

Visual Basic .NET 2005 中文版

经典案例设计与实现

孙更新 周峰 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

内容简介

Visual Basic.NET 2005

中文版经典案例设计与实现

孙更新 周 峰 编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书采用几十个经典案例详细讲解如何使用Visual Basic.NET 2005进行开发和应用的方法和技巧。其中,实例的选取特别考虑到Visual Basic.NET 2005的新特性。本书的作者融合了多年实际开发中遇到的各种常见问题和编程技巧,着重于Visual Basic.NET 2005技术的实用性和可操作性,并结合Visual Basic.NET 2005技术的核心问题,针对目前读者的实际需要进行了内容结构的设计和分类,涉及的内容从高级控件的使用方法到关于多媒体处理技术,从网络编程到ADO.NET数据库开发,从Windows应用程序到Web应用程序,几乎涵盖了Visual Basic.NET 2005中的所有核心技术。

本书适合初、中级用户,对高级程序员也有一定参考价值,并可作为社会上Visual Basic.NET 2005培训班的教材。本书涉及到不同类型的应用,多角度地引导读者学习相关知识,通过本书的学习,读者可以迅速提高自己的编程水平,达到实际商业开发的要求。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Visual Basic.NET 2005中文版经典案例设计与实现/孙更新,周峰编著.—北京:电子工业出版社,2008.6
ISBN 978-7-121-06354-1

I. V… II. ①孙… ②周… III. Basic语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2008)第047370号

责任编辑:徐云鹏

特约编辑:底波

印刷:北京天竺颖华印刷厂

装订:三河市金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

北京市海淀区翠微东里甲2号 邮编:100036

开本:787×1092 1/16 印张:28 字数:710千字

印次:2008年6月第1次印刷

定 价:50.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至zlbs@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

本书通过实例分析的方式讲解了Visual Basic.NET 2005中各项核心技术开发和应用的方法和技巧,使读者在看完本书后,就能够结合实际,快速、高效、灵活地设计出专业级的VB.NET应用程序来。本书着重于VB.NET技术的实用性和可操作性,书中所有的案例可以直接应用到以后的实际开发中,从而使读者能学到真正的符合实际开发要求的技能。

本书结构

本书共10章,分别讲解了Visual Basic.NET 2005中的各种核心技术的应用:

- 第1章讲解Windows程序设计。
- 第2章讲解Visual Basic.NET 2005中的数据库程序设计。
- 第3章主要介绍Visual Basic.NET 2005中文件系统的应用。
- 第4章讲解Visual Basic.NET 2005中图形图像程序设计。
- 第5章主要介绍Visual Basic.NET 2005中多媒体技术的应用。
- 第6章讲解Web程序设计。
- 第7章讲解Visual Basic.NET 2005中XML程序设计。
- 第8章讲解Visual Basic.NET 2005中网络程序设计。
- 第9章讲解Visual Basic.NET 2005中的一些高级技术应用。
- 第10章讲解Visual Basic.NET 2005中的游戏开发技术。

本书特色

本书的特色归纳如下:

- 实用性

本书首先着眼于应用程序的实用性,然后通过具体实例再探讨深层次的技巧问题。

- 详尽的例子

本书中附有大量的例子,通过对这些例子所涉及到的知识点的详细介绍,使读者能学到真正的实战本领。

- 全面性

本书几乎包含了VB.NET的所有功能,特别是详细讲解了Visual Basic.NET 2005中新增的各项功能。

本书适合的读者

本书不仅适用于各种层次的大中专院校学生、Visual Basic.NET 应用程序开发人员以及编程爱好者,并且对于软件开发的高级专业人士也有很高的参考价值,尤其适合作为社

会上Visual Basic.NET培训班的教材。

以下人员对本书的编写提出过宝贵意见并参与了本书部分资料搜集的工作，他们是孙宁、王荣芳、李德路、李岩、周科峰、陈勇、高云、于凯、王春玲、李永杰、韩亚男、陈卓、王伟、姚国发，感谢北京美迪亚电子信息有限公司的各位老师，谢谢你们的帮助和指导。

由于时间仓促，加之水平有限，书中的缺点和不足之处在所难免，敬请读者批评指正。

本书特色

- 第1章讲解Windows程序设计。
- 第2章讲解Visual Basic.NET 2005中的数据源程序。
- 第3章主要介绍Visual Basic.NET 2005中文件系统的操作。
- 第4章讲解Visual Basic.NET 2005中窗体窗口的操作。
- 第5章主要介绍Visual Basic.NET 2005中多媒体技术的应用。
- 第6章讲解Web程序设计。
- 第7章讲解Visual Basic.NET 2005中XML程序设计。
- 第8章讲解Visual Basic.NET 2005中网络程序设计。
- 第9章讲解Visual Basic.NET 2005中的一些高级技术应用。
- 第10章讲解Visual Basic.NET 2005中的高级开发技术。

本书特色

本书的特色如下：
• 实用性。

本书首先着眼于应用案例，然后通过具体的案例来讲解相关的理论知识。

• 循序渐进。

本书中每个章节都包含大量的例子，通过这些例子来讲解相关的理论知识，使读者能够循序渐进地学习。

• 本书配套的源代码。

为了方便读者阅读，本书配套资料请登录“华信教育资源网” (<http://www.hxedu.com.cn>)，在“资源下载”频道的“图书资源”栏目下载。

本书特色

本书不仅适用于各种层次的大中专院校计算机专业，而且对于从事软件开发的高级专业技术人员也有较高的参考价值。本书特色如下：
• 实用性。

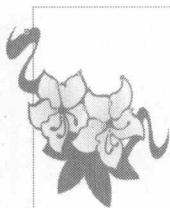
目 录

第1章 Windows程序设计	1
案例1 转换VB 6窗体到VB.NET	1
案例2 开发一个MDI应用程序	5
案例3 使用菜单	11
案例4 使用ListView控件	14
案例5 网页浏览器程序设计	17
案例6 清空回收站	19
案例7 对话框的综合应用	22
案例8 TrackBar与ProgressBar控件的使用	25
案例9 计算器	28
案例10 椭圆形状的窗体	33
案例11 继承窗体的设计	35
案例12 渐显的窗体	38
案例13 时间管理程序	39
本章小结	47
第2章 数据库程序设计	48
案例1 使用ADO.NET控件查询SQL Server数据库	48
案例2 使用ADO.NET控件查询Access数据库	55
案例3 连接环境下的参数化查询	59
案例4 ADO.NET执行存储过程	65
案例5 使用事务机制	68
案例6 使用连接环境读写SQL Server数据库	70
案例7 使用非连接环境读写SQL Server数据库	75
案例8 数据的过滤与排序	80
案例9 使用Crystal Report	83
本章小结	89
第3章 文件系统程序设计	90
案例1 文本文件的读写操作	90
案例2 二进制文件的读写操作	94

案例3	文件及目录的操作	96
案例4	在VB.NET中检索文件信息	100
案例5	用VB.NET实现记事本程序	102
案例6	使用VB.NET加密文件	118
案例7	在VB.NET中监视文件夹的变化	120
案例8	综合实例——Windows资源管理器	123
	本章小结	134
第4章	图形图像程序设计	135
案例1	使用画笔	135
案例2	绘制线图	140
案例3	绘制其他图形	144
案例4	递归图案	150
案例5	绘制填充图形	153
案例6	调节图像色彩	155
案例7	拾色器	158
案例8	滤镜效果	159
	本章小结	162
第5章	多媒体技术	163
案例1	MP3播放器	163
案例2	视频播放器	170
案例3	Flash播放器	175
案例4	综合实例——多媒体播放程序	179
	本章小结	187
第6章	Web程序设计	188
案例1	Repeater控件的使用	188
案例2	DataList控件的使用	194
案例3	GridView控件的使用	198
案例4	使用GridView编辑和删除数据	204
案例5	XML数据显示和TreeView控件	210
案例6	使用验证控件	214
	本章小结	223

第7章 XML程序设计	224
案例1 使用XmlTextReader类的对象读取XML文档	224
案例2 根据节点属性获取节点类型信息	226
案例3 使用XmlNodeReader对象读取XML文档	231
案例4 利用XmlValidatingReader对象执行DTD验证	234
案例5 利用XmlValidatingReader对象执行xsd验证	236
案例6 使用XmlTextWriter创建XML文档	239
案例7 使用DOM浏览XML文档	240
案例8 使用DOM操作XML文档——添加节点	251
案例9 使用DOM操作XML文档——修改和删除节点	258
案例10 使用DATASET读/写XML及XML模式	265
案例11 创建嵌套的XML数据	273
案例12 使用XslTransform对象转换XML文档	277
案例13 使用XsltArgumentList对象向样式表传递参数	280
案例14 利用扩展对象扩展样式表	303
案例15 序列化和反序列化	307
案例16 序列化SOAP消息	313
本章小结	316
第8章 网络程序设计	317
案例1 发送和接收电子邮件	317
案例2 使用Socket实现文件传输	323
案例3 网络订餐订单接收处理系统	325
案例4 基于Socket的网络聊天程序	333
本章小结	339
第9章 高级应用	340
案例1 VB.NET多线程开发实例	340
案例2 VB.NET实现PTM查看器	345
案例3 线程管理程序	351
案例4 调用Windows API函数	361
案例5 在VB.NET中读写INI文件	364
案例6 验证邮件地址的合法性	365
案例7 获取CPU和硬盘的信息	370
案例8 注册表管理程序	375
案例9 获得bios信息	381

334	案例10 打印预览	383
334	案例11 打印机程序	387
336	案例12 精灵	394
331	本章小结	398
334	第10章 游戏编程	399
339	案例1 贪食蛇游戏	399
340	案例2 打字游戏	411
351	案例3 迷宫寻宝游戏	424
358	本章小结	439
365	案例10 使用DataSet操作XML数据	450
373	案例11 创建嵌套的XML数据	455
377	案例12 使用XslTransform对象转换XML文件	460
380	案例13 使用XslArgumentList对象向转换函数传递参数	465
383	案例14 利用XPath对象选择XML节点	470
387	案例15 利用XPath对象选择XML节点	475
393	案例16 利用SOAP消息	480
398	本章小结	495
399	第8章 网络编程设计	399
399	案例1 发送和接收电子邮件	405
399	案例2 使用Socket类进行文件传输	410
399	案例3 网络订票订票系统	415
399	案例4 基于Socket的网络聊天程序	420
399	本章小结	425
399	第9章 高级应用	399
399	案例1 VB.NET之线程开发	405
399	案例2 VB.NET实现PTM查看器	410
399	案例3 线程管理程序	415
399	案例4 调用Windows API函数	420
399	案例5 在VB.NET中读写INI文件	425
399	案例6 验证邮件地址的合法性	430
399	案例7 获取CPU和硬盘的信息	435
399	案例8 注册表管理程序	440
399	案例9 获得bios信息	445



第1章

Windows程序设计

本章内容

本章的重点内容是使用VB.NET 2005开发Windows应用程序，通过对各实例的分析，读者可以掌握使用VB.NET 2005开发Windows应用程序中的一些常用的技巧。

- 转换VB 6窗体到VB.NET
- 使用菜单
- 网页浏览器程序设计
- 对话框的综合应用
- 计算器
- 继承窗体的设计
- 时间管理程序
- 开发一个MDI应用程序
- 使用ListView控件
- 清空回收站
- TrackBar与ProgressBar控件的使用
- 椭圆形状的窗体
- 渐显的窗体



案例1 转换VB 6窗体到VB.NET

VB.NET改变了开发人员在VB 6中常用的编程方法，VB 6的GUI对开发人员隐藏大部分内部工作，但是在VB.NET中每一个外观设计使用的都是开放GUI控件。VB.NET语言对于VB来讲有了一些变化，如果我们要把VB 6应用程序转换为VB.NET，那么绝大部分工作将会是重新逐个创建所有VB 6中的窗体。在微软的VB.NET 2005中提供了VB 6应用程序到VB.NET应用程序的转换工具。

步骤详解

例如对于以下这个VB 6应用程序：

功能：输出不同维数的动态数组的值。

设计步骤如下：

(1) 在窗体上添加三个标签控件、三个文本框控件和两个命令按钮。窗体及控件的运行效果如图1-1所示。注意，Text1和Text3的Multiline属性设置为True。Text3的ScrollBars属性设置为2-Vertical。

(2) 编写代码。

```
Dim a() As Integer '定义动态数组a()类型为Integer
'对单击命令按钮“输出数组”的事件编写代码
Private Sub Command1_Click()
    Dim i, j As Integer
    ReDim a(10) '在过程中将动态数组a()定义为一维数组，大小为11
```

```

For i = 0 To 10
    a(i) = i
    p = p & Str(a(i))
Next i
ReDim a(2, 3) '在过程中将动态数组a()定义为二维数组大小为3*4
For i = 0 To 2
    For j = 0 To 3
        a(i, j) = i + j
        Text2.Text = Text2.Text & Str(a(i, j))
    Next j
    Text2.Text = Text2.Text & Chr(13) & Chr(10)
Next i
Text1.Text = p
ReDim c(2, 2, 2) '在过程中直接定义动态数组c(2, 2, 2)为三维数组, 大小为3*3*3
For i = 0 To 2
    For j = 0 To 2
        For k = 0 To 2
            c(i, j, k) = i + j + k
            Text3.Text = Text3.Text & c(i, j, k) & " "
        Next k
        Text3.Text = Text3.Text & Chr(13) & Chr(10)
    Next j
    Text3.Text = Text3.Text & Chr(13) & Chr(10)
Next i
End Sub
Private Sub Command2_Click()
Unload Me
End Sub

```

保存本案例后, 会生成两个文件: 窗体文件examx1-1.frm和工程文件examx1-1.vbp。

下面, 我们使用转换工具将这个VB 6应用程序转换为VB.NET应用程序, 步骤如下:

第一步: 在工程文件examx1-1.vbp上右键单击, 在弹出的菜单上选择“打开方式”, 将出现如图1-2所示的提示窗口。

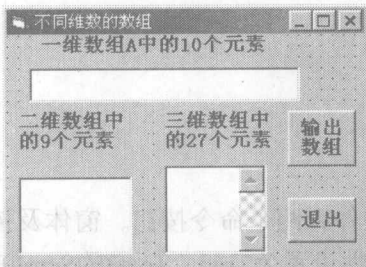


图1-1 “不同维数的数组”工程设计界面

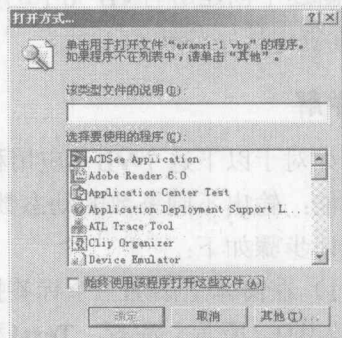


图1-2 “打开方式”提示窗口

第二步: 在“打开方式”提示窗口中选择“Microsoft Visual Studio 2005”, 即使用VB.NET 2005打开此应用程序, 将出现如图1-3所示的提示窗口。

第三步: 设置升级向导。

单击“下一步”, 将显示如图1-4所示的提示窗口。

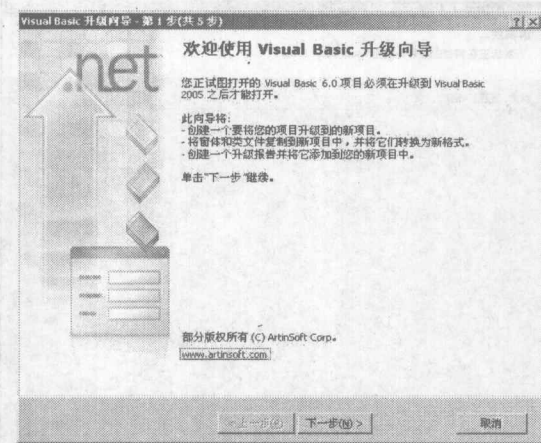


图1-3 “升级向导”窗口

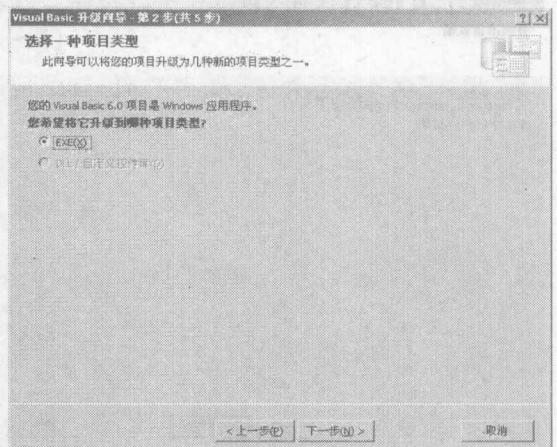


图1-4 “选择一种项目类型”提示窗口

因为我们转换的是Windows应用程序，所以使用默认的选项即可，单击“下一步”，将显示如图1-5所示的提示窗口。

单击“下一步”，将显示如图1-6所示的消息框。

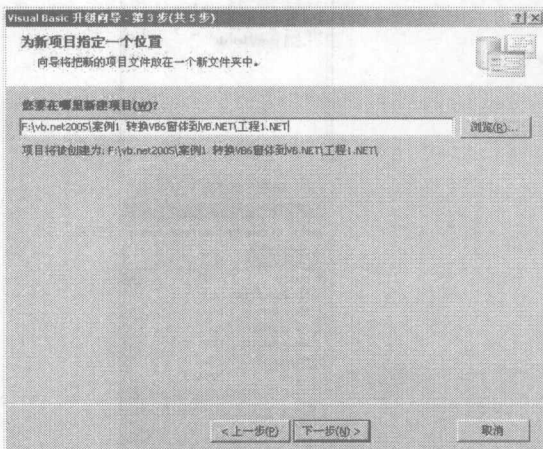


图1-5 “指定位置”提示窗口

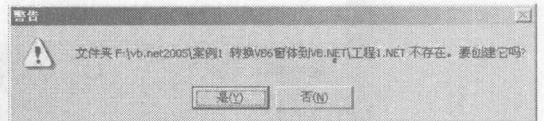


图1-6 提示消息框

选择“是”，将显示如图1-7所示的提示窗口。

单击“下一步”，系统就将进行转换，显示如图1-8所示的提示窗口。

转换成功后，就会将此VB 6应用程序在VB.NET 2005中打开，如图1-9所示。

制作要点

我们需要注意，VB.NET有许多和旧版本的Visual Basic一样的语法，但是在许多方面，它是一门全新的语言。和VB先前的版本不一样，VB 6的语法与VB.NET的语法稍有不同，因此，转换过后的应用程序不一定能够直接运行，需要对代码进行修改后才可以正常执行。

下面我们介绍VB 6的语法与VB.NET的语法的不同之处。

1. 过去的Long变成了Integer，过去的Integer变成了Short。VB.NET中Long可容纳64位。

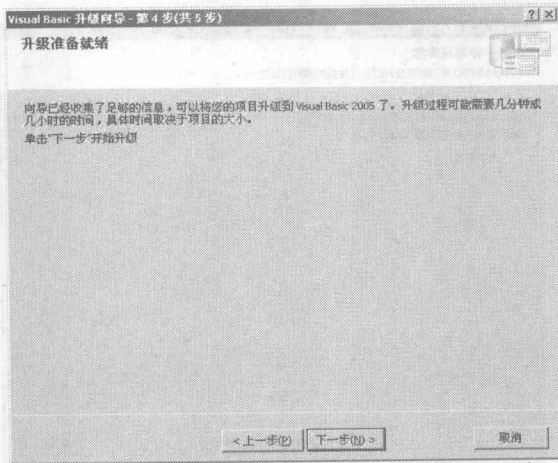


图1-7 “准备升级”提示窗口

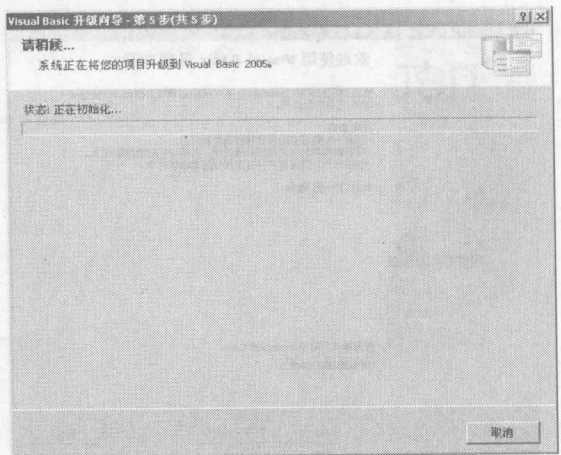


图1-8 转换进度图

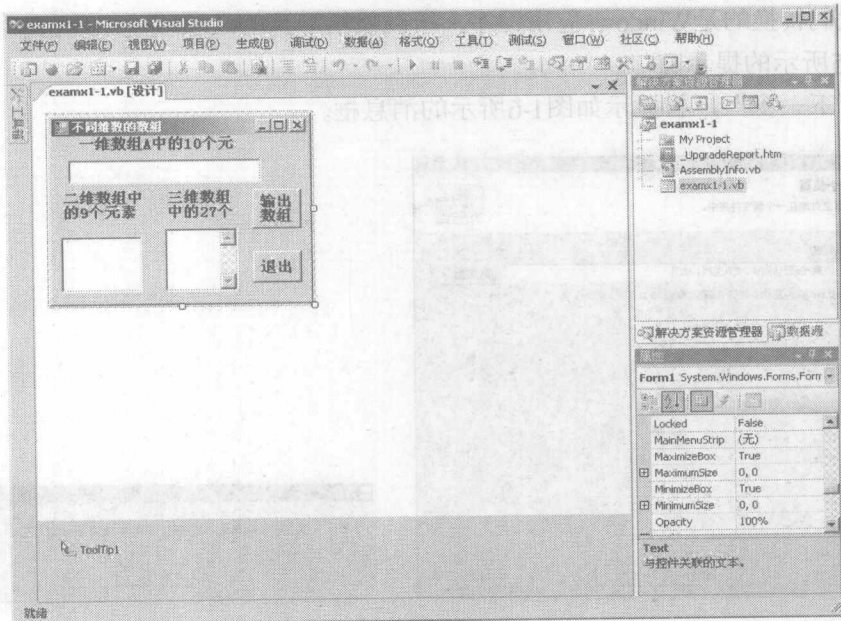


图1-9 转换成功结果图

2. VB.NET引入了一个名为Char的新的数据类型。它占用2字节，并可以存放Unicode字符。
3. Variants消失了，代替它的是Object数据类型。
4. Currency数据类型不再存在，它被更强大的128位Decimal（十进制）数据类型所代替。
5. 新的操作符：VB.NET带来了一些新的数学操作符，从而缩减了代码。如VB.NET中的“X+=4”与VB 6中的“X=X+4”作用相同。
6. 错误处理：Try、Catch和Finally的结构中，如果发生了错误，运行Try区内的代码，Catch区内的代码也被运行。不管发生什么，Finally区总是要运行的。
7. 不再有一个简单的Form.Show方法。VB.NET中的一切都是对象，因此我们需要实际的“Dim FormName As New Form1”，然后执行一个“FormName.Show”。

8. **Garbage Collection** (垃圾回收): 当操作系统 (.NET框架) 认为该清除对象引用一类的东西时就会运行**Garbage Collection**程序。

9. **Return**关键字: 在一个函数内部, 我们不向返回值或对象设置函数, 而是简单地声明“**Return MyData**”, 这样就允许不修改实际代码即可改变函数名。

10. 在VB 6中用**Type**关键字来声明用户定义的结构。在VB.NET中用**Structure**关键字来达到相同的目的, 其余部分的语法是不变的。

11. 声明变量。

让我们看一个在VB6中的简单例子:

```
Dim x,y as integer
```

在这个例子中, VB 6会认为x是一个**variant**变量, 而y是一个**integer**变量, 这是个有点古怪的语法现象。VB.NET纠正了这个问题, 将x和y都创建为**integer**变量。此外, VB.NET允许我们在声明一个变量的同时赋予它初始值, 如:

```
Dim str1 as string="hello"
```

VB.NET还引入了只读 (**Read-Only**) 变量。与常数变量不同的是只读变量可以在声明的时候不赋予它初始值, 但是一旦我们给它赋了值, 它就不能再改变了, 例如:

```
Dim readonly x as integer
```

12. 数组。

在VB 6中, 可以定义数组的下界和上界。但在VB.NET中, 数组的下界总是零。



案例2 开发一个MDI应用程序

步骤详解

1. 单击屏幕左下角的“开始”, 选择“程序/Microsoft Visual Studio 2005/Microsoft Visual Studio 2005”, 启动Microsoft Visual Studio 2005集成开发环境, 如图1-10所示。

2. 新建项目。

单击“文件”菜单中的“新建项目”, 将出现如图1-11所示的提示窗口。

注意, 因为我们所创建的是Windows应用程序, 所以选择的是第一个模板 (同默认设置)。将项目名称改为**FirstMdiApp**, 如图1-12所示。

单击“确定”后, 系统将进入设计界面, 如图1-13所示。

我们可以看到, 当新建一个Windows应用程序时, 系统会自动新建一个名为**Form1**的初始窗体, 自动生成的代码中也使用**Form1**。如果修改了这个名字, 则需要手动修改代码, 并且需要重新设置项目的启动对象, 否则项目将不能正常启动。

3. 添加窗体。

单击“项目”菜单中的“添加Windows窗体”, 系统将弹出如图1-14所示提示窗口。

后来添加的窗体默认名字为**Form2**、**Form3**, 依此类推。单击“添加”按钮, 即可为项目再次添加一个新窗体。至此为止, **FirstMdiApp**项目中一共有两个窗体, 名称分别为**Form1**和**Form2**。

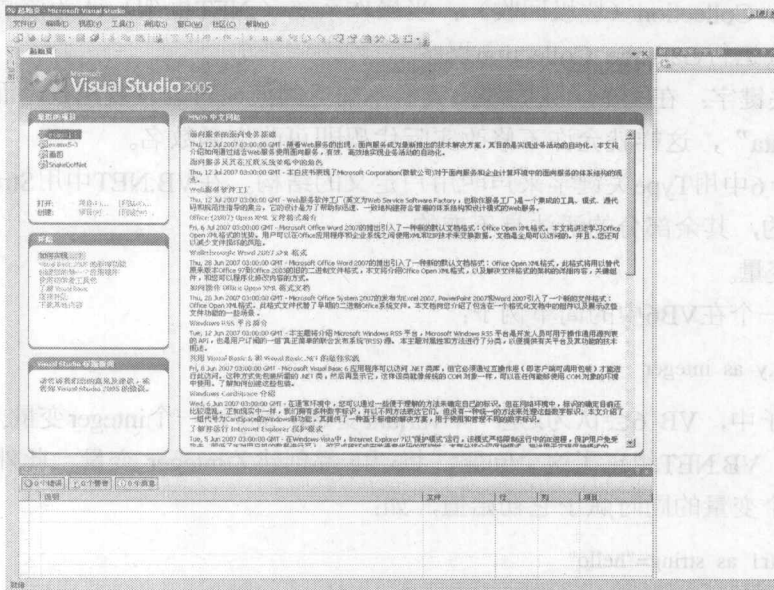


图1-10 “Microsoft Visual Studio 2005”起始页

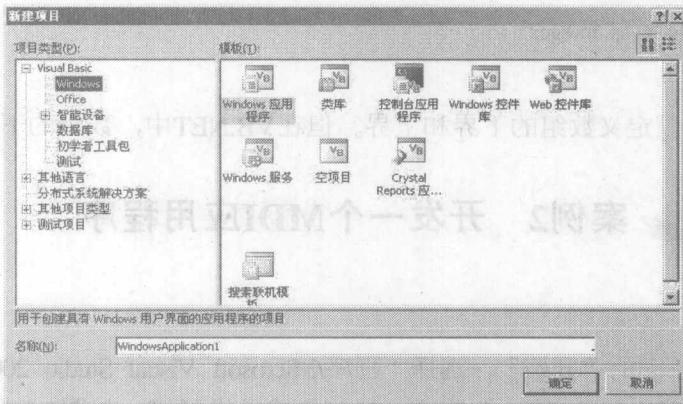


图1-11 “新建项目”提示窗口

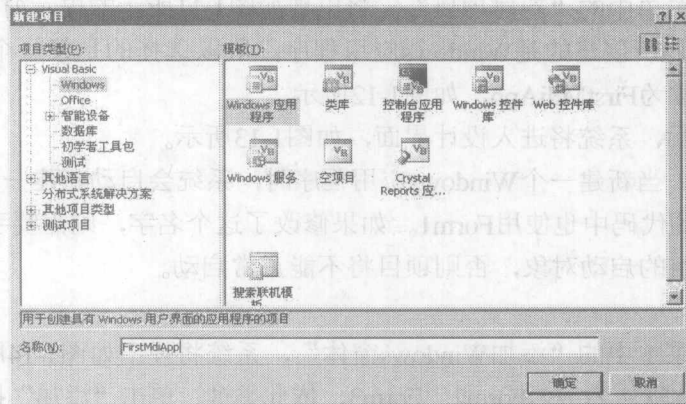


图1-12 设定项目名称

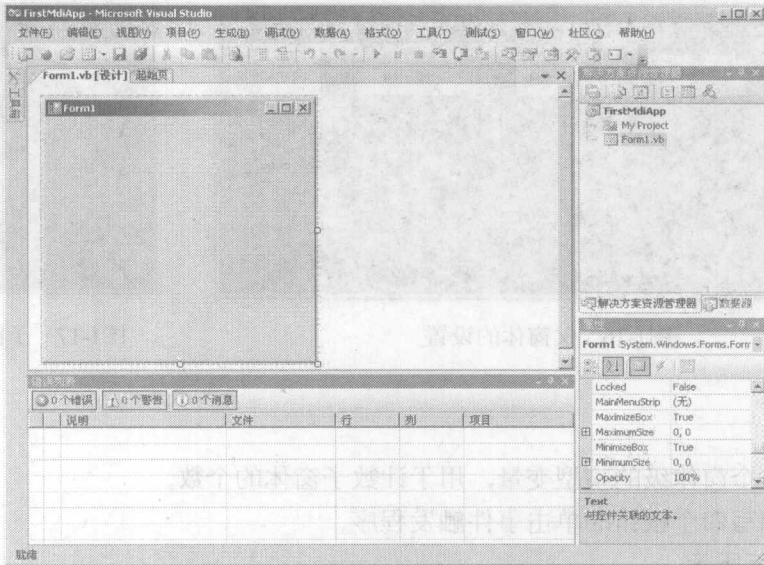


图1-13 设计界面

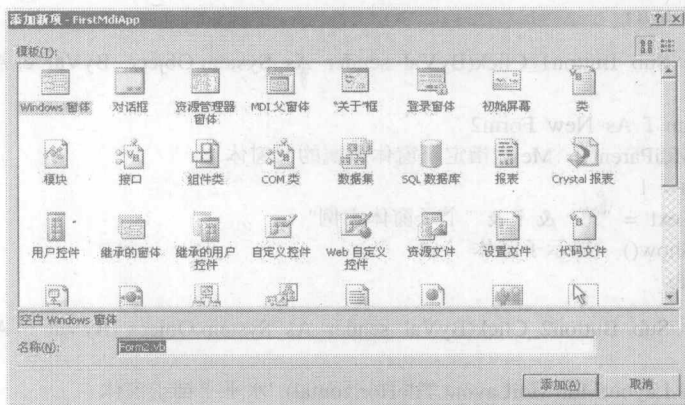


图1-14 “添加Windows窗体”提示窗口

4. 设置父窗体和子窗体。

我们令Form1为父窗体，Form2为子窗体。下面分别对这两个窗体进行设置。

对于Form1：

(1) 我们将窗体的IsMdiContainer属性设置为True，即将窗体指定为MDI父窗体，此时，窗体的背景色变为深灰色，如图1-15所示。

(2) 将WindowState属性设置为Maximized，因为当父窗体最大化时操作MDI子窗口最为容易。

(3) 在其上放置四个按钮，并设置按钮及窗体的Text属性及大小和位置，如图1-16所示。

对于Form2：我们将其Text属性改为“子窗体实例”，并将此窗体的背景色（BackColor属性）设置为红色，如图1-17所示。

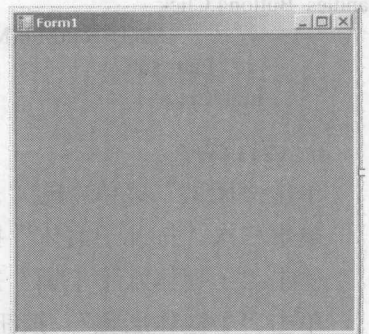


图1-15 MDI父窗体

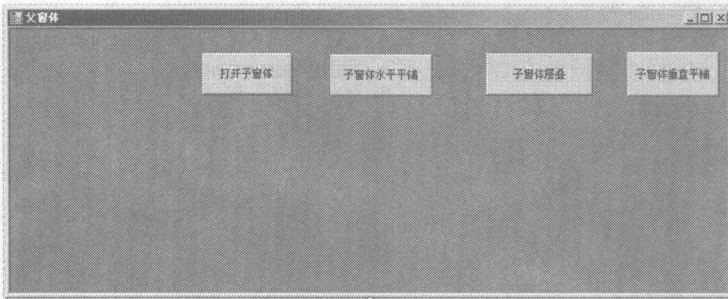


图1-16 父窗体的设置

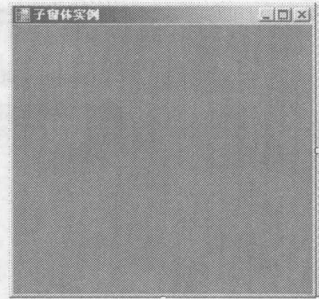


图1-17 子窗体的设置

5. 编写代码。

对于Form1:

- (1) 声明一个窗体级的整型变量，用于计数子窗体的个数。
- (2) 分别编写四个按钮的单击事件触发程序。

整体代码如下所示:

```
Public Class Form1
    Dim i As Integer

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        Dim f As New Form2
        f.MdiParent = Me '指定子窗体所属的父窗体
        i += 1
        f.Text = "第" & i & "个子窗体实例"
        f.Show() '显示子窗体
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
        Me.LayoutMdi(MdiLayout.TileHorizontal) '水平平铺子窗体
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button3.Click
        Me.LayoutMdi(MdiLayout.Cascade) '子窗体层叠排列
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button4.Click
        Me.LayoutMdi(MdiLayout.TileVertical) '垂直平铺子窗体
    End Sub
End Class
```

6. 运行程序。

单击“调试”菜单中的“启动调试”，运行程序，初始界面如图1-18所示。

单击三次“打开子窗体”按钮，程序显示如图1-19所示。

单击“子窗体水平平铺”按钮，程序显示如图1-20所示。

单击“子窗体层叠”按钮，程序显示如图1-21所示。

单击“子窗体垂直平铺”按钮，程序显示如图1-22所示。