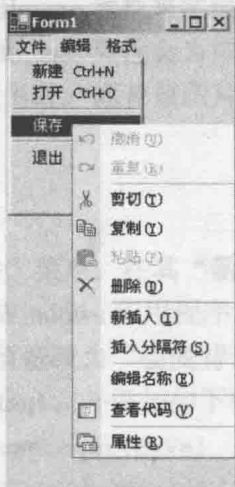


图 1-2 在菜单中插入分隔符

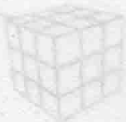
2. 下级菜单建立快捷键，用鼠标在所需要建立快捷键的下级菜单上单击右键，在弹出的菜单中选择属性，并在其属性栏中找到 Shortcut 选项，在其选项里选择相应的快捷方式，这样就可为下级菜单建立相应的快捷方式。

STEP 实现步骤

1. 新建一个项目，单击“文件”菜单，在其“新建”选项中选中“项目”项，并在弹出提示框的模板栏里选择“Windows 应用程序”，此时，出现了一个窗体 Form1。
2. 在 Form1 中增加一个菜单控件 MainMenu（确定你已经打开了工具箱。打开工具箱的方法是点击视图菜单，再点击工具箱菜单。此控件在 Windows 窗体中）。
3. 此时，在 Form1 的菜单栏位置会出现“请在此处输入”的提示性文字，然后在里面输入所需要的菜单名称，例如“文件”。
4. 在新建菜单下方或右方会出现下一级菜单或同级菜单的输入提示，如果需要建立下一级菜单或同级菜单，在“文件”菜单下方或右方的提示处输入下级或同级的菜单名称即可。

归纳注释

通过该实例可以看出，本例是一个没有程序代码的实例，这主要是为了让读者掌握一些 VB.NET 建立菜单的基本方法，使读者先打一个良好的根基，对如何在 VB.NET 中建立菜单有个相应的了解，同时也体验一下与 VB6.0 中建立菜单的不同之处。至于菜单项目中具体功能的实现，我们会在今后的实例中加以介绍。



实例 2

会变背景图片的按钮

实例说明

本例主要说明如何实现会变背景图片按钮的功能。运行程序，把鼠标放在按钮上，按钮会自动变换图片，该实例执行的结果如图 2-1 所示。

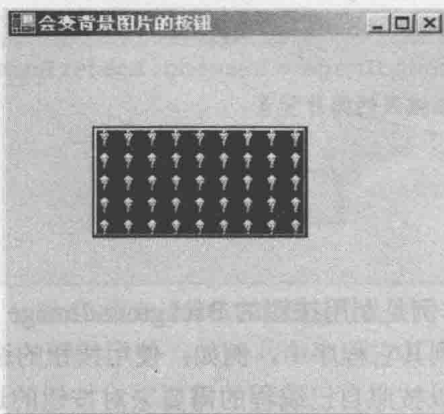


图 2-1 会变背景图片的按钮

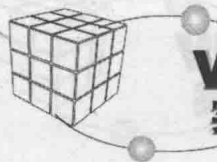
技术要点

- 本例利用按钮的 `BackgroundImage` 属性来实现按钮图片的转变，当鼠标移动进入按钮区域时按钮的 `BackgroundImage` 属性会为按钮设置一幅背景图片，当鼠标移出按钮区域时属性会为按钮设置另外一幅背景图片。`BackgroundImage` 属性对按钮图片的变换有着重要的作用。

实现步骤

1. 新建一个项目，单击“文件”菜单，在其“新建”选项中选项“项目”项，并在弹出提示框的模板栏里选择“Windows 应用程序”，此时，已经出现了一个窗体 `Form1`。
2. 将窗体 `Form1` 的 `Text` 属性值修改为“会变背景图片的按钮”。
3. 在窗体上增加 1 个按钮 `Button1`。并添加如下代码（两个独立的函数）：

```
Private Sub Button1_MouseEnter(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.MouseEnter
    Dim mypath As String
    mypath = Application.StartupPath
```



Visual Basic.NET

实用编程100例

```
Button1.BackgroundImage = Drawing.Image.FromFile(mypath & "\LIGHTOn.ico") '该按钮装载灯点亮的图片背景
```

```
End Sub
```

```
Private Sub Button1_MouseLeave(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.MouseLeave
```

```
Dim mypath As String
```

```
mypath = Application.StartupPath
```

```
Button1.BackgroundImage = Drawing.Image.FromFile(mypath & "\LIGHTOFF.ico")
```

```
End Sub
```

4. 在窗体 Form1 中添加如下代码:

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
```

```
Dim mypath As String
```

```
mypath = Application.StartupPath
```

```
Button1.BackgroundImage = Drawing.Image.FromFile(mypath & "\LIGHTOFF.ico") '该按钮装载灯熄灭的图片背景
```

```
End Sub
```



归纳注释

通过该实例可以看出,本例是利用按钮的 **BackgroundImage** 属性来改变按钮的背景图片,我们也可以将这个属性运用到其它程序中,例如:使用按钮的这个属性做一个按钮图片循环切换的程序,当然读者也可以按照自己编程的需要来对按钮的这个属性进行应用。

图 2-1 改变背景图片的实例

要点提示

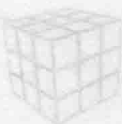
本例利用按钮的 **BackgroundImage** 属性来改变按钮图片的切换,当鼠标移动到按钮区域时按钮的 **BackgroundImage** 属性会被设置为另一张图片,当鼠标移出按钮区域时属性会被设置为另外一张背景图片。**BackgroundImage** 属性对按钮图片的切换有着重要的作用。

操作步骤

1. 新建一个项目,单击“项目”菜单,在其“新建”子菜单“窗体”项下,单击并拖入窗体 Form1。
2. 将窗体 Form1 的 **Text** 属性值修改为“改变背景图片的按钮”。
3. 在窗体上增加 1 个按钮 Button1,并添加如下代码(两个独立的函数)。

```
Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Dim mypath As String
    mypath = Application.StartupPath
    Button1.BackgroundImage = Drawing.Image.FromFile(mypath & "\LIGHTOn.ico")
End Sub

Private Sub Button1_MouseLeave(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.MouseLeave
    Dim mypath As String
    mypath = Application.StartupPath
    Button1.BackgroundImage = Drawing.Image.FromFile(mypath & "\LIGHTOFF.ico")
End Sub
```



实例 3

获取计算机名称

实例说明

本例主要说明如何获得计算机的名称。运行程序后，将显示该计算机的用户名，实例执行的结果如图 3-1 所示。

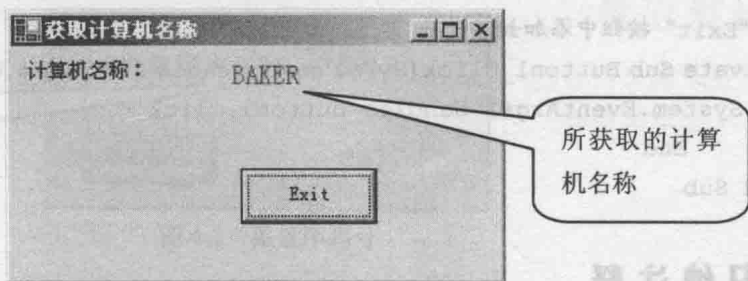


图 3-1 获取计算机名称的运行结果

技术要点

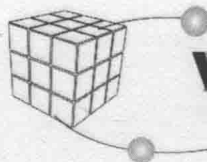
- 本例使用了 API 函数中的 `getComputerName` 函数来返回一个指定路径中包含与系统名称相对应的值，其返回的值也就是当前计算机的名称，然后再将这个值赋于所需的变量，最后该计算机的系统名称会显示到当前窗体。

实现步骤

1. 新建一个项目，单击“文件”菜单，在其“新建”选项中选中“项目”项，并在弹出提示框的模板栏里选择“Windows 应用程序”，此时，已经出现了一个窗体 Form1。
2. 将窗体 Form1 的 Text 属性值修改为“获取计算机名称：”。
3. 在窗体 Form1 上增加 1 个标签框 Label1，并将其 Text 属性值修改为“计算机名称：”。
4. 在窗体 Form1 上增加 1 个标签框 Label2，并将其 Text 属性值修改为“”。
5. 在窗体 Form1 的代码中加入以下代码，以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编程过程中，经常需要调用系统提供的一些 API 函数，这样大大提高了编程速度。后面的例子中也用到了类似的函数，声明方法与此类似。

```
Private Declare Function GetComputerName Lib "kernel32" Alias
"GetComputerNameA" (ByVal lpBuffer As String, ByRef nSize As Integer) As
Integer
```

6. 在窗体 Form1 的代码中加入一个函数 `getName()`，其返回值为当前计算机名称：



Visual Basic.NET

实用编程100例

```

Function getName() As String
    Dim l1 As String
    Dim l2 As Integer
    Dim l3 As Integer

    l2 = 255
    l1 = New String(" ", l2)
    l3 = GetComputerName(l1, l2)
    getname = ""
    If l3 <> 0 Then
        getname = Trim(l1)
    End If
End Function

```

7. 在窗体 Form1 中增加 1 个按钮 Button1, 并将其 Text 属性值修改为“Exit”。

在“Exit”按钮中添加如下代码:

```

Private Sub Button1_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Button1.Click
    End
End Sub

```



归纳注释

该实例利用 API 函数中的 `getComputerName` 函数来获取一个与计算机的系统名称相对应的值, 并且将所得到的值显示到当前窗体。读者通过对该实例的了解可以更深入的去认识 API 函数, 该实例中的 `getComputerName` 函数仅仅为一个简单的 API 函数, 目的是为了读者先由浅处着手对 API 函数有一个大概的了解, 以后再循序渐进的深入了解。

1. 在窗体 Form1 中增加 1 个按钮 Button1, 并将其 Text 属性值修改为“Exit”。
2. 在窗体 Form1 的代码中加入以下代码, 以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编写过程中, 经常要用到系统提供的 API 函数, 这些函数的名字通常位于 `kernel32.dll` 中。
3. 在窗体 Form1 的代码中加入以下代码, 以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编写过程中, 经常要用到系统提供的 API 函数, 这些函数的名字通常位于 `kernel32.dll` 中。
4. 在窗体 Form1 的代码中加入以下代码, 以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编写过程中, 经常要用到系统提供的 API 函数, 这些函数的名字通常位于 `kernel32.dll` 中。
5. 在窗体 Form1 的代码中加入以下代码, 以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编写过程中, 经常要用到系统提供的 API 函数, 这些函数的名字通常位于 `kernel32.dll` 中。
6. 在窗体 Form1 的代码中加入以下代码, 以调用 `kernel32.dll` 中提供的函数 `getComputerName()`。在编写过程中, 经常要用到系统提供的 API 函数, 这些函数的名字通常位于 `kernel32.dll` 中。

实例 4

查看磁盘序列号程序

实例说明

本例说明如何取得磁盘序列号。运行程序，单击“磁盘序列号”按钮，窗体将显示磁盘的序列号，该实例执行的结果如图 4-1 所示。

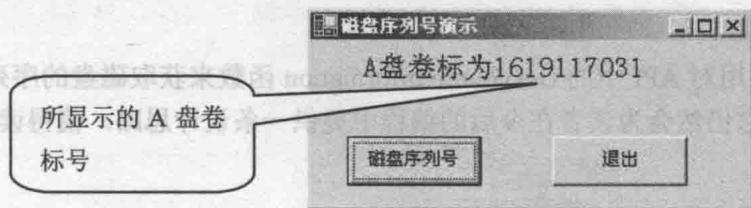


图 4-1 磁盘序列号

技术要点

- 本例利用了 API 函数中的 GetVolumeInformation 函数实现该实例功能。磁盘序列号在每次软盘或硬盘格式化后都重新生成，并且不会重复。许多程序员用此加密。其实也可以通过修改该函数，得到磁盘卷标和文件系统类型信息。

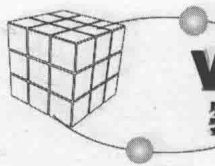
实现步骤

1. 新建一个项目，单击“文件”菜单，在其“新建”选项中选中“项目”项，并在弹出提示框的模板栏里选择“Windows 应用程序”，此时，已经出现了一个窗体 Form1。
2. 将窗体 Form1 的 Text 属性值修改为“磁盘序列号演示”。
3. 添加如下声明函数，以取得磁盘卷标信息：

```
Private Declare Function GetVolumeInformation Lib "kernel32.dll" Alias  
"GetVolumeInformationA" (ByVal lpRootPathName As String, ByVal  
lpVolumeNameBuffer As String, ByVal nVolumeNameSize As Short, ByRef  
lpVolumeSerialNumber As Integer, ByRef lpMaximumComponentLength As Integer,  
ByRef lpFileSystemFlags As Integer, ByVal lpFileSystemNameBuffer As String,  
ByVal nFileSystemNameSize As Integer) As Integer
```

4. 在窗体上新建 1 个标签框 Label1 和 2 个按钮 Button1, Button2。将标签框 Label1 的 Text 属性里的默认文字 Label1 删除，将 2 个按钮 Button1 和 Button2 的 Text 属性值分别修改为“磁盘序列号”和“退出”。

(1) 在“磁盘序列号”按钮中添加如下代码：



Visual Basic.NET

实用编程100例

```
Private Sub Button1_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Button1.Click
    Dim i As Object
    i = GetSerialNumber("a:\")
    Labell.Text = "A 盘卷标为" & CStr(i)
End Sub
```

(2) 在“退出”按钮中添加如下代码:

```
Private Sub Button2_Click(ByVal eventSender As System.Object, ByVal
eventArgs As System.EventArgs) Handles Button2.Click
    End
End Sub
End Class
```



归纳注释

通过该实例可以看出,利用对 API 中的 `GetVolumeInformation` 函数来获取磁盘的序列号,虽然是一个简单的程序,但它仍然会为读者在今后的编程中提供一条设计思路,值得读者仔细参考。

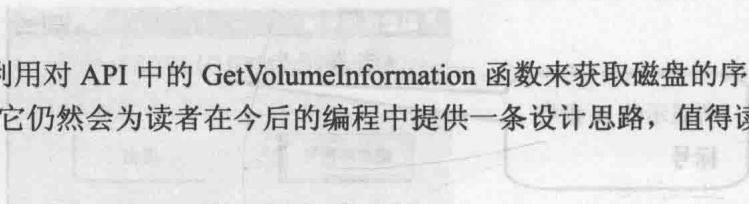


图 14-1 序列号

- (1) 在“磁盘序列号”按钮中添加如下代码。
- 将窗体 Form1 的 Text 属性修改为“磁盘序列号”。
- 添加如下代码,以取得磁盘卷标信息。
- 在窗体上添加 1 个标签框 Label1 和 2 个按钮 Button1、Button2。将标签框 Label1 的 Text 属性里的默认文字删除。将 2 个按钮 Button1 和 Button2 的 Text 属性分别修改为“磁盘序列号”和“退出”。