

时尚百例丛书



Visual Basic.NET

时尚编程百例



●网冠科技 编著



100

时尚百例丛书

Visual Basic.NET

时尚编程百例

网冠科技 编著

光盘包含本书系列、效果文件



机械工业出版社

Visual Basic.NET 是 Microsoft 公司最新推出的 Visual Studio.NET 组件。

本书不是对 Visual Basic.NET 的各种控件进行简单的罗列,而是运用了大量的 ActiveX 控件制作实例来讲解 Visual Basic.NET。

本书通过 100 个实例,全面讲解 Visual Basic.NET 的编程方法和技巧。内容包括:如何使用控件、如何制作控件、如何对图片进行处理、如何对文件进行操作。

本书适用于中级程序设计人员。

图书在版编目(CIP)数据

Visual Basic.NET 时尚编程百例 / 网冠科技编著.

—北京:机械工业出版社,2002.5

(时尚百例丛书)

ISBN 7-111-10410-2

I. V … II. 网… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2002)第 037774 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划:胡毓坚

责任编辑:汪汉友

责任印制:付方敏

三河市宏达印刷有限公司印刷 • 新华书店北京发行所发行

2002 年 7 月第 1 版 • 第 1 次印刷

787mm × 1092mm $\frac{1}{16}$ • 19.25 印张 • 473 千字

0001-6000 册

定价:35.00 元(1CD)

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话:(010) 68993821、68326677-2527

封面无防伪标均为盗版

时尚百例丛书

追求时尚 追求完美

出版说明

随着 21 世纪的到来,人们更深切地感受到了计算机在生活和工作中的作用越来越重要,越来越多的职业需要具有计算机的应用技能。掌握计算机是职业的需要,更是事业发展的需要。

目前计算机技术不但广泛地应用在办公自动化中,它还全面渗透到各行各业。如果要从平面设计的相关行业,就应该学会平面设计软件,如 Photoshop、CorelDRAW、FreeHand 等;如果要从三维设计的相关行业,就应该学会三维设计软件,如 3DS MAX、Maya、Poser 等;如果要从多媒体设计的相关行业,就应该学会多媒体制作软件,如 Authorware、Director、Premiere 等;如果要从与网络相关的行业,就应该学会 Flash、Dreamweaver、Fireworks、ASP、PHP、JavaScript 等;如果要从建筑产品、工业产品设计的相关行业,就应该学会 AutoCAD、3DS VIZ、Protel 等;如果要从软件开发的相关行业,就应该学会 VB、VC、VFP、Delphi、PowerBuilder 等编程。

所有与计算机相关的职业都要求工作者有很强的计算机操作技能,做到运用自如,熟练而且深入地掌握软件的应用。而要做到这一点,必须从软件的各个方面入手,通过实例演练的方式训练自己,而且要反复练习,做到举一反三。

为了让大家能深入而且熟练地掌握相关软件的应用方法,机械工业出版社特别为广大读者推出了这套时尚百例丛书。本丛书对每一个应用软件精心制作了 100 个实例,其宗旨就是让读者全方位掌握软件的应用,为广大读者提供一条快速掌握计算机应用技能的捷径。

本丛书采用新颖的版式,将知识和实例紧密结合,通过对各种实例的详细讲解,使读者不必事先学习各种软件,而从实例的制作过程中体会到每个软件每项功能的使用方法,并自己做出各种实例效果,这样既节省了大量时间,同时也使读者有身临其境的感觉,并可以反复演练,将所学知识运用到职业工作中去。

书山有路勤为径。愿广大读者能通过本丛书的学习掌握计算机技能,并应用到自己的工作和事业中去。

机械工业出版社



前 言

《Visual Basic.NET 时尚编程百例》是“时尚百例丛书”中的一本。

在短短的几年内，随着组建对象技术的不断进步，以及 Internet 应用的不断普及，Visual Basic 也不断进步。Microsoft 公司最新推出的 Visual Basic.NET 是围绕 .NET 框架设计更新的。

Visual Basic.NET 有许多改进的语言特征，例如继承、接口和重载。这些特征使得 Visual Basic.NET 成为一种强大的面向对象的编程语言。Visual Basic.NET 的其他特征包括结构化异常处理、定制标志和通用语言兼容性。Visual Basic.NET 中添加了新特性以利用 CLS 的优势。使得所有与 CLS 兼容的语言都可以使用由 Visual Basic.NET 创建的类、组件和对象。Visual Basic.NET 也能访问由其他 CLS 兼容语言创建的类、组件和对象。

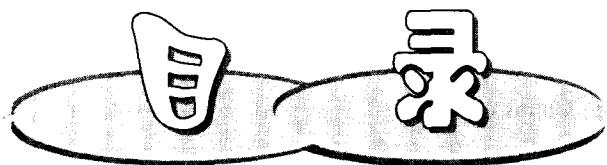
本书以实例的形式向读者讲解如何使用 Visual Basic.NET。本书共分为七篇，第一篇制作一些基于窗体的实例，第二篇是关于文件和文字的应用，第三篇对位图和图片的应用，第四篇应用 ActiveX 控件，第五篇控件制作，第六篇网络控件的应用，第七篇其他一些算法。

相信通过本书 100 个实例的学习，可以使程序设计者更好地编写程序。真诚希望我们的讲解对您有所帮助。



网冠科技

本书光盘含配套素材（使用方法请见光盘中“光盘使用说明书”），技术支持请点击网冠科技站点 Netking.163.com。E-mail: Netking_@yeah.net。



出版说明 前言

第一篇 用户界面

实例 1	文字从左至右显示	2
实例 2	冒泡排序	5
实例 3	查找字符串	8
实例 4	简单的运算	11
实例 5	金额大写转换	14
实例 6	彩票投注器	17
实例 7	移动的标题文字	20
实例 8	字符串比较	22
实例 9	特殊的窗口关闭	25
实例 10	文字上卷	27
实例 11	循环移动的文字	30
实例 12	删除文件中的空行	33
实例 13	变色的超链接	36
实例 14	将阿拉伯数字转换为中文大写	39
实例 15	图片界面	42
实例 16	程序启动界面	44
实例 17	寻找质数	47
实例 18	计算 MHR	50
实例 19	计算执行程序的时间	53
实例 20	十进制转换为二进制	55

第二篇 文字和文件处理

实例 21	自动存盘	59
实例 22	打开文本文件	61
实例 23	记事本	64
实例 24	改变记事本颜色	67
实例 25	压缩文件	69



实例 26	滚动的文字	72
实例 27	翻转字符串	74
实例 28	写文件	77
实例 29	文件读入到 Chunks	80
实例 30	扫描文件	83
实例 31	创建文件夹	86
实例 32	分解单词	88
实例 33	文件的保存	91
实例 34	显示 SQL 语句	94
实例 35	在 ListBox 中查找	97
实例 36	文字和二进制的转换	100
实例 37	将 ListBox 中的内容写入文件	103
实例 38	判断文件是否存在	106
实例 39	MP3 播放器	109
实例 40	文字由大变小	112
实例 41	加密的文件	115
实例 42	文件解码	118

第三篇 位图 和 图像处理

实例 43	浏览 GIF 动画	122
实例 44	ICO 文件制作器	125
实例 45	淡入淡出效果	128
实例 46	模拟画图板	131
实例 47	BMP 格式转换为 GIF 格式	133
实例 48	绘制贝塞尔曲线	136
实例 49	图片编辑器	138
实例 50	调色板	142

第四篇 控件 使用

实例 51	注册表程序	146
实例 52	天气预报	149
实例 53	交通工具类	152
实例 54	计算器	155
实例 55	时速表	158
实例 56	拷贝进程动画	161
实例 57	二维和三维图形控件	164
实例 58	多媒体播放器	167

实例 59	DVD 播放器	170
实例 60	监测电池电量	173
实例 61	演示斐波纳契数列	176
实例 62	井字游戏	179
实例 63	打开对话框	183
实例 64	保存对话框	186
实例 65	颜色对话框	189
实例 66	红绿灯程序	192
实例 67	倒数计时	195
实例 68	软件安装	198
实例 69	写日志文件	201
实例 70	跳动的时间	204

第五篇 控件制作

实例 71	第一个 ActiveX 控件	208
实例 72	显示网络文件控件	211
实例 73	路径确定控件	214
实例 74	计算文件行数控件	217
实例 75	计时控件	220
实例 76	显示所有四位数的组合	223
实例 77	特制的 TextBox 控件	226
实例 78	动画按钮控件	228
实例 79	闪烁文本框控件	230
实例 80	验证 IP 地址正确性	232

第六篇 网络

实例 81	下载网页	236
实例 82	html 文件浏览器	239
实例 83	发邮件程序	242
实例 84	获取超级连接	245
实例 85	拨号上网程序	248

第七篇 其他

实例 86	字符串处理	252
-------	-------------	-----

实例 87	单位转换	255
实例 88	绝对时间	258
实例 89	十进制与十六进制的转换	260
实例 90	小数转换为分数	263
实例 91	启动其他程序	266
实例 92	图片的不同转换效果	269
实例 93	制作 JPG 文件	272
实例 94	图片特殊显示效果	275
实例 95	数字转换为英文显示	278
实例 96	数字与时间的转换	281
实例 97	逻辑运算器	284
实例 98	查询日期	287
实例 99	计算相隔天数	290
实例 100	跑表程序	293

第一篇

用户界面

本篇总览

本篇主要应用 Visual Basic.NET 的一些基本控件制作比较简单的界面。

通过本篇中各个实例的制作，读者可以对 Visual Basic.NET 中基本控件的属性有一个大概的了解，并可以了解到 Visual Basic.NET 对界面的处理简单和功能齐全的特点。

实例 1 文字从左至右显示

实例说明

本例制作一个文字从左至右显示效果的程序。运行效果如图 1-1 所示。

程序运行后，在 TextBox2 中会将 TextBox1 中的文字从左至右的显示。

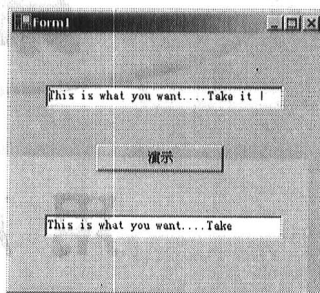


图 1-1 效果图

编程思路

本实例应用工具箱中的 TextBox、Timer 控件，并运用 Timer 事件和 Left() 函数使文字从左至右显示。本实例利用该控件的 Tick 属性，循环逐个向后从上面文本框中取字符，并显示在下面的文本框中，并每隔一个时间间隔刷新一下下面的文本框。

创作步骤

一、对话框的建立

1. 启动 Visual Basic.NET，打开一个新的项目。如果 Visual Basic.NET 已经运行，单击“文件”→“新建”→“项目”菜单项，打开一个新的项目。
2. 单击工具箱中的 TextBox 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将它属性中的名称设置为“TextBox1”，将 Text 设置为“This is what you want....Take it !”。
3. 单击工具箱中的 Timer 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将它属性中的 Interval 设置为 100，将 Enabled 属性设为 True。
4. 单击工具箱中的 TextBox 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将它属性中的名称设置为“TextBox2”，将 Text 设置为空。

二、程序代码

```
Public Class Form1
Inherits System.Windows.Forms.Form

Dim a As String
Dim t As String
Dim b As Integer
```

```
Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
```

```
Dim I As Integer
```

```
#Region " Windows Form Designer generated code "
```

```
Public Sub New()
```

```
    MyBase.New()
```

```
    InitializeComponent()
```

```
End Sub
```

```
Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
```

```
    If disposing Then
```

```
        If Not (components Is Nothing) Then
```

```
            components.Dispose()
```

```
        End If
```

```
    End If
```

```
    MyBase.Dispose(disposing)
```

```
End Sub
```

```
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
```

```
Friend WithEvents Button1 As System.Windows.Forms.Button
```

```
Friend WithEvents Timer1 As System.Windows.Forms.Timer
```

```
Private components As System.ComponentModel.IContainer
```

```
<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
```

```
    Me.components = New System.ComponentModel.Container()
```

```
    Me.TextBox1 = New System.Windows.Forms.TextBox()
```

```
    Me.Timer1 = New System.Windows.Forms.Timer(Me.components)
```

```
    Me.TextBox2 = New System.Windows.Forms.TextBox()
```

```
    Me.Button1 = New System.Windows.Forms.Button()
```

```
    Me.SuspendLayout()
```

```
    Me.TextBox1.Location = New System.Drawing.Point(32, 48)
```

```
    Me.TextBox1.Name = "TextBox1"
```

```
    Me.TextBox1.Size = New System.Drawing.Size(224, 21)
```

```
    Me.TextBox1.TabIndex = 0
```

```
    Me.TextBox1.Text = "This is what you want....Take it !"
```

```
    Me.Timer1.Enabled = True
```

```
    Me.TextBox2.Location = New System.Drawing.Point(32, 168)
```

```
    Me.TextBox2.Name = "TextBox2"
```

```
    Me.TextBox2.Size = New System.Drawing.Size(224, 21)
```

```
    Me.TextBox2.TabIndex = 2
```

```
    Me.TextBox2.Text = ""
```

```
    Me.Button1.Location = New System.Drawing.Point(80, 104)
```

```
    Me.Button1.Name = "Button1"
```

```

Me.Button1.Size = New System.Drawing.Size(120, 24)
Me.Button1.TabIndex = 1
Me.Button1.Text = "演示"
Me.AutoScaleBaseSize = New System.Drawing.Size(6, 14)
Me.ClientSize = New System.Drawing.Size(292, 237)
Me.Controls.AddRange(New System.Windows.Forms.Control() { Me.TextBox2, Me.Button1,
Me.TextBox1 })
Me.Name = "Form1"
Me.Text = "Form1"
Me.ResumeLayout(False)
End Sub
#End Region

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
    CheckAgain() '调用函数
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
    CheckAgain() '调用函数
End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Timer1.Tick
    t = Microsoft.VisualBasic.Left(a, b) '取得字符串a左边的第b个字符
    TextBox2.Text = t
    b = b + 1
    If b > I Then b = 0
End Sub

Sub CheckAgain()
    a = TextBox1.Text '取得字符串a左边的第b个字符
    I = Len(a)
    b = 0
End Sub
End Class

```

三、运行程序

按〈F5〉键或Visual Basic.NET工具栏中的“运行”按钮，编译并运行程序。

实例2 冒泡排序

实例说明

本例制作一个冒泡排序的程序。运行效果如图 2-1 所示。

程序运行后，单击“冒泡排序”按钮，程序会将 5 个文本框中的数字根据冒泡排序法进行排序，并显示出排序结果。

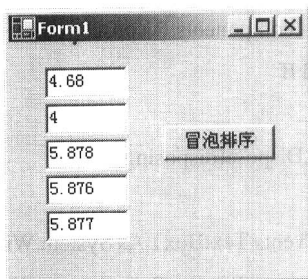


图 2-1 效果图

编程思路

本实例应用工具箱中的 TextBox 控件和 Button 控件，本实例通过按钮触发排序程序，对数字进行排序并显示排序结果。

创作步骤

一、对话框的建立

1. 启动 Visual Basic.NET，打开一个新的项目。如果 Visual Basic.NET 已经运行，单击“文件”→“新建”→“项目”菜单项，打开一个新的项目。
2. 单击工具箱中的 TextBox 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，同理添加 5 个控件，单击将它属性中的 Text 分别设置为“4.68”、“4”、“5.878”、“5.876”、“5.877”。
3. 单击工具箱中的 Button 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将它属性中的 Text 设置为“冒泡排序”。

二、程序代码

```
Public Class Form1
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "
    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call
    End Sub
```

'Form overrides dispose to clean up the component list.

Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)

 If disposing Then

 If Not (components Is Nothing) Then

 components.Dispose()

 End If

 End If

 MyBase.Dispose(disposing)

End Sub

Friend WithEvents TextBox1 As System.Windows.Forms.TextBox

Friend WithEvents TextBox2 As System.Windows.Forms.TextBox

Friend WithEvents TextBox3 As System.Windows.Forms.TextBox

Friend WithEvents TextBox4 As System.Windows.Forms.TextBox

Friend WithEvents TextBox5 As System.Windows.Forms.TextBox

Friend WithEvents Button1 As System.Windows.Forms.Button

'Required by the Windows Form Designer

Private components As System.ComponentModel.Container

'NOTE: The following procedure is required by the Windows Form Designer

'It can be modified using the Windows Form Designer.

'Do not modify it using the code editor.

<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()

 Me.TextBox1 = New System.Windows.Forms.TextBox()

 Me.TextBox2 = New System.Windows.Forms.TextBox()

 Me.TextBox3 = New System.Windows.Forms.TextBox()

 Me.TextBox4 = New System.Windows.Forms.TextBox()

 Me.TextBox5 = New System.Windows.Forms.TextBox()

 Me.Button1 = New System.Windows.Forms.Button()

 Me.SuspendLayout()

 "TextBox1

 Me.TextBox1.Location = New System.Drawing.Point(24, 16)

 Me.TextBox1.Name = "TextBox1"

 Me.TextBox1.Size = New System.Drawing.Size(56, 21)

 Me.TextBox1.TabIndex = 0

 Me.TextBox1.Text = "4.68"

 "TextBox2

.....此处代码略，详见光盘。

Me.TextBox4, Me.TextBox3, Me.TextBox2, Me.TextBox1))

 Me.Name = "Form1"

 Me.Text = "Form1"

```

        Me.ResumeLayout(False)
    End Sub
#End Region

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
    Dim Array1(5) As Double
    Array1(1) = CDb1(Text1.Text)
    Array1(2) = CDb1(Text2.Text)
    Array1(3) = CDb1(Text3.Text)
    Array1(4) = CDb1(Text4.Text)
    Array1(5) = CDb1(Text5.Text)
    Call BubbleSort(Array1)
    Text1.Text = CStr(Array1(1)) '显示结果
    Text2.Text = CStr(Array1(2))
    Text3.Text = CStr(Array1(3))
    Text4.Text = CStr(Array1(4))
    Text5.Text = CStr(Array1(5))
End Sub

Sub BubbleSort(ByVal List() As Double)
    Dim First As Double, Last As Double
    Dim i As Integer, j As Integer
    Dim Temp As Double
    First = LBound(List) '取得数组上界
    Last = UBound(List) '取得数组下界
    For i = First To Last - 1 '冒泡排序法
        For j = i + 1 To Last
            If List(i) > List(j) Then
                Temp = List(j)
                List(j) = List(i)
                List(i) = Temp
            End If
        Next j
    Next i
End Sub
End Class

```

三、运行程序

按〈F5〉键或 Visual Basic.NET 工具栏中的“运行”按钮，编译并运行程序。

实例 3 查找字符串

实例说明

本例制作一个查找字符串的程序。运行效果如图 3-1 所示。

程序运行后，在文本框中输入字符串，单击“查找”按钮，弹出 InputBox。输入字符串中任意字符，程序返回该字符右边的所有字符。



图 3-1 效果图

编程思路

本实例应用工具箱中的 TextBox 控件和 Button 控件，主要应用“Right”等一些字符串函数来实现得到字符串中某一个字符右边的字符。

Right (string, length)，string 代表字符串表达式，其最右边的字符被返回。length 是数值表达式，指明返回的字符数目。如果是 0，返回零长度字符串“”；如果大于或等于 string 参数中的所有字符数目，则返回整个字符串。

Mid (string, start[, length])，string 是字符串表达式，从中返回字符。Start 表示 string 中被提取的字符部分的开始位置。如果 start 超过了 string 中字符的数目，Mid 将返回零长度字符串“”。length 表示要返回的字符数。如果省略或 length 超过文本的字符数（包括 start 处的字符），将返回字符串中从 start 到字符串结束的所有字符。

创作步骤

一、对话框的建立

1. 启动 Visual Basic.NET，打开一个新的项目。如果 Visual Basic.NET 已经运行，单击“文件”→“新建”→“项目”菜单项，打开一个新的项目。
2. 单击工具箱中的 TextBox 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将 Text 属性设置为空。
3. 单击工具箱中的 Button 控件，将该控件用单击加拖动的方法添加到窗口中，在添加的过程中同时调整好该控件的坐标和大小，单击将它属性中的 Text 设置为“查找”。

二、程序代码

```
Public Class Form1
```

```
Inherits System.Windows.Forms.Form
```