

21st CENTURY  
规划教材

面向21世纪高等学校计算机系列规划教材

COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

# Visual Basic

## 程序设计实践指导与习题

吴定雪 主编



科学出版社

[www.sciencep.com](http://www.sciencep.com)

TP312/2831A

2008

· 面向 21 世纪高等学校计算机系列规划教材 ·

# Visual Basic 程序设计

## 实践指导与习题

吴定雪 主编

科学出版社

北京

## 内 容 简 介

本书是《Visual Basic 程序设计》的配套实验教材,全书包括四个部分:上机指导部分介绍 Visual Basic 集成开发环境的基本操作、使用与设置等;实验部分针对教材各章节中的内容,精心设计相应的上机实验内容;课程设计部分精心选择两个设计课题(学生成绩管理系统、图书借阅系统);习题精选部分所精选的习题在形式和内容上都与教育部全国计算机等级考试的形式和内容贴近,有助于学生巩固所学的知识。

本书可作为本科学校计算机专业及理工科类非计算机专业学生学习 Visual Basic 程序设计的实验教材,还可作为有关工程技术人员和计算机爱好者学习 Visual Basic 程序设计的参考书。

### 图书在版编目(CIP)数据

Visual Basic 程序设计实践指导与习题/吴定雪主编.——北京:科学出版社,2008

(面向 21 世纪高等学校计算机系列规划教材)

ISBN 978-7-03-020266-6

I.V… II.吴… III.BASIC 语言—程序设计—高等学校—教学参考资料 IV.TP312

中国版本图书馆 CIP 数据核字(2008)第 021146 号

责任编辑:张颖兵 吉正霞/责任校对:梅莹

责任印制:董艳辉/封面设计:苏波

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

武汉中科兴业印务有限公司印刷

科学出版社发行 各地新华书店经销

\*

2008 年 2 月第 一 版 开本:16(787×1092)

2008 年 2 月第 1 次印刷 印张:13 1/4

印数:1—5 000 字数:299 000

定价:21.50 元

(如有印装质量问题,我社负责调换)

# 《Visual Basic 程序设计实践指导与习题》

## 编委名单

主 编 吴定雪

副主编 郭峰林

编 委 (按姓氏笔画为序)

王转利 朱泽民 关玉蓉 汤恒耀

杨改贞 肖 飞 吴定雪 余志超

陈晓林 周 芬 郭峰林 崔艳莉

## 前 言

Visual Basic 是微软公司在原有 Basic 语言的基础上进一步发展而来的,其既保持了原有 Basic 简单易学的特点,又可以可视化的方式实现图形用户界面的开发,具有强大的软件开发功能。Visual Basic 是目前开发 Windows 应用程序最快捷的工具与使用人数最多的程序设计语言。目前 Visual Basic 正在被越来越多的高校选定 Visual Basic 作为非计算机专业学习程序设计的第一语言,作为非计算机专业大学生必修的计算机程序设计课程,而且随着教育部全国计算机等级考试的逐步推进,也有越来越多的考生希望通过计算机等级考试二级 Visual Basic 考试,获得相应的计算机等级证书。

“Visual Basic 程序设计”是一门实践性很强的课程,针对实用型、应用型人才培养特点,我们根据多年从事计算机程序设计 Visual Basic 实践教学的经验,秉承“学得会、用得上”的宗旨,编写上充分考虑语言的特点、程序设计课程的教学规律、读者群等,注意内容的紧凑、结构的循序渐近等,采用实验教学中行之有效的“实验目标——知识结构——实践应用”组织结构,精心设计每一个实验实例、实验内容,编写了这本《Visual Basic 程序设计实践指导与习题》。实践证明,这更能激发读者学习 Visual Basic 程序设计的兴趣,培养学生的实际编程能力。

全书包括 Visual Basic 上机指导、实验部分、课程设计与习题精选等四部分。其中,上机指导部分主要分析了 Visual Basic 程序设计的特点,以帮助初学者建立起 Visual Basic 程序设计特有的思维方式,介绍了 Visual Basic 集成开发环境的基本操作,向系统请求帮助,Visual Basic 程序调试、排错与错误处理方法,以及一些 Visual Basic 常见程序错误。

实验内容部分由 12 个实验组成,针对教材各章节中的内容,精心设计和安排了相应的上机实验内容,每个实验都给出了具体的操作步骤、设计要求、运行界面与详细的操作步骤以及详细的分析和参考源代码,并对实验中所涉及的所有重点与难点,都给出了适当的提示,这不但有利于初学者尽快掌握必备的知识,同时还有利于学生的进一步学习和提高。

课程设计部分,精心选择安排了两个设计课题(学生成绩管理系统、图书借阅系统),每个课题都给出了功能介绍、具体要求、详细的设计步骤和各个功能模块的参考源代码,这对学生系统、综合地应用 Visual Basic 课程所学过的各部分知识,提高应用开发能力,起到了较大的促进和帮助作用。

习题精选部分,所精选的习题在形式和内容上都与教育部全国计算机等级考试的形式和内容贴近,有助于学生复习、巩固所学的知识,为参加计算机等级考试二级 Visual Basic 考试的读者提供一些帮助。

本书可作为应用型 and 教学型本科学校计算机专业及理工科类非计算机专业学生学习 Visual Basic 程序设计的教材,还可供有关工程技术人员和计算机爱好者学习 Visual Basic 程序设计的参考书。

本书由在教学第一线并具有丰富计算机程序设计教学经验的多位教师共同编写。全书由吴定雪任主编,郭峰林、朱泽民任副主编,最后由吴定雪主审与统稿,其中第一部分上机指导和第二部分实验一由余志超编写;第二部分实验二、实验三、实验六由杨改贞编写;第二部分实验四、实验八、实验九由崔艳莉编写;第二部分实验五、实验十一由王转利编写;第二部分实验七、实验十由关玉蓉编写;第二部分实验十二由陈晓林编写、汤恒耀校对;第三部分课程设计由朱泽民、肖飞编写;第四部分习题精选由周芬、朱泽民编写。

本书在编写过程中得到了学校各级领导和教务处以及计算机科学与技术学院全体教师的大力支持,在此表示衷心感谢。还要感谢科学出版社的各级领导和编辑们对我们编写的教材的精心策划、组织和编辑!同时,也对编写过程中所参阅的大量文献的作者致以谢意。限于作者学识水平,研究工作还不够深入,本书难免有疏漏和不妥之处,诚请读者和同行批评指正。

我们的电子邮件地址:computer@hgnc.net

网站地址:<http://www.hgnc.net/yuanxi/jsjx/>

# 目 录

<b>第一部分 上机指导</b> .....	1
1.1 Visual Basic 环境配置 .....	1
1.2 Visual Basic 编程规范 .....	1
1.2.1 命名规范 .....	1
1.2.2 Visual Basic 代码书写格式规范 .....	5
1.2.3 养成良好的编程习惯 .....	5
1.2.4 什么是好的代码 .....	6
1.3 应用程序调试与错误处理 .....	6
1.3.1 错误的种类 .....	7
1.3.2 程序的三种操作模式 .....	8
1.3.3 Visual Basic 调试工具 .....	9
1.4 程序常见错误 .....	13
1.5 发行应用程序 .....	14
1.5.1 设置工程的版本信息 .....	14
1.5.2 发布应用程序 .....	14
1.5.3 运行安装程序 .....	18
1.5.4 卸载应用程序 .....	19
1.6 联机帮助系统 .....	19
1.6.1 在线帮助 .....	19
1.6.2 上下文相关帮助 .....	20
1.7 全国计算机等级考试上机考试说明 .....	20
1.7.1 基本要求 .....	20
1.7.2 内容提要 .....	21
1.7.3 上机考试说明 .....	23
1.7.4 考试题型分析 .....	23
<b>第二部分 实验指导书</b> .....	25
<b>实验一 创建简单的 Visual Basic 应用程序</b> .....	25
一、实验目的与要求 .....	25
二、实验准备 .....	25
三、实验案例 .....	25
四、问题与思考 .....	31
五、课外实验 .....	31
<b>实验二 数据的输入与输出</b> .....	31
一、实验目的与要求 .....	31
二、实验准备 .....	31
三、实验案例 .....	32

四、问题与思考 .....	37
五、课外实验 .....	37
实验三 选择结构程序设计 .....	37
一、实验目的与要求 .....	37
二、实验准备 .....	37
三、实验案例 .....	37
四、问题与思考 .....	45
五、课外实验 .....	45
实验四 循环结构程序设计 .....	45
一、实验目的与要求 .....	45
二、实验准备 .....	45
三、实验案例 .....	45
四、问题与思考 .....	55
五、课外实验 .....	55
实验五 常用控件 .....	56
一、实验目的与要求 .....	56
二、实验准备 .....	56
三、实验案例 .....	56
四、问题与思考 .....	63
五、课外实验 .....	63
实验六 数组程序设计 .....	63
一、实验目的与要求 .....	63
二、实验准备 .....	63
三、实验案例 .....	63
四、问题与思考 .....	72
五、课外实验 .....	72
实验七 过程程序设计 .....	72
一、实验目的与要求 .....	72
二、实验准备 .....	73
三、实验案例 .....	73
四、问题与思考 .....	82
五、课外实验 .....	82
实验八 鼠标与键盘事件过程 .....	82
一、实验目的与要求 .....	82
二、实验准备 .....	82
三、实验案例 .....	83
四、问题与思考 .....	85
五、课外实验 .....	85
实验九 Visual Basic 界面程序设计 .....	85
一、实验目的与要求 .....	85
二、实验准备 .....	85
三、实验案例 .....	86



四、问题与思考 .....	91
五、课外实验 .....	91
实验十 文件操作 .....	91
一、实验目的与要求 .....	91
二、实验准备 .....	92
三、实验案例 .....	92
四、问题与思考 .....	97
五、课外实验 .....	97
实验十一 高级控件应用 .....	98
一、实验目的与要求 .....	98
二、实验准备 .....	98
三、实验案例 .....	98
四、问题与思考 .....	104
五、课外实验 .....	104
实验十二 数据库的建立与使用 .....	104
一、实验目的与要求 .....	104
二、实验准备 .....	105
三、实验案例 .....	105
四、问题与思考 .....	109
五、课外实验 .....	109
第三部分 课程设计 .....	110
3.1 课程设计基础 .....	110
3.2 学生成绩管理系统 .....	113
3.2.1 设计目的 .....	113
3.2.2 功能需求 .....	113
3.2.3 设计过程 .....	113
3.2.4 系统完善与扩展 .....	115
3.2.5 程序参考代码 .....	115
3.3 图书借阅系统 .....	120
3.3.1 设计目的 .....	120
3.3.2 功能需求 .....	120
3.3.3 设计过程 .....	120
3.3.4 系统完善与扩展 .....	123
3.3.5 程序参考代码 .....	124
第四部分 习题精选 .....	132
习题1 概述 .....	132
习题2 程序设计基础 .....	134
习题3 数据的输入与输出 .....	136
习题4 控制结构设计 .....	138
习题5 常用控件 .....	143
习题6 数组 .....	149

习题 7 过程 .....	154
习题 8 鼠标与键盘事件 .....	158
习题 9 应用程序界面设计 .....	160
习题 10 文件 .....	161
习题 11 上机练习 .....	163
<b>参考文献</b> .....	<b>183</b>
<b>附录 1 习题精选参考答案</b> .....	<b>184</b>
<b>附录 2 教材课后习题参考答案</b> .....	<b>186</b>

# 第一部分 上机指导

## 1.1 Visual Basic 环境配置

### 1. Visual Basic 的运行环境

操作系统: Windows NT Server 4.0, Windows NT Workstation 4.0, Windows 98, 或更高版本。

CPU: 486DX/66 MHz 或更高的处理器。

内存: Windows 95/98 要求 24MB 以上的内存, Windows NT 要求 32MB 以上。

磁盘空间: 对于标准版或专业版, 典型安装 48MB, 完全安装 80MB。

### 2. Visual Basic 的安装

Visual Basic 6.0 的安装与常用软件的安装方法类似。可用 Visual Studio 6.0 产品的第一张光盘或者 Visual Basic 6.0 单行版光盘进行安装, 安装方式有典型安装、自定义安装和最小安装三种。

另外, 也可以从网上下载免费软件安装, 可以在相关网站搜索“VB 6.0 下载”等关键字, 下载免费的安装程序, 双击安装包并根据相关向导安装即可。

### 3. Visual Basic 的启动与退出

单击 Windows“开始”按钮, 指向“程序”, 再指向 Microsoft Visual Basic 6.0 图标, 就可以启动 Visual Basic 6.0, 并进入到 Visual Basic 6.0 集成开发环境。

当需要退出 Visual Basic 时, 可以关闭 Visual Basic 集成环境窗口, 或单击“文件”菜单下的“退出”命令。

## 1.2 Visual Basic 编程规范

在编程中, 指定一个统一的规范是非常必要的, 这样可以提高程序的可读性和可维护性, 使编程效率大大提高。

### 1.2.1 命名规范

#### 1. 工程命名

如果是通用组件工程, 可直接命名; EXE 工程直接命名; ActiveX 控件和 DLL 工程命名格式为“\* Lib”; 如果是项目或产品工程, 则使用项目或产品缩写作为前缀, 如: XWY... Lib。(工程名、文件夹、文件名、函数名、变量名一般都不使用中文命名, 除非题目要求或者文件需要和用户交互!)

#### 2. 变量命名

如果是具有某种含义的变量, 则尽量使用其含义名称来命名, 如:

```
Dim nameStr As String
```

```
Dim count As Integer
```

对于一般的临时性变量定义,应该尽可能地简单,如:

```
Dim i As Integer
For i=0 to 100
Next i
```

如果是类对象或自定义类型变量,则可使用类名或自定义类型名的简写,如:

```
Dim em As EnityManager
```

**注意** 习惯上,所有前缀都全部小写,后面的单词首字母大写。

另外,声明所有的变量将会节省编程时间,因为键入操作引起的错误减少了。在“工具”菜单下的“选项”对话框的“编辑器”标签中,选中“要求变量声明”选项或采用 Option Explicit 语句要求在 Visual Basic 程序中声明所有的变量。常用下列前缀来指明一个变量的数据类型,见表 1-1。

表 1-1 常用变量前缀

数据类型	前缀	例子
String(字符型)	str	strFName
Integer(整型)	int	intQuantity
Long(长整型)	lng	lngDistance
Single(单精度型)	sng	sngAverage
Double(双精度型)	dbl	dblTolerance
Boolean(布尔型)	bln	blnFound
Byte(字节型)	byt	bytRasterData
Date(日期型)	dte	dteNow
Currency(货币型)	cur	curRevenue
Object(对象型)	obj	objCurrent
Variant(变体型)	vnt	vntChecksum

### 3. 控件命名

在一个应用程序中,当一个窗体或者控件被创建后,系统首先会给它们设置一个默认的名称,如窗体默认为 Form1, Form2, 命令按钮默认为 Command1, Command2, 标签默认为 Label1, Label2 等,控件命名一般使用“控件类型缩写”加“控件用途”的命名方式,类型缩写应控制在 3 个字母以内,缩写规则同变量命名,如果使用了新的控件,则首先应该在小组内协同一致其类型名称缩写后再进行使用。控件名称的常用前缀见表 1-2。

表 1-2 常用控件前缀

控件类型	前缀	例子
CheckBox	chk	chkReadOnly
CommandButton	cmd	cmdExit
CommonDialog	dlg	dlgFileOpen
Control	ctr	ctrCurrent

控件类型	前缀	例子
Data	dat	datBiblio
DataCombo	dbc	dbcAuthor
DataGrid	dgd	dgdTitles
DataList	dbl	dblPublisher
DirListBox	dir	dirSource
DriveListBox	drv	drvTarget
FileListBox	fil	filSource
Form	frm	frmEntry
Frame	fra	fraLanguage
Grid	grd	grdPrices
HScrollbar	hsb	hsbVolume
Image	img	imgIcon
ImageList	ils	ilsAllIcons
Label	lbl	lblHelpMessage
Line	lin	linVertical
ListBox	lst	lstPolicyCodes
ListView	lvw	lvwHeadings
Menu	mnu	mnuFileOpen
OptionButton	opt	optGender
PictureBox	pic	picVGA
ProgressBar	prg	prgLoadFile
RichTextBox	rtf	rtfReport
Shape	shp	shpCircle
Slider	sld	sldScale
StatusBar	sta	staDateTime
TabStrip	tab	tabOptions
TextBox	txt	txtLastName
Timer	tmr	tmrAlarm
ToolBar	tlb	tlbActions
TreeView	tre	treOrganization
UpDown	upd	updDirection
VScrollbar	vsb	vsbRate

用下列前缀来指示数据访问对象,见表 1-3。

表 1-3 常用数据库对象前缀

数据库对象	前缀	例子
Container	Con	conReports
DataBase	Db	dbAccounts
DBEngine	Dbe	dbeJet
Document	Doc	docSalesReport
Field	Fld	fldAddress
Group	Grp	grpFinance
Index	Idx	idxAge
Parameter	Prm	prmJobCode
QueryDef	Qry	qrySalesByRegion
RecordSet	Rec	recForecast
Relation	Rel	relEmployeeDept
TableDef	Tbd	tbdCustomers
User	Usr	usrNew
WorkSpace	Wsp	wspMine

应用程序频繁使用许多菜单控件,对于这些控件具备一组唯一的命名约定很实用。除了“mnu”标记以外,菜单控件的前缀要被扩展:对每一级嵌套增加一个附加前缀,将最终菜单的标题放在名称字符串的最后。常用的菜单前缀见表 1-4。

表 1-4 常用菜单前缀

菜单标题序列	菜单处理器名称
File Open	mnuFileOpen
File Send Email	mnuFileSendEmail
Format Character	mnuFormatCharacter
Help Contents	mnuHelpContents

#### 4. 函数命名

函数(此处包括 Sub 和 Function)表示的是一个动作,所以它的结构一般是“动词+名词”,动词小写,后面的名称首字母大写,如:

```
updateGrid
readOrder
```

函数命名尽量不要使用缩写,而且它的名称应该一目了然,让人能够从名称就知道这个函数的功能。参数命名的原则是全部小写,如果参数包括两个或两个以上的单词时,首单词字母小写,其他单词首字母大写,如 showCol,isUpdate。

#### 5. 常量命名

常量的命名应该全部大写,使用“\_”作为单词间的分隔符,单词尽量使用全名称,如:

```
Public Const MSG_EMPTY_ROW As String="有空行存在!"
```

**注意** 对常量的声明必须带上类型,如上面的 As String。

## 6. 类、窗体与模块

类的命名使用功能名词,不必加任何前缀和后缀,单词首字母大写,如 SystemConfig。

窗体命名使用“功能名词+Form 后缀”,如 ListFrm。

模块命名不必加任何前缀和后缀,直接命名。

### 1.2.2 Visual Basic 代码书写格式规范

#### 1. 定义

定义的代码块应该放在一起,尽量不要在代码中间定义变量。变量的定义应该顶行对齐,不能缩进,同时尽量保证“As”关键字的对齐,如:

```
Dim i As Integer
Dim s As String
Dim em As EntityManager
```

#### 2. 空行

空行是区分代码块与块的间隔,在函数之间可以加上空行,而函数内部、变量声明块和实现块也可以使用空行来间隔。实现块的内部通过空行来标识一个功能段,如:

```
Private Sub Form_Load()
Dim i As Integer
ReDim a(20)           '在过程内直接用 Redim 声明数组 a
    For i=1 To 20
        a(i)=i
        Print a(i)
    Next i
End Sub               '注意: 不要使用过多的空行,空行太多影响代码阅读!
```

#### 3. 缩进

习惯上,代码编写时应采用缩进格式。变量声明块不缩进,功能实现块全部缩进。对于基本的控制结构,应该有缩进,如 If, Do, With, For, Open, Select 等。

对于过长的语句,可以使用续行符号(空格加下划线)来续行,如:

```
sql="SELECT [code],[name] FROM [Person] " _
&" WHERE [code] LIKE '001% '"
```

#### 4. 注释

注释以尽可能少为宜,但应做到别人能够结合注释来读懂代码,也就是说,注释是在代码无法充分表达函数功能时才进行提供,注释同样应该做到准确简洁。如:

```
***
'读取单据信息
'@ param OrderID 单据号
'@ param Order 单据
Private Function ReadOrder(OrderID As String, Order As NYSaleBackLib.Order)
As Boolean
End Function           '函数结束
```

### 1.2.3 养成良好的编程习惯

(1) 用“+”来连接字符串,容易导致歧义,良好的习惯是用“&”来连接字符串。如:

不正确

```
Dim sMessage As String
    sMessage="1"+2
```

正确

```
Dim sMessage As String
    sMessage="1"&"2"
```

(2) 变量命名大小写一致,使用适当的缩进,代码错落有序。

例如,比较以下两段代码:

较难读懂的代码:

```
Dim SNAME As String
Dim NTURN As Integer
If NTURN=0 Then
If SNAME="vbeden"Then
Do While NTURN<4
NTURN=NTURN+1
Loop
End If
End If
```

容易读懂的代码:

```
Dim sName As String
Dim nTurn As Integer
If nTurn=0 Then
    If sName="vbeden" Then
        Do While nTurn<4
            nTurn=nTurn+1
        Loop
    End If
End If
```

(3) 在重复对某一对象的属性进行修改时,尽量使用 With... End With,如:

```
Form1.Height=5000
Form1.Width=6000
Form1.Caption="This is MyLabel"
```

可以写为

```
With Form1
    .Height=5000
    .Width=6000
    .Caption="This is MyLabel"
End With
```

这种结构程序执行效率比较高。

## 1.2.4 什么是好的代码

- (1) 可读性很强的代码格式,能够区分不同的代码块。
- (2) 清晰明了的命名,在尽可能短的名称长度下能够传递足够多的信息。
- (3) 注释和代码相得益彰的注释(不要让注释重复代码所能表达的信息)。
- (4) 变量的生存期尽可能地短,这样读程序的人不用去记大量的变量声明。
- (5) 使用小函数,将功能复杂的大函数进行分隔。

总之,代码的好坏应该由别人是否容易读懂进行区分。

## 1.3 应用程序调试与错误处理

无论多么仔细的编写代码,程序中都可能会有错误,程序调试是指从程序中找到每个错误和存在的问题,然后逐一加以解决。调试是程序开发过程中必不可少的阶段。



### 1.3.1 错误的种类

#### 1. 编辑错误

在编辑代码时, Visual Basic 会对键入的代码直接进行语法检查。当发现代码存在输入错误、遗漏关键字或标点符等语法错误, Visual Basic 会弹出一个窗口, 提示出错信息, 出错的那一行变成红色。这时, 用户必须单击“确定”按钮, 关闭出错提示窗, 然后对出错行进行修改。

例如, 若 If 语句没有与之相对应的 End if 语句, 使用了一个 Next 语句而没有与之对应的 For 语句等, 编译应用程序时会检测到这些错误。如果已经设置了“自动语法检测”功能, 那么, 只要在代码窗口中输入一个语法错误, Visual Basic 就会立即显示错误消息。要设置“自动语法检测功能”, 执行“工具”→“选项”→“编辑器”选项卡→“自动语法检测”即可。如图 1-1 所示的第三行代码中, 因为输入的括号和引号为中文标点, 所以出错。

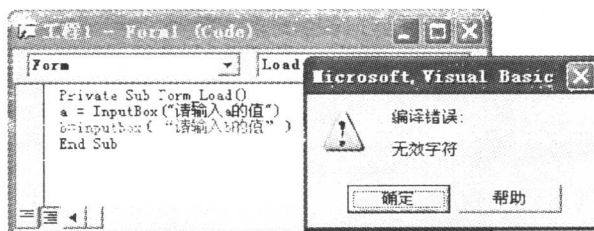


图 1-1 编辑错误示例

#### 2. 编译错误

单击了“启动”按钮后, 在 Visual Basic 开始执行程序前, 先编译执行程序段, 由于用户未定义变量、遗漏关键字等原因, 使一个语句执行一个不能执行的操作时就会发生运行时错误, 如图 1-2 所示。语句本身的语法是正确的, 编译错误只有当执行到该行程序时才能被检测到。

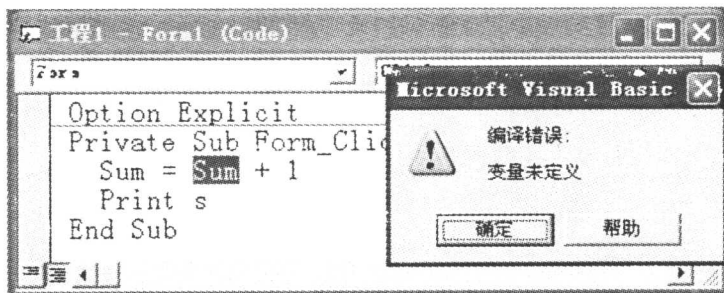


图 1-2 编译错误示例

#### 3. 逻辑错误

程序运行后, 得不到所期望的结果, 从语法角度来看, 应用程序的代码是正确的, 在运行时也未执行无效操作, 但是产生了不正确的结果, 这说明程序存在逻辑错误。这类错误往往是逻辑上的缺陷所引起, 例如, 运算符使用不正确、语句的次序不对、循环语句的起