

V i s u a l B a s i c

Visual Basic

学习指导与试题解析

孙一平 吴琼雷 编著

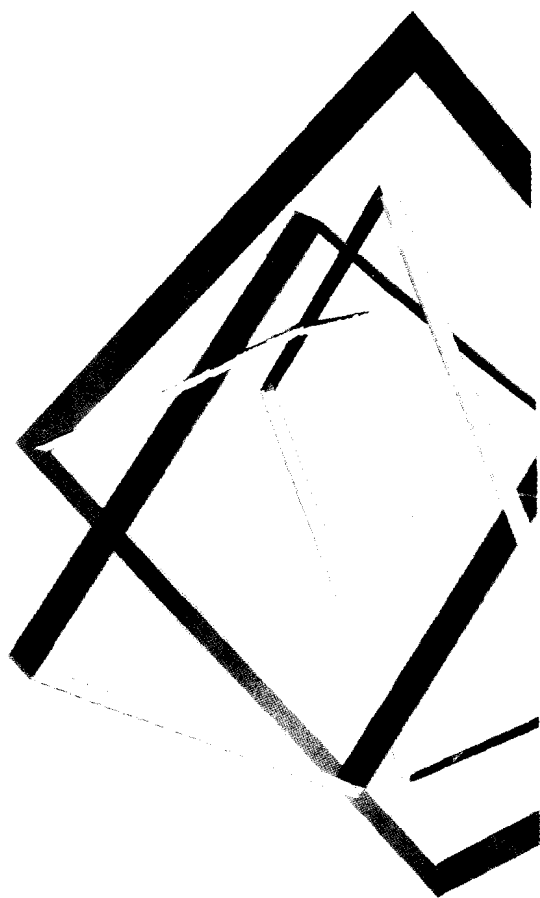
东南大学出版社

Visual Basic

Visual Basic

学习**指**导与试题解析

孙一平 吴琼雷 编著



东南大学出版社
·南京·

内 容 提 要

本书通过大量实例深入浅出地介绍 Visual Basic 的操作、使用和编程,辅助学生深入学习 Visual Basic 程序设计课程。本书内容实用、层次分明、讲解清晰,其中融入了大量的测试题及难点解析,是学生进行编程实战的好助手,是各类高等院校、高等职业学校及相关院校非计算机专业的 Visual Basic 程序设计课程教学辅导教材,也可作为培训班教材和自学读物。

图书在版编目(CIP)数据

Visual Basic 学习指导与试题解析 / 孙一平,吴琼雷编著. — 南京:东南大学出版社,2004.5

ISBN 7-81089-415-3

I. V... II. ①孙... ②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(2004)第 002072 号

Visual Basic 学习指导与试题解析

出版发行 东南大学出版社
出 版 人 宋增民
社 址 南京市四牌楼 2 号(邮编:210096)
电 话 (025)83794844 (025)83362442(传真)
网 址 press.seu.edu.cn
印 刷 南京玉河印刷厂
开 本 787×1092 mm 1/16
印 张 13
字 数 312 千
版 次 2004 年 5 月第 1 版 2004 年 5 月第 1 次印刷
印 数 1—3000 册
定 价 25.00 元(随书附赠光盘一张)

* 未经本社授权,本图书内任何文字图片不得以任何方式转载、演绎,违者必究。

* 东大版图书若有印装质量问题,请直接向发行科调换,电话:025—83795801。

前 言

计算机科学技术是当代发展最为迅速的科学技术。新技术、新概念、新知识、新应用层出不穷,要求我们不断学习、终生学习。

Visual Basic 6.0 是微软公司推出的可视化编程语言。可视化编程为我们提供了一条便捷之路,让我们能摆脱过去 DOS 下那种既枯燥又复杂的编程方式。在各种可视化编程语言中,Visual Basic 是最容易学习的一种编程语言。它提供了生成向导、拖放技术、属性检查以及丰富的功能控件,可以进行数据处理、文字、声音、影像的制作和播放,以及数据库的访问等操作,是目前可视化程度最高的开发工具,适于开发 Windows 下的任何应用程序。

目前高校非计算机专业常以 Visual Basic 作为第一门计算机程序设计语言课程,向学生传授程序设计的基本知识,使学生熟悉一个实用的图形界面的开发环境,掌握程序设计的基本思想和基本方法。但是由于学生是首次接触程序设计,而且课时数有限,难以较快地掌握 Visual Basic 的精华和熟练地编程。编者撰写本书的目的就是想帮助读者缩短这一过程,尽快掌握编程技巧。书中共分 7 章,首先介绍了 Visual Basic 的基本概念和数据类型,然后以算法与程序结构、过程和函数、过程和变量的作用域为核心,讲述程序结构、基本算法、过程、函数、作用域的概念,每章辅以大量实例,并且进行解析,说明算法、流程和知识点。最后一章是综合测试,给出多份模拟测试卷,对较难的题目或者概念性强的题目设有难点提示。本书所举实例均在 VB5.0、VB6.0 下调试通过,具有可实践性。希望读者阅读本书的同时,能把握好实践环节,上机操作验证,举一反三,做到融会贯通,归纳总结,进一步提高程序设计方法和应用软件开发能力。

本教材配有一套自学用的演示光盘,课件和书中的例题、测试题均录制在光盘上,可安装到硬盘上使用,也可以使用大屏幕投影进行教学。

在本书编写过程中得到王庆宝老师、吴继光老师、王栋老师、张琦老师等的大力支持,在此一并表示衷心的感谢。限于时间和编者水平,书中不妥之处难免,敬请批评指正。

编 者

2004. 1

目录

mulu

第1章 初识 Visual Basic	(1)
1.1 Visual Basic 能做什么	(1)
1.2 Visual Basic 特点	(1)
1.3 Visual Basic 的集成开发环境及窗体	(2)
1.3.1 启动 VB 并编写第一个程序	(2)
1.3.2 对象、属性、方法和事件概念以及在程序设计中的用法	(4)
1.3.3 窗体对象	(4)
1.3.4 控件简介	(8)
1.3.5 VB 应用程序界面设计的要点	(13)
1.4 测试题	(16)
第2章 Visual Basic 语言基础	(18)
2.1 Visual Basic语言的基本元素	(18)
2.2 数据类型	(18)
2.3 变量、常量、表达式	(20)
2.3.1 变量	(20)
2.3.2 常量	(21)
2.3.3 表达式	(21)
2.4 常用内部函数	(23)
2.5 程序语句	(29)
2.6 测试题	(30)
第3章 算法和程序设计	(31)
3.1 算法的基本概念	(31)
3.2 程序的基本结构	(33)
3.2.1 程序的基本结构	(33)
3.2.2 VB提供的主要语句	(33)
3.3 数组和控件数组	(39)
3.3.1 数组的概念	(39)
3.3.2 数组的基本操作	(41)
3.3.3 控件数组	(44)
3.4 常用算法应用举例	(45)
3.5 测试题	(65)
第4章 过程和函数	(66)
4.1 Visual Basic应用程序的组成	(66)

4.2 过程分类	(67)
4.2.1 事件过程	(67)
4.2.2 通用过程	(68)
4.2.3 Function(函数)过程	(74)
4.3 向过程传递参数	(76)
4.4 过程的嵌套与递归调用	(85)
4.5 测试题	(93)
第5章 过程与变量的作用域	(96)
5.1 过程与变量的作用域	(96)
5.1.1 过程级(局部)变量	(96)
5.1.2 模块级(文件级)变量	(96)
5.1.3 应用程序级(项目级、工程级、全局)变量	(97)
5.2 静态变量	(97)
5.3 变量的作用域应用实例	(98)
5.4 测试题	(105)
第6章 文件操作和管理	(108)
6.1 文件类型和文件操作	(108)
6.1.1 文件的分类和访问	(108)
6.1.2 文件访问语句和函数	(109)
6.2 文件系统控件的使用	(124)
6.3 利用公共对话框打开文件	(128)
6.4 测试题	(132)
第7章 综合测试	(135)
综合测试一	(135)
综合测试二	(140)
综合测试三	(146)
综合测试四	(153)
综合测试五	(165)
综合测试六	(174)
综合测试七	(180)
测试答案和难点解析	(188)
参考文献	(199)



第1章 初识 Visual Basic

学习要点

- 掌握 Visual Basic 集成开发环境 IDE 的使用;
- 了解 Visual Basic 的常用控件和属性;
- 掌握对象、属性、方法和事件驱动原理及创建 Visual Basic 程序的一般步骤。

1.1 Visual Basic 能做什么

Visual Basic(简称 VB)是微软(Microsoft)公司推出的一套用来开发 Windows 应用程序的编程工具。“Visual”的意思是“视觉的、看见的”,对编程来说就是“可视化的程序设计”。可视化编程可以让我们走出过去 DOS 下那种既枯燥又复杂的编程方式。VB 提供了生成向导、拖放技术、属性检查以及丰富的功能控件。只需按设计的要求,用系统提供的工具在屏幕上“画出”各种对象,VB 将自动产生界面设计代码。程序员所需要编写的只是程序中的运算及动作的处理那部分代码,大大提高了编程效率。而这又是以广为流传的 Basic 编程语言为基础。VB 继承了 Basic 语言简单易用的特点,采用面向对象的程序技术(OOP),引用对象、属性、方法、事件等新概念来解决程序设计中的问题。VB 语言简单且功能强大,是市场上最强大的 Windows 程序设计语言之一,支持高级编程技术,又是现有编程语言中最容易的一种。

VB 能做什么?

VB 的帮助中说:“VB 没有什么不能干!”确实如此。从设计新型的用户界面到利用其他应用程序的对象,从处理运算、文字、图像到使用数据库,VB 提供了完成这些工作的所有工具。

1.2 Visual Basic 特点

Visual Basic 作为一种可视化计算机编程语言,首先是用户界面的制作非常友好和方便,从一般的菜单和窗体界面到特殊菜单和窗体的处理功能都很齐全,成为用户界面处理中的佼佼者。VB 具有丰富的图形、图像功能,可以很容易地画出点、线、矩形、圆形等基本图形,还提供多达 25 万种色彩,利用 VB 6.0 自带的控件和丰富的 Windows API 函数,可以实现有关图形的各种操作,满足各种需求。VB 在文字及文件处理方面也是卓越不凡,在文件的操作上提供了与系统直接相连的函数,在文字处理上具有一般字符串处理功能,可以实现特殊的文字效果。VB 还具有很强的多媒体处理控制功能,它的多媒体控制 MCI(Media Control Interface)包括一套控制音频和视频设备,但具有与设备无关的命令。VB 在数据库开发方面也具有强大的功能,能够读取和访问 Access、Excel、dBase、FoxPro、Btrieve 和 ODBC 等多种数据库,并能利用 VB 自身所带的数据库引擎创建 Access 数据库,所以 VB 在



管理信息系统(MIS)的开发和建设方面也得到广泛应用。

对于初学者应当首先从 VB 的集成开发环境 IDE 入手,了解 VB 程序的编写、修改和调试。使编程方法、思想与集成开发环境完美结合,才能将 VB 巨大的潜力充分发挥。

1.3 Visual Basic 的集成开发环境及窗体

VB 集成开发环境 IDE 是在 Windows 平台上的一个应用程序,它的启动和其他应用程序一样,从“开始”→“程序”→“应用程序名”。对学习 IDE 的要求如下:

- (1) 能够正确使用菜单条、工具栏,在集成环境中放置各种窗口,在代码编辑器中操作;
- (2) 了解工程文件、窗体文件的基本概念以及创建新工程文件、保存和打开工程文件和生成可执行文件的方法;
- (3) 能够在集成开发环境中获得帮助,使用功能键【F1】迅速取得帮助以及使用“帮助”菜单中的命令、使用联机手册。

1.3.1 启动 VB 并编写第一个程序

在计算机中安装了 VB5 或 VB6 之后,单击屏幕左下角的“开始”按钮,将鼠标指向“程序(P)”,再指向“Microsoft Visual Basic 5.0(或 6.0)”,最后单击“Microsoft Visual Basic 5.0(或 6.0)”执行程序,就可以启动 VB。如果在 Windows 的桌面已经建立了“VB 快捷方式”,可以双击“VB 快捷方式”启动 VB。

启动 VB 后,很快会看到 VB 的封面(包括版本名称、编号等信息),略等片刻,封面消失,随即进入 VB 的开发环境,再单击“新建工程”窗口中的“标准 EXE”图标,然后单击右边的“打开”按钮,就可以创建一个新的工程(或直接双击“标准 EXE”图标即可),如图 1-1 所示。

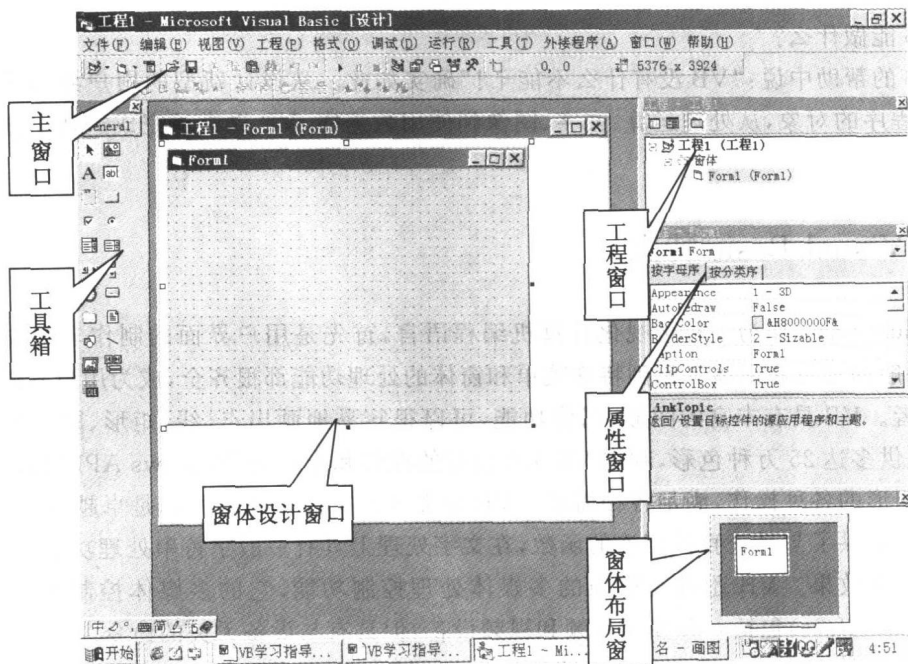


图 1-1 在 VB 中创建新的工程的界面



1) VB 主界面简介

VB 安装后,默认的主界面由六大部分组成:主窗口、窗体窗口、工具箱、工程窗口、属性窗口和窗体布局窗口。

主窗口是 Windows 应用程序的标准窗口,包括标题栏、菜单栏、常用工具栏等。

屏幕中间是“窗体设计窗口”,这是设计屏幕界面的窗口。屏幕左边的长方条是工具箱,显示了常用的控件 20 个,这是创建界面的主要元素。

单击菜单栏中的“视图”项,打开视图菜单,可以看到菜单中有代码窗口、对象窗口、立即窗口等多个窗口选项,单击这些选项之一,就可以在屏幕上显示相应的窗口。单击各个窗口右上角的“✕”关闭按钮,可以关闭相应的窗口。

2) 编写第一个程序

例 1-1 该程序的要求是:单击窗体,显示“欢迎进入 Visual Basic”,单击按钮“口令”,则将文本框中输入的口令显示在消息框中。编写该程序的主要步骤是:第一步界面设计,根据要求设计如图 1-2a 所示的操作界面;第二步代码设计,根据操作要求,本题有两个事件,单击窗体和单击按钮,因此编写了两个事件过程(Form_Click 和 Command1_Click),如图 1-2b 所示,运行结果如图 1-2c 所示。

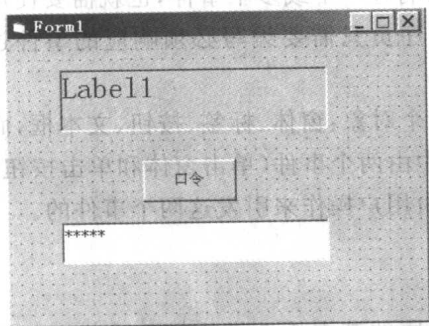


图 1-2a 界面设计

```
Private Sub Command1_Click()  
    vv = Val(Text1.Text)  
    MsgBox "口令是:" + Str(vv)  
End Sub  
  
Private Sub Form_Click()  
    Label1.Caption = "欢迎进入 Visual Basic!"  
End Sub
```

图 1-2b 代码窗口中的代码

表 1-1 对象属性表

对象	对象名	属性名称	属性值
窗体	Form1		
标签	Label1	Caption	
按钮	Command1	Caption	口令
文本框	text1	text	

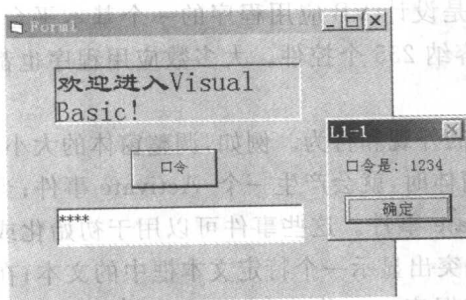


图 1-2c 程序的运行结果

3) 退出 VB

按 VB 窗口右上角的“✕”关闭按钮,则关闭 VB 窗口,退出 VB 程序。如果所编写的程序未保存,则会提示保存。



1.3.2 对象、属性、方法和事件概念以及在程序设计中的用法

面向对象的编程方法是把程序和数据封装起来作为一个对象,并为每个对象赋予相应的属性。在设计对象时,不必编写建立和描述每个对象的程序代码,而是用工具(工具箱中的控件)“画”在界面上,由 VB 自动生成对象的程序代码并封装起来。对象的操作是通过事件来执行的,事件是完成某种特定功能的若干个子程序,即过程。而应用程序的执行可以由用户引发某个事件(如单击界面上的按钮或单击窗体等事件)来驱动完成某种功能,或由事件驱动程序,调用通用过程执行指定的操作。这又称为事件驱动编程机制。

每个对象具有特定的属性和方法,属性是描述对象外观的相关数据,设置其不同的值,可以改变对象的外观;方法是 VB 系统为对象预先设计好的一些功能子程序,供每个对象操作使用。

对象的事件(Event)就是对象上所发生的事情,在 VB 中,事件是预先定义好的、能够被识别的动作,如单击(Click)事件、双击(DblClick)事件、装载(Load)事件、鼠标移动(Mouse Move)事件等,不同的对象能够识别不同的事件。当事件发生时,VB 将检测两条信息,即发生的是哪种事件和哪个对象接收了事件。事件过程是一段独立的程序代码,它在对象检测到某个特定事件时执行,称作响应事件。一个对象可以响应一个或多个事件,也就需要使用相应的事件过程来对用户或系统的事件做出响应。程序员只需要编写必须响应的事件过程,而无用的事件过程则可不必要编写。

对上述的例 1-1 进行分析,界面上(图 1-2a)有 4 个对象:窗体、标签、按钮、文本框,而每个对象又具有各自的属性,如表 1-1 中所示。例题中由两个事件(单击窗体和单击按钮)来完成题目给定的任务,这两个事件就是两个过程,是由用户操作来引发这两个事件的。

1.3.3 窗体对象

窗体是 VB 最重要的对象,是 Visual Basic 应用程序的基本构造模块。在运行应用程序时,作为与用户交互操作的实际窗口。它是包容用户界面或对话框所需的各种控件的容器,相当于画布,也相当于 Windows 中的桌面。这是设计 VB 应用程序的一个基本平台,几乎所有的控件都是添加在窗体上的,每个窗体可容纳 255 个控件。大多数应用程序也都是由窗体开始执行的。

窗体有自己的属性、事件和方法,用来控制窗体的外观和行为。例如,调整窗体的大小,会触发一个 Resize 事件。每当一个窗体变成活动窗体时,就会产生一个 Activate 事件;当另一个窗体或应用程序被激活时,就会产生 Deactivate 事件。这些事件可以用于初始化或关闭窗体。例如,在 Activate 事件中,可以编写代码突出显示一个特定文本框中的文本;在 Deactivate 事件中,可以更改保存到一个文件或数据库中。若要使一个窗体可见,调用 Show 方法(如 Form2.Show)等。

设置窗体的方法是:当启动 VB 程序时,在“建立工程”窗口中双击“标准 EXE”图标,在建立工程的同时就会自动创建一个窗体,或者在 VB 窗口中,选择“文件”菜单下的“创建工程”命令,也可以创建一个窗体。如图 1-1 所示。下一步就是设置它的属性,在设计时可以在“属性”窗口中直接设置完成,或者运行时由代码“窗体名.属性=属性值”来实现。

为了学习方便,将常用的窗体属性归纳在表 1-2 中。



表 1-2 常用的窗体属性

	属性名称	属性含义	属性值	用 途
1	名称 (Name)	窗体(对象)名称	名称字符串	在程序代码中作为对象名被引用
2	Caption	窗体标题	字符串	显示在窗体的标题栏中
3	BackColor	窗体的背景色	背景色的代码值	用于改变窗体的背景色
4	ForeColor	窗体的前景色	前景色的代码值	用于改变窗体的前景色
5	Border- Style	窗体边框风格	0~5	改变窗体边框的样式: 0 无(没有边框或与边框相关的元素) 1 固定单边框。可以包含控制菜单框,标题栏,"最大化"按钮,和"最小化"按钮。只有使用最大化和最小化按钮才能改变大小 2 (缺省值)可调整的边框。可以使用设置值1列出的任何可选边框元素重新改变尺寸 3 固定对话框。可以包含控制菜单框和标题栏,不能包含最大化和最小化按钮,不能改变尺寸 4 固定工具窗口。不能改变尺寸 5 可变尺寸工具窗口。可改变大小
6	Enabled	允许(活动)属性	True 或 False	允许在运行时使窗体成为有效或无效。返回或设置一个值,该值用来确定一个窗体是否能够对用户产生的事件做出反应
7	Icon	图标属性	某图标位置及名称	设置窗体标题上的图标
8	Left	窗体的左边界	坐标值	设置窗体左边界的位置
9	Top	窗体的顶边	坐标值	设置窗体顶边界的位置
10	Height	窗体的高	高度值	设置窗体的高度
11	Width	窗体的宽	宽度值	设置窗体的宽度
12	Font	字体	字体或字型或字号的代码值	设置窗体所显示正文的字体、字型和字号
13	Picture	背景图像	图像的位置和名称	设置窗体上的背景图像
14	Visible	可见属性	True 或 False	控制窗体可见或隐藏。返回或设置一个逻辑值,指示对象为可见或隐藏

方法可以使对象执行一个动作或任务,这是面向对象程序设计中引入的称为方法的特殊过程和函数。方法的操作与过程、函数的操作相同,并且是特定对象的一部分,用于完成某种特定的功能,如打印、显示、移动等。窗体上常用的主要方法如表 1-3 所示。



表 1-3 窗体使用的主要方法

	方法名称	语法	执行的动作或任务
1	Hide	<i>object.Hide style, ownerform</i> 如果省略 <i>object</i> , 则带有焦点的窗体就认为是该 <i>object</i>	隐藏指定的窗体, 该窗体就从屏幕上被删除, 并将其 Visible 属性设置为 False。用户将无法访问隐藏窗体上的控件, 但不能使其卸载。但是对于运行中的 Visual Basic 应用程序, 或对于通过 DDE 与该应用程序通讯的进程及对于 Timer 控件的事件, 隐藏窗体的控件仍然是可用的。窗体被隐藏时, 用户只有等到被隐藏窗体的事件过程的全部代码执行完后才能够与该应用程序交互。如果调用 Hide 方法时窗体还没有加载, 那么 Hide 方法将加载该窗体但不显示它
2	Show	<i>object.Show style, ownerform</i>	显示指定的窗体。如果调用 Show 方法时指定的窗体没有装载, Visual Basic 将自动装载该窗体。 <i>style</i> 为可选的。为一个整数时, 它用以决定窗体是有模式还是无模式。如果 <i>style</i> 为 0, 则窗体是无模式的; 如果 <i>style</i> 为 1, 则窗体是模式的。 <i>ownerform</i> 为可选的。为字符串表达式时, 指出部件所属的窗体被显示。对于标准的 Visual Basic 窗体, 使用关键字 Me
3	Move	<i>object.Move left, top, width, height</i>	用以移动窗体, 按左、顶、宽、高给出的值移动
4	Print	<i>object.Print [outputlist]</i>	在窗体上打印或显示输出表列 <i>outputlist</i> [<i>outputlist</i>] 的参数为 {Spc(<i>n</i>) Tab(<i>n</i>)} <i>expression charpos</i> Spc(<i>n</i>) 可选, 用来在输出中插入空白字符, <i>n</i> 为要插入的空白字符数。 Tab(<i>n</i>) 可选, 用来将插入点定位在绝对列号上, <i>n</i> 为列号。使用无参数的 Tab(<i>n</i>) 将插入点定位在下一个打印区的起始位置。 <i>expression</i> 可选, 为要打印的数值表达式或字符串表达式。 <i>charpos</i> 可选, 指定下个字符的插入点: 使用分号 (;) 直接将插入点定位在上一个被显示的字符之后; 使用 Tab(<i>n</i>) 将插入点定位在绝对列号上, 使用无参数的 Tab 或逗号 将插入点定位在下一个打印区的起始位置。 如果省略 <i>charpos</i> , 则在下一行打印下一字符。 对于 Boolean 数据, 则打印 True 或者打印 False
5	PrintForm	<i>object.PrintForm</i>	用以将窗体对象的图像逐位发送给打印机, 即打印窗体对象的全部可见对象和位图窗体
6	Refresh	<i>object.Refresh</i>	强制全部重绘一个窗体, 即刷新窗体上的内容。通常, 如果没有事件发生, 窗体或控件的绘制是自动处理的。但是, 有些情况下希望窗体或控件立即更新。例如, 如果使用文件列表框、目录列表框或者驱动器列表框显示当前的目录结构状态, 当目录结构发生变化时可以使用 Refresh 更新列表

窗体的方法只能在程序代码中使用, 其用法依赖于所需的参数的个数以及它是否有返



返回值。当方法不需要参数并且也没有返回值时,调用方法的用法是:对象名.方法名。

窗体可以响应的常用事件如表 1-4 所示。

表 1-4 窗体可以响应的常用事件

	事件名称	触发该事件的动作或原因	产生的事件过程名	事件用途
1	Initialize (初始化事件)	当应用程序创建或引用 Form 或 MDI 或 User 控件、类属性或类的实例时发生	Private Sub object_Initialize()	应用此事件初始化窗体或多窗体所用的数据。对 Form 或 MDIForm, Initialize 事件在 Load 事件之前发生
2	Activate(激活事件)	当一个窗体成为当前窗体时发生	Private Sub object_Activate()	窗体激活时需要处理的事件
3	Deactivate (失活事件)	当一个对象不再是当前窗体时发生	Private Sub object_Deactivate()	窗体失活时需要处理的事件
4	Load(加载事件)	在一个窗体被装载时发生。或者当使用 Load 语句启动应用程序,或引用未装载的窗体属性或控件时,此事件发生	Private Sub Form_Load()	Load 事件过程用来包含一个窗体的启动代码。例如,指定控件缺省设置值,指明将要装入 ComboBox 或 ListBox 控件的内容,以及初始窗体级变量等。Load 事件是在 Intialize 事件之后发生
5	Click(单击事件)	在一个对象上单击按下然后释放一个鼠标按钮时(即单击鼠标左键)发生。它也会在一个控件的值改变时发生。对于一个 Form 对象来说,该事件是在单击一个空白区或一个无效控件时发生	Private Sub Form_Click()	使用这个事件来触发一个动作以响应控件中的变化
6	DblClick (双击事件)	当在一个对象上按下和释放鼠标按钮并再次按下和释放鼠标按钮(即双击)时,该事件发生。对于窗体而言,当双击被禁用的控件或窗体的空白区域时,DblClick 事件发生	Private Sub Form_DblClick()	使用这个事件来触发一个动作以响应控件中的变化。注意:如果在 Click 事件中有代码,则 DblClick 事件将永远不会被触发,因为 Click 事件是两个事件中首先被触发的事件。其结果是鼠标单击被 Click 事件截断,从而使 DblClick 事件不会发生
7	Resize(改变大小事件)	当一个对象第一次显示或当一个对象的窗口状态改变时该事件发生	Private Sub Form_Resize()	用于一个窗体被最大化、最小化或被还原或者拖动窗口边界改变大小时需要处理的事件代码
8	Paint(重绘事件)	在一个对象被移动或放大之后,或在一个覆盖该对象的窗体被移开之后,该对象部分或全部暴露时,此事件发生	Private Sub Form_Paint()	如果需要代码中各种图形方法的输出,使用 Paint 过程,可以确保这样的输出,在必要时能被重绘

1.3.4 控件简介

“控件”是在图形用户界面上进行输入、输出信息,启动事件程序等交互操作的图形对象,是进行可视化程序设计的基础和重要工具。在 VB 中是预先定义好的、程序中能够直接使用的对象,可以用它来获取用户的输入信息和显示输出信息。每个控件都有一组属性、方法和事件,可以在设计时或在代码中修改和使用。利用控件编程使程序员免除了大量重复性的工作,能够以最快的速度 and 效率开发具有良好用户界面的应用程序。

VB 中的控件通常分为 3 种类型:

(1) 内部控件 在默认状态下工具箱中显示的控件都是内部控件,这些控件被“封装”在 VB 的 EXE 文件中,不可从工具箱中删除。如命令按钮、单选框、复选框等。

(2) ActiveX 控件 这类控件单独保存在 .ocx 类型的文件中,其中包括各种版本提供的控件,如数据绑定网格、数据绑定组合框等和仅在专业版或企业版中提供的控件,如标准公共对话框控件、动画控件和 MCI(多媒体)控件等,还有软件厂商提供的 ActiveX 控件。

(3) 可插入的对象 用户可将 Excel 工作表或 PowerPoint 幻灯片等作为一个对象添加到工具箱中,编程时可根据需要随时创建。

VB 开发环境中的工具箱中列有 20 个内部控件,每个控件都用一个图形按钮来表示,并标有一个名称,称为 Visual Basic 中的标准控件。表 1-5 列出这些控件的名称、形状、常用属性和方法、用途。

表 1-5 VB 内部控件的主要参数

控件图标	控件名称及用途	主要属性	主要方法和事件
	命令按钮 (CommandButton) 控件,创建命令按钮对象,用于执行命令	Name, Caption, Cancel, Default, Index, Enabled, picture	方法 SetFocus 事件 Click
	标签 (Label) 控件,创建一个标签对象,用于保存不希望用户改动的文本	Name, Caption, Alignment, AutoSize, Visible, BorderStyle, Enabled, Font, ForeColor	方法 Refresh, Move 事件 Change, Click, Dblclick
	文本框 (TextBox) 控件,创建用于显示和输入数据的文本对象,用户可以在其中输入或更改文本	Name, Text, Alignment, Passwordchar, MaxLength, MultiLine, ScrollBars, BorderStyle, Enabled, Font, ForeColor, Locked (锁定), TopStop (Tap 键属性)	方法 Refresh, SetFocus 事件 Change, Click, Dblclick, Keypress
	列表框 (ListBox) 控件,创建列表框对象,用于显示供用户选择的列表项。当列表项很多,不能同时显示时,列表可以滚动	Name, List, ListCount, ListIndex, Text, Columns, Sorted	方法 AddItem, Clear, RemoveItem 事件 Click, Dblclick
	组合框 (ComboBox) 控件,创建组合框或下拉列表对象,用户可以从列表中选择某一项或人工输入一个值	Name, style(0:缺省值,下拉式组合框。包括一个下拉式列表和一个文本框。可以从列表选择或在文本框中输入;1:简单组合框,包括一个文本框和一个不能下拉的列表;2:下拉式列表,这种样式仅允许从下拉式列表中选择)	方法 AddItem, Clear, RemoveItem 事件 Click, Dblclick



续表 1-5

控件图标	控件名称及用途	主要属性	主要方法和事件
	水平滚动条 (HScrollBar) 控件和垂直滚动条 (VScrollBar) 控件, 用于辅助浏览显示内容、确定位置, 可按比例指示当前位置。可以作为数量或速度的指示器及数据输入工具, 控制模拟量的输入, 例如声音控制、颜色调整或者查看定时处理中已用的时间	Name(引用的控件名) Max(滚动条所能表示的最大值) 整型表示, 取值范围 32768~32767 min(滚动条所能表示的最小值) 整型表示, 取值范围 32768~32767 Value(设置滚动块当前位置值) 注意, 不能把 Value 属性值设在 Min 和 Max 范围之外 LargeChange(滚动幅度) 整型, 范围 1~32767 之间的整数 SmallChange(Value 的变化量) 整型, 范围 1~32767 之间的整数	方法 Move, SetFocus 事件 Scroll: 当在滚动条内拖动滚动块时触发该事件 Change: 该事件发生在滚动块移动后或通过代码对 Value 属性赋值后
	计时器 (Timer) 控件, 创建计时器对象, 以设定的间隔捕捉计时器事件。此控件运行时不可见。主要用于在程序中监视和控制时间进程。这对不需要与用户交互代码的执行非常有用	Interval(整型) 取值范围 0~65535 毫秒, 因此, 最大时间间隔不能超过 65 秒, 通常 60000 毫秒为 1 分钟。Interval 设为 1000 时, 每秒产生一个 Timer 事件; 设为 n/1000, 每秒钟产生 n 个 Timer。Interval 设置小于 56, 则不会有什么效果, 设置为 0, 则计时器无效	事件 Timer, 每经过一个 Interval 属性规定的时间间隔, 就触发 Timer 事件
	选择按钮 (OptionButton, 又称单选按钮) 控件, 创建选择按钮组对象, 用于显示多个选项, 用户只能从中选择一个选项	Name, Caption, Style, Alignment, Value	方法 Move, Refresh 事件 Click, Dblclick
	复选框 (CheckBox) 控件, 创建复选框对象, 允许用户选择开关状态, 或显示多个选项, 用户可以从中选择多个选项	Name, Caption, Style, Alignment, Value	方法 Move, Refresh 事件 Click, 不支持 Dblclick
	框架 (Frame) 控件, 用于窗体上的对象分组, 可以提供视觉上的区分和总体上的激活/屏蔽特性, 并可用来美化其他控件	Name(用于程序代码中识别一个框架), Caption(用于定义框架的可见文字部分), Enable(设置为 True, 使框架内的对象是“活动”的, 设置为 false, 则标题会变灰色, 框架中的所有对象均被屏蔽)	
	驱动器列表框 (DriveListBox) 控件, 用来显示用户系统中所有有效磁盘驱动器的列表, 供用户选择	Name(用于设定控件名), Drive(用于显示当前可用的驱动器或用户在驱动器列表框中选取的驱动器) Enabled(控制能否对用户产生的事件做出反应)	方法 ChDrive, Refresh (当目录结构发生变化时更新列表) 事件 Change
	目录列表框 (DirListBox) 控件, 显示目录列表, 供用户选择	Name, Path, Enabled(控制能否对用户产生的事件做出反应)	方法 ChDir, Refresh (同上) 事件 Change
	文件列表框 (FileListBox) 控件, 在 Path 属性指定的目录中, 该控件将文件定位并列举出来。即显示当前路径下的文件列表, 供用户选择	Name, Path, Pattern, FileName, ListCount, ListIndex, Enabled(同上) FileListBox 可以创建对话框, 通过它从任一可用驱动器的磁盘文件列表中打开文件	方法 Move, Refresh 事件 PathChange PatternChange

续表 1-5

控件图标	控件名称及用途	主要属性	主要方法和事件
	形状(Shape)控件,创建形状对象,设计时用于画各种类型的形状。可以画矩形、圆角矩形、正方形、圆角正方形、椭圆或圆	Name, BackColor, ForeColor, Height, Width, BackStyle, BorderColor, Left, Top, BorderStyle, BorderWidth, Shape, DrawMode, Tag, FillColor, Visible, FillStyle, Index(控件数组)	方法 Move, Zorder, Refresh
	线条(Line)控件,创建线条对象,设计时用于在窗体上画各种类型的线条	Name, BorderColor, BorderStyle, Parent, BorderWidth, Tag, Container, Visible, DrawMode, X1, Y1, X2, Y2, Index	方法 Move, Zorder, Refresh
	图片框(PictureBox)控件,用于显示图形文件或文本文件,也可以作为其他控件的容器(父对象),显示位图、图标、JPEG、GIF 文件的图形,并可剪裁图形	CurrentX 和 CurrentY 坐标属性,可以设置水平坐标和垂直坐标。Picture:在设计阶段用于放置图片	方法 Move, Refresh paintpicture(绘制图形文件), 事件 Change, Click, Dblclick
	图像(Image)控件,用于创建图像对象,在窗体上显示位图、图标、JPEG、GIF 等图形文件。它使用较少的系统资源,重画起来比 PictureBox 控件要快,Image 能放在容器里,但不能作为容器。单击时,其动作类似于命令按钮	CurrentX 和 CurrentY 坐标属性,用于设置水平坐标和垂直坐标。Picture:在设计阶段用于放置图片	方法 Move, Refresh 事件 Click, Dblclick
	数据(Data)控件,用于连接数据库,并在窗体的其他控件中显示数据库信息	Connect(数据控件要访问的数据库的类型), DatabaseName(使用的数据库名), RecordSource(访问的数据表的名称)	方法 Move, Refresh 事件 Validate(激活另一条记录时启动某些动作), Resize(对象的窗口状态改变时发生该事件)
	OLE 容器(OLE Container)控件,创建 OLE 容器对象,用于把其他应用的数据嵌入到 VB 的应用程序中		

怎样使用表 1-5 中的控件呢?下面说明使用控件的几点基本原则:

(1) 用户需要与应用程序交互 最简便的方法是给用户提供一个按钮,即使用 CommandButton 控件,使用户通过简单的敲击按钮来执行操作。当用户选中按钮时,不仅会执行相应操作,还会使该按钮看上去像被按下并释放一样。无论何时,只要用户单击按钮,就会调用 Click 事件过程。将需要的执行代码写入 Click 事件过程,就可执行想要执行的动作。也可以利用包含图形的 Image 控件,如一个图标,来创建自己的“按钮”。

(2) 应用程序中只需要显示一些提示文本 使用 Label 控件,应用程序在窗体中会显示预先输入在 Label 上的文本,Label 中的文本为只读文本。

(3) 需要显示和编辑文本 使用 TextBox 控件,允许用户输入自己的文本到 TextBox 控件中。TextBox 中的文本为可编辑文本。

(4) 应用程序需要向用户提供选择 如果只允许用户选择单项,使用 OptionButton 控

件;如果允许用户选择一项或多项,使用 CheckBox 控件;如果需要多组选择,则再使用 Frame 控件进行分组控制。

(5) 如果需要用户从可滚动列表的表格中进行选择 使用 ListBox 控件。如果还需要附加一个文本编辑域,可使用 ComboBox 控件。

(6) 函数 LoadPicture(图片名):用于程序运行阶段,可以加载图片。

(7) 如果要控制声音音量或调整图片颜色,可以使用滚动条控件 ScrollBar。ScrollBar 控件可按比例指示当前位置,可以单独使用,以控制在程序中输入模拟量。滚动条又分 HScrollBar(水平滚动条)和 VScrollBar(垂直滚动条)两个控件,它们是独立的,各有自己的事件、属性和方法,可以作为输入设备。

注意:TextBox、ListBox、ComboBox 或 MDI 窗体具有 ScrollBars 属性,可以在这些控件上增加或删除附属于它们的 ScrollBar。

例 1-2 利用计时器和滚动条控件来控制图像的闪烁和闪烁的速度。

其工作界面如图 1-3 所示,界面上的控件有 5 个,各控件的属性如表 1-6 所示。

其事件有两个:定时器事件和滚动条变化事件,其程序代码如下:

' 滚动条变化事件代码

```
Private Sub HScroll1_Change()
```

```
    Tmr1.Interval = HScroll1.Value
```

```
End Sub
```

' 定时器事件代码

```
Private Sub Tmr1_Timer()
```

```
    Img1.Visible = Not Img1.Visible
```

```
End Sub
```

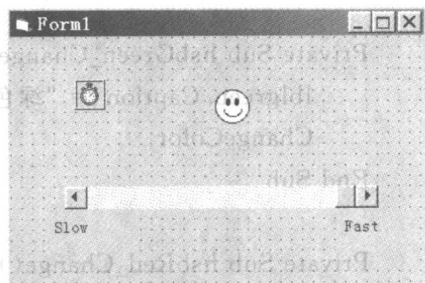


图 1-3 例 1-2 的工作界面

表 1-6 例 1-2 使用的控件属性表

控件名称	属性名称	属性值
LBL1(标签)	Caption	Slow
LBL2(标签)	Caption	Fast
Hscroll1(滚动条)	Max Min	300 1500
Tmr1(计时器)	Interval	0
Img1(图像框)	Picture	Vb\Graphics\Misc\Face02

运行时,拖动滚动条中的滑动块,引发滚动条变化事件,执行其中的代码,获取滚动条的 Value 属性作为计时器 Tmr1 的时间间隔。每当计时器的时间间隔到达时引发一次定时器事件,执行其中的代码,对 Img1 控件的 Visible 取反,使得图形交替地显示和隐藏,达到闪烁的效果。

为了使滚动条右端表示闪烁得快,左端表示闪烁得慢,滚动条的 Max 属性值设置为 300,Min 属性值设置为 1500。这是因为计时器的 Interval 属性值取自滚动条的 Value 属性值,时间间隔越大,对应的闪烁速度越慢,所以将计时器的 Max 设定为小于 Min。

例 1-3 利用滚动条调节文本框的背景色。