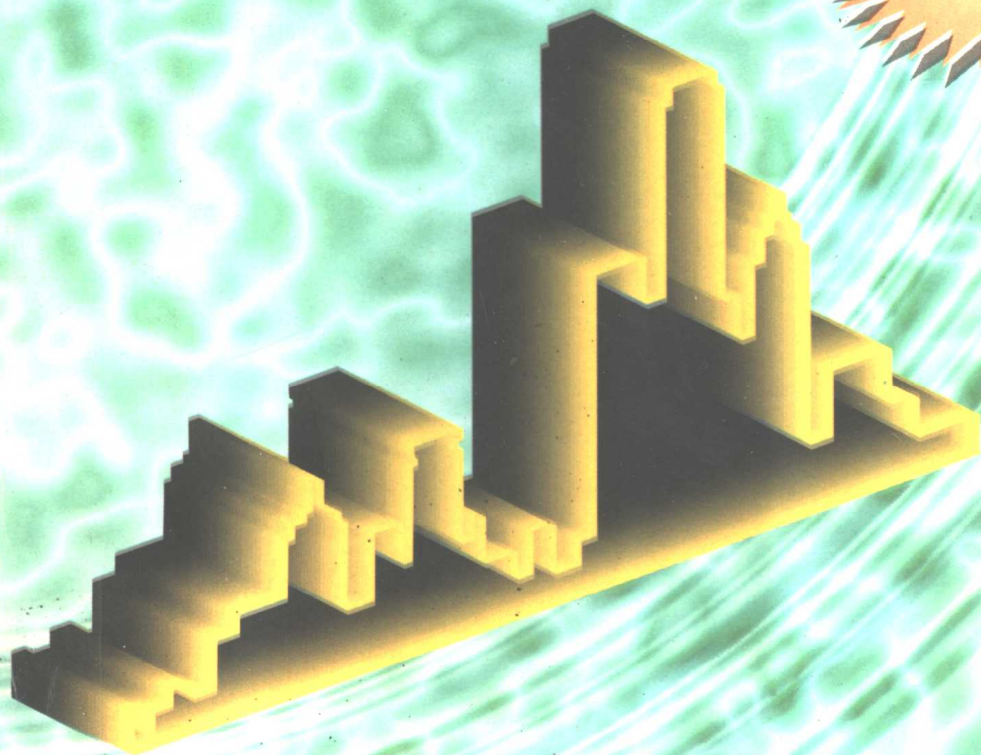


家用电脑软件硬件自学入门

编程技巧
专题讨论
实用程序
随附磁盘



杨 睿 裔 隽 主 编
金 雪 清 尹 伯 文 编 著

Visual Basic

(V3.0/4.0)

编程技巧与程序实例



四川大学出版社

Visual Basic(V3—4)编程技巧与程序实例

裔 隽 主编

杨 睿 金雪清 尹伯文 编著

四川大学出版社

(川)新登字014号

责任编辑:曾 鑫

封面设计:晓 毅

技术设计:彬 彬

内 容 介 绍

本书是一本有关Visual Basic 编程技巧的书,作为本书重点,在第四章中对VB 编程特技,作了四十个专题讨论,其中前二十五个专题直接介绍了VB 本身,后十五个专题分别介绍了十多个第三方提供的VBX。

本书涉及了Windows 编程的各个方面,诸如文本输入的Undo、窗口之间的拖放操作、打字效果、封锁键盘输入等。我们试图利用VB 这一最流行的通用Windows 编程语言来达到一些一般难以达到的效果,使读者认识到VB 不仅仅是一种简单易学的入门级语言,也能够利用其创造出富有魅力的程序。

本书后面所介绍的第三方控件可以说是本书的精华部分。这些控件是笔者平时收集的VBX 中的精品,其中有不少是刚刚发布在Internet 上的共享软件。

在本书所附的磁盘中,不仅有所有的源程序代码,也包括了所有的第三方控件(均为Shareware 版本,使用时不会有什么问题,只是会多显示一个版权页)。

Visual Basic(V3—4)编程技巧与程序实例

裔 隽 主编

杨 睿 金雪清 尹伯文 编著

四川大学出版社出版发行

(成都市望江路29号)

新华书店经销

四川大学印刷厂印刷

开本:787×1092毫米 1/16

印张:17.5 字数:294千字

1996年12月第1版

1996年12月第1次印刷

印数:1—6000册

ISBN 7—5614—1470—6/TP·43

定价:31.00元

(附赠光盘)

序 言

很高兴四川中外科技文化交流中心给了我们这一机会,使我们可以将一些自己研究的东西和收集的资料在这里整理好、写出来,奉献给大家。

作为主要作者,我在编程领域里滚打了许多年,从最早APPLE的FBASIC开始,经历了GW BASIC、dBASE、PASCAL、C等,到近几年的C++、FoxPro、Visual Basic、Delphi,感觉编程的确是一件很辛苦的事。不但软件的总件设计耗费大量的精力,在具体的编程中,还会碰到许许多多的问题:有操作系统本身的,也有编程语言和环境带来的,更多的则是编程者本身的问题。我们应该感谢Windows操作系统的诞生,以及随之而来的真正意义上的编程革命。Microsoft公司充满创造力的Visual Basic,使我们从烦琐的传统编程中解放出来。我们或许只要花几个小时的时间,就可编制出过去需要几个月才能完成的项目。并且,由于VBX这类控件结构的诞生,使我们能在最大程度上享受编程高手们的杰作,将其运用于自己的程序中。编程不再是一项神秘的工作,利用Visual Basic这样的可视化语言,只要稍下工夫,你就能把自己烦琐的工作简化。

Visual Basic的入门是很容易的,但精通并不是人人都能做得到的,我们在这条路上,稍微先走了一步,我们总结了一些心得体会,收集了一些优秀的控件提供给大家,能对编程有些帮助。

在写作本书时,我们各人时间都很紧,加上世界上计算机技术的突飞猛进,我们在这里只能起到一个抛砖引玉的作用,希望大家在仔细阅读之后,能够有助于提高自己的编程能力,并且由此扩展思路,在Windows的世界里任意翱翔。

如果我们的努力能给你带来一点帮助,我们将感到很高兴。

在写作本书的过程中,得到了白晓毅先生的大力支持,在此,表示由衷的感谢。

同时,我要感谢我的父母,从小对于我的支持,以及所有帮助过、鼓励过我们的人。

最后我要感谢本书的合作者:杨睿和金雪清,一起走过的这段不平凡的日子。

裔 隽

一九九六年十月于上海

目 录

第一章 VB 能为我们带来什么	(1)
1.1 直观的界面设计	(2)
1.2 面向对象的设计思想	(2)
1.3 简单易学的语言	(2)
1.4 自动的事件驱动	(3)
1.5 具有相当丰富的控件	(3)
1.6 动态数据交换(DDE)和对象的链接与嵌入(OLE)	(3)
1.7 动态链接库(DLL)的使用	(3)
1.8 强大的数据库访问功能	(4)
1.9 与VBA 的良好兼容性	(4)
第二章 Visual Basic 基本知识	(5)
2.1 对象、属性、方法、事件、事件驱动	(5)
2.1.1 对象(Object)	(5)
2.1.2 属性(Properties)	(5)
2.1.3 方法(Methods)	(5)
2.1.4 事件(Events)、事件驱动	(6)
2.2 数据类型、变量、变量作用范围	(6)
2.2.1 数据类型(Data Type)	(6)
2.2.2 变量(Variable)	(7)
2.2.3 变量作用范围	(7)
2.3 数组、自定义数据类型	(7)
2.3.1 数组(Array)	(7)
2.3.2 自定义数据类型	(8)
2.4 过程、函数	(8)
2.4.1 过程(Procedure)	(8)
2.4.2 函数	(9)
第三章 Visual Basic 编程快速入门	(10)
3.1 基本语法	(10)
3.1.1 判断语句	(10)
3.1.2 循环语句	(11)
3.1.3 其他语句	(11)
3.2 窗体、菜单	(11)
3.2.1 窗体(Form)	(11)
3.2.2 菜单(Menu)	(12)
3.3 标签、文本框	(13)

3.3.1	标签(Label)	(13)
3.3.2	文本框(Text)	(13)
3.4	按钮、图形框、图象框	(14)
3.4.1	按钮(Command)	(14)
3.4.2	图象框(Image)	(14)
3.4.3	图形框(Picture)	(14)
3.5	滚动条、单选框、复选框、框架	(15)
3.5.1	滚动条(Scroll Bar)	(15)
3.5.2	单选框(Option)、复选框(Check)	(15)
3.5.3	框架(Frame)	(16)
3.6	列表框、组合框、通用对话框	(16)
3.6.1	列表框(List)	(16)
3.6.2	通用对话框(CMDialog)	(17)
3.7	计时器	(17)
第四章	专题与技巧	(19)
4.1	字符串操作入门	(19)
4.1.1	引出问题	(19)
4.1.2	知识点	(19)
4.1.3	界面设计	(19)
4.1.4	程序输入	(20)
4.1.5	讨论	(20)
4.2	产生不重复的随机数 随机数的应用	(22)
4.2.1	引出问题	(22)
4.2.2	程序设计	(22)
4.2.3	讨论	(23)
4.3	旋转图片 数学函数的应用	(26)
4.3.1	引出问题	(26)
4.3.2	知识点	(26)
4.3.3	界面设计	(26)
4.3.4	程序输入	(27)
4.3.5	讨论	(28)
4.4	为文本框增加Undo功能	(31)
4.4.1	引出问题	(31)
4.4.2	知识点	(31)
4.4.3	界面设计	(31)
4.4.4	程序输入	(32)
4.4.5	讨论	(33)
4.5	计算相隔日期:时间日期函数的应用	(36)
4.5.1	引出问题	(36)

4.5.2	用到的知识	(36)
4.5.3	界面设计	(37)
4.5.4	程序输入	(38)
4.5.5	讨论	(39)
4.6	创建用户自定义的鼠标形状	(43)
4.6.1	引出问题	(43)
4.6.2	用到的技术和知识点	(43)
4.6.3	界面设计	(43)
4.6.4	程序输入	(44)
4.6.5	讨论	(45)
4.7	封锁键盘和鼠标的输入	(48)
4.7.1	引出问题	(48)
4.7.2	用到的技术和知识点	(48)
4.7.3	界面设计	(48)
4.7.4	程序输入	(49)
4.7.5	讨论	(50)
4.8	“关于”窗口和简单动画实现	(51)
4.8.1	用到的技术和知识点	(51)
4.8.2	界面设计	(51)
4.8.3	程序输入	(52)
4.8.5	讨论	(53)
4.9	多文档用户接口使用	(54)
4.9.1	引出问题	(54)
4.9.2	知识点	(55)
4.9.3	讨论	(59)
4.10	通用对话框的使用方法	(62)
4.10.1	引出问题	(62)
4.10.2	用到的技术和知识点	(62)
4.10.3	界面设计	(62)
4.10.4	程序输入	(64)
4.10.5	讨论	(65)
4.11	鼠标的应用:隐藏鼠标、单击和双击	(70)
4.11.1	引出问题	(70)
4.11.2	知识点	(70)
4.11.3	界面设计	(70)
4.11.4	程序输入	(71)
4.11.5	讨论	(72)
4.11.6	引出问题	(73)
4.11.7	知识点	(74)

4.11.8	界面设计	(74)
4.11.9	程序输入	(75)
4.11.10	讨论	(76)
4.12	两个窗口之间拖放控件	(77)
4.12.1	引出问题	(77)
4.12.2	用到的知识点和技术	(77)
4.12.3	界面设计	(77)
4.12.4	程序输入	(79)
4.12.5	讨论	(82)
4.13	密码输入窗口	(85)
4.13.1	引出问题	(85)
4.13.2	知识点	(86)
4.13.3	界面设计	(86)
4.13.4	程序输入	(88)
4.13.5	讨论	(90)
4.14	专业化的统计图——Graph 控件的使用	(91)
4.14.1	引出问题	(91)
4.14.2	用到的知识:	(91)
4.14.3	界面设计	(92)
4.14.4	建立程序	(93)
4.14.5	讨论	(94)
4.15	记录型文件的应用——完整的通讯录	(101)
4.15.1	引出问题	(101)
4.15.2	知识点	(101)
4.15.3	界面设计	(101)
4.15.4	程序输入	(103)
4.15.5	讨论	(109)
4.16	检查特定的文件是否存在	(113)
4.16.1	引出问题	(113)
4.16.2	用到的技术和知识点:	(113)
4.16.3	界面设计	(113)
4.16.4	程序输入	(115)
4.16.5	讨论	(116)
4.17	OutLine 控件的使用	(119)
4.17.1	引出问题	(119)
4.17.2	知识点	(119)
4.17.3	界面设计	(120)
4.17.4	程序输入	(121)
4.17.5	讨论	(123)

4.18	打字效果——Sendkey 的应用	(124)
4.18.1	引出问题	(124)
4.18.2	用到的知识	(124)
4.18.3	界面设计	(125)
4.18.4	程序输入	(126)
4.18.5	讨论	(127)
4.19	运行时自动提示的实现	(128)
4.19.1	引出问题	(128)
4.19.2	用到的知识	(128)
4.19.4	界面设计	(128)
4.19.5	程序输入	(129)
4.19.5	讨论	(131)
4.20	图形化菜单	(132)
4.20.1	引出问题	(132)
4.20.2	用到的知识	(133)
4.20.3	界面设计	(133)
4.20.4	程序输入	(134)
4.20.5	讨论	(135)
4.21	显示计算机和Windows 的资源	(137)
4.21.1	引出问题	(137)
4.21.2	用到的知识	(137)
4.21.3	界面设计	(137)
4.21.4	程序输入	(141)
4.21.5	讨论	(145)
4.22	利用Shell 函数调用其他应用程序	(147)
4.22.1	引出问题	(147)
4.22.2	用到的知识点	(147)
4.22.3	界面设计	(147)
4.22.4	程序输入	(148)
4.22.5	讨论	(149)
4.23	进程条显示	(152)
4.23.1	引出问题	(152)
4.23.2	知识点	(152)
4.23.3	界面设计	(152)
4.23.4	程序输入	(153)
4.23.5	讨论	(154)
4.24	双记时钟	(155)
4.24.1	引出问题	(155)
4.24.2	用到的知识	(155)

4.24.3	界面设计	(155)
4.24.4	程序输入	(156)
4.24.5	讨论	(157)
4.25	浮在最上面的的时钟	(159)
4.25.1	引出问题	(159)
4.25.2	用到的知识	(159)
4.25.3	界面设计	(159)
4.25.4	程序输入	(160)
4.25.5	讨论	(161)
4.26	工具条	(163)
4.26.1	介绍	(163)
4.26.2	程序范例	(164)
4.26.3	讨论	(172)
4.27	分隔棒控件	(175)
4.27.1	介绍	(175)
4.27.2	程序范例	(176)
4.27.3	讨论	(183)
4.28	VB 常用子程序	(187)
4.28.1	介绍	(187)
4.28.2	程序	(187)
4.29	华丽的界面显示	(192)
4.29.1	介绍	(192)
4.29.2	程序范例	(192)
4.29.3	讨论	(208)
4.30	闹钟控制	(213)
4.30.1	介绍	(213)
4.30.2	程序范例	(213)
4.30.3	讨论	(215)
4.31	磁盘和文件的详细信息	(216)
4.31.1	介绍	(216)
4.31.2	程序范例	(216)
4.31.3	讨论	(219)
4.32	Tab 控制	(222)
4.32.1	介绍	(222)
4.32.2	程序范例	(222)
4.32.3	讨论	(231)
4.33	菜单的辅助说明	(232)
4.33.1	介绍	(232)
4.33.2	程序范例	(233)

4.33.3	讨论	(235)
4.34	进程条控件	(237)
4.34.1	介绍	(237)
4.34.2	程序范例	(237)
4.34.3	讨论	(238)
4.35	万年历:年历控件的使用	(240)
4.35.1	介绍	(240)
4.35.1	程序范例	(241)
4.35.3	讨论	(243)
4.36	INI文件控制	(245)
4.36.1	介绍	(245)
4.36.2	程序	(245)
4.36.3	讨论	(246)
4.37	Zip文件控制	(247)
4.37.1	介绍	(247)
4.37.2	程序	(247)
4.37.3	讨论	(250)
4.38	图片按钮的各类效果	(251)
4.38.1	介绍	(251)
4.38.2	程序范例	(251)
4.38.3	讨论	(252)
4.39	状态栏	(255)
4.39.1	介绍	(255)
4.39.2	程序范例	(256)
4.39.3	讨论	(257)
4.40	液晶数字显示效果	(259)
4.40.1	介绍	(259)
4.40.2	程序介绍	(260)
4.40.3	讨论	(267)

第一章 VB 能为我们带来什么

1. 简介

Visual Basic 是由大名鼎鼎的美国 Microsoft 公司推出的编程语言。Microsoft 的总裁就是由 Basic 语言白手起家的,因此 Microsoft 对 Basic 语言情有独钟。从 GWBASIC 到 Quick Basic,再到今天的 Visual Basic,他们将快要淘汰的 BASIC 语言又重新发扬光大了!

Visual Basic 已经成为 Microsoft 的一个重要武器,其孪生兄弟 Visual Basic for Application(简称 VBA)已成为 MS OFFICE 的标准宏语言。Visual Basic 更有可能成为 Microsoft 对付 JAVA(一种用于网络的跨平台编程语言)的秘密武器。在 Microsoft 公司最近推出的 Internet Explore 3.0 中,已经全面支持用 BasicScript 和 ActiveX(VBX、OCX 的最新接班人)写的程序。

有过编程经验的人都知道,为应用软件写一个漂亮实用的界面是一件很费功夫的事。据专家评测,在传统的 DOS 编程之中,设计界面及其相应的实现程序,大约要耗掉 80% 的工作量,并且无论是采用多么好的软件设计方法,这一沉重的包袱总甩不掉。幸好,出现了 Windows 操作系统,Windows 提供了许多标准的界面元素和标准的操作方法,使程序员的编程变得容易,用户的使用也变得简单。在 Windows 下,所有的软件都具有基本相同的图形界面、菜单、按钮和对话框等。你所设计的软件只要遵循一些 Windows 软件开发的规定,那么用户就用不着费力地寻找和思考,就能投入实际应用中,这是软件开发人员期盼的多么好的事!

简而言之,在 Visual Basic 中,我们通过这样的两个步骤写程序:

我们直接用鼠标构造用户界面:菜单、按钮、文本输入框、列表框等都能象画图一样画出来,你全然不用管如何画按钮、如何控制按钮按下的效果、多层次的下拉式菜单如何操作、漂亮的三维外观如何构造等等。

然后你告诉 Visual Basic 什么时候干什么:比如某个按钮按下后显示什么信息,某个菜单选择后进行什么样的计算等。至于捕获按钮按下、处理文本显示、窗口移动、屏幕覆盖后的还原操作等这些难题交给 Visual Basic 和 Windows 本身。

在 Visual Basic 中你还可以调用许多用户熟悉的标准的 Windows 资源(如标准下拉式菜单、你经常见到的带个叹号或是问号的消息窗口、文件打开窗口、打印设置窗口、调色板等),这样你的应用程序就具有和其他 Windows 软件一样标准的界面和一致的操作方法。

2. Visual Basic 的版本

我们使用的 Visual Basic 版本全称为: Visual Basic Professional Edition Version 3.0 (VB3.0 专家版)和 Visual Basic 4.0 16bits Version。后者是 Microsoft 推出的新的升级版本,并且还有一个 32 位版本和企业版本(限于用途,本书不具体介绍这两个版本,VB 的企业版在界面和 OLE 功能上提高了许多)。本书以 VB4.0 的 16 位为平台,同时兼顾 VB3.0 版本。本书中的程序未加说明的都可以在 VB3.0 上使用,本书后面所介绍的第三方控件的源程序全部在 VB3.0 版上运行。

3. Visual Basic 的特点

下面介绍的Visual Basic 的几大特点让你先有一个全局的印象,具体的内容都会在后面的章节逐步展开,在之后的两章内,将会复习一下Visual Basic 编程的概念,如果你现在对VB还是一窍不通,那么希望你先看一本有关如何使用VB的入门书籍,这样,在阅读本书的过程中,就能起到事半功倍的作用。

从以下的说明,可以让你知道VB能做什么和怎样利用VB来编制程序,这其中,涉及到不少Windows的术语,如果你对Windows本身还不太熟悉,不要紧,慢慢你就会明白的(当然,还是最好抽空去看一些有关Windows的基础书籍)。其实,利用VB可以写出几乎所有的Windows应用软件,如果你觉得某些方面有不可逾越之处,那只是因为你可能没有想出关键之处,或是创造力还不太够。希望在阅读完本书之后,同时加上你的努力,你能成为一个自由驾驭Windows的VB编程高手。

VB提供了这样的一条道路:从羡慕别人的软件到亲手写程序,最后可以根据自己的想法来写出所需的应用软件。

1.1 直观的界面设计

在Visual Basic中,提供了一个可视化的界面设计环境,设计界面的主要工作是如同画图一般地用鼠标将各种界面元素,如按钮、标签、文本输入框等画在窗口之上。编程人员设计界面较其他语言更为直接,可以立刻看到最后的效果,如果有不满意的地方,可以马上修改。Visual Basic专家版还提供了效果相当美观的三维立体控件,使你可以设计出具有专业效果的界面。由于VB已经成为一个Windows下的标准编程平台,因而其界面设计也就涵盖了Windows下所有的可视元素。在有关界面设计的专题中,我们将举例说明如何用VB来实现Windows下各种商业化软件的界面。

1.2 面向对象的设计思想

虽然Visual Basic不是一种纯粹的面向对象的语言,但她已经引入了很多有益的面向对象语言的优点,使你开发程序时,能够享受到面向对象的优越性。Visual Basic将会为你自动完成许多在C++中必须自己完成的繁琐之事,在Visual Basic中,你能充分利用各种对象,比如屏幕、打印机、按钮、表格,通过简单的属性设置和方法调用,来完成你所需要的功能,至于如何声明这些对象、如何封装对象,是用不着你担心的。由于Windows本身是一种以对象组织起来的操作系统,因而VB作为一个开发环境,就自然地继承了这些对象的基本概念。

1.3 简单易学的语言

既然叫Visual Basic,当然还是Basic。Basic语言具有简单、易理解、接近自然语言的特点,而且Visual Basic融入了许多类似于Pascal的结构化语言思想(或者说是继承了Quick-Basic的特点),使程序更有组织性和可读性。Visual Basic在其代码编辑器中使用了彩色代码显示的技术,使你一眼就能区别保留字、变量名等,同时Visual Basic还能自动进行更换,比如属性名字、VB内部函数等,不管你在输入的时候怎样输入,在敲过回车之后,Visual Basic都会将其转换为同一的格式(大写字母开头,其余为小写字母);对于象变量名这样用

户自己起的名字,VB 也具有全程范围内自动替换的功能,使你在程序中的这一名字显示相同的格式;以此来增强程序的可读性。Visual Basic 在你每输入一程序后,就检查语法的正确性(比如括号是否匹配等),帮助你及时更改输入错误或是程序上的错误。另外在VB 中,提供了大量的过程的函数,包括各个方面,从字符串到文件处理等等,使你在写程序时,感到非常便捷。简单易学的语言并不意味着VB 只能写简单的应用软件,虽然用VB 写Win 软件在有些领域有限制,但在很多方面,它可以发挥强大的功效!

1.4 自动的事件驱动

事件驱动是Windows 的一个重要特性,在传统的C 和C++编程中,你必须自己构造一个循环,来捕获各类事件,处理这些事件,同时处理一些意外情况。这在一般的C 编程中是相当麻烦的一件事。但在VB 中,这些对你来说是隐含的,你需要写的代码只是事件的处理代码,如何捕获事件的发生由VB 代劳了,如何保持一个循环来反复监测用户的操作和系统的事件,你不用担心。

1.5 具有相当丰富的控件

控件(VBX)在VB 中指的是一种用来构造应用程序的对象,诸如按钮、列表框、立体标签等等都是控件,在构造界面和程序时,你主要就是对控件的属性(如颜色、位置、显示什么文字等等)和控件的事件(如被按下、被激活等)进行控制和撰写代码,并利用方法(能对控件进行怎样的操作)来实现这些控件的特殊功能。VB 本身提供了大量的控件,使你方便的利用这些控件构造你的应用程序。同时,有许多厂家出品了大量功能丰富的控件,用来给你的应用程序增大功能,使你完成以前想都不敢想的功能。在VB4 中,控件已有VBX 升级为OCX,并且分16 位和32 位。将来,可能OCX 也要被ActiveX 所取代,因为它能够跨操作系统、跨硬件平台。

1.6 动态数据交换(DDE)和对象的链接与嵌入(OLE)

这是Windows 中的两大热门话题。动态数据交换(DDE)可以实现两个不同的程序之间的数据交换;对象的链接与嵌入(OLE)可以将不同的应用程序连接起来,利用OLE 可以将别的软件作为一个对象连接到你的应用程序中。OLE 是Microsoft 的一个重要武器,用来对抗Apple 等公司的OpenDoc。在Win95 中,OLE 被广泛地应用于桌面、应用程序中。

1.7 动态链接库(DLL)的使用

由于VB 为了方便程序开发者,省却了一些低级语言的功能,这样在有些场合就会觉得不太方便,比如要想获得一个控件的句柄等。没有关系,你可以用动态链接库。动态链接库(DLL)是Windows 一种特殊的库函数,从某种意义上来说,Windows 本身就是在几个核心DLL 上建立起来的。VB 可以让你方便调用DLL 中的函数,首先声明某个DLL 库中的函数,然后就可以当它是VB 自己的函数一样使用。也已经有公司专门开发了供VB 使用的DLL 函数库。

1.8 强大的数据库访问功能

在VB中,你可以直接利用一些控件来建立数据库应用程序,并且实现与Access数据库的直接连接,同样只要通过少量的语句就可以与Foxpro、Pardox等数据库连接起来。目前,Windows环境下的可视化编程语言都拥有很强的数据库功能,比如支持SQL语言的查询等。同样,VB也能让你方便地建立复杂的数据库应用程序。VB还支持开放式数据连接(ODBC),这样可访问各种大型数据库,如Oracle、SQL Server、SYBase等。

1.9 与VBA的良好兼容性

从Microsoft的OFFICE产品中,EXCEL5.0和Project4.0都支持统一的VBA(Visual Basic for Application),熟悉VB的人会发现在这两个软件中的VBA与VB几乎是完全一样的。现在,在VB4中你也可以直接开发VBA的应用程序,调用EXCEL等中的对象。在Microsoft于1995年8月推出的OFFICE7.0中的EXCEL7.0和Project7.0中也已经全面支持VBA。在Microsoft的另一主流数据库产品Access中,其编程语言和环境几乎就是VB3.0的翻版,会用VB,几乎就会了1/3的Access。

第二章 Visual Basic 基本知识

Visual Basic 作为一种面向对象的编程方法,引进了许多全新的计算机概念。正确理解和掌握这些概念,对今后的程序设计会有很大帮助。

2.1 对象、属性、方法、事件、事件驱动

2.1.1 对象(Object)

Visual Basic 提供了一套标准的Windows 图形资源。只需轻按一下鼠标,就能轻易地得到我们所熟悉的各类窗体、菜单和对话框。这样,复杂的界面设计就变得容易多了!我们把窗体和各种控件统称为对象。对象可以从Visual Basic 提供的工具箱(图2.1)中获得。

请注意,在这里我们必须明确两个概念、资源和控件。

窗体、菜单和对话框对于程序员来说,是编程所能利用的工具,所以称作为资源。而对于用户来说,则是可接受控制的元件,故称作为控件。也就是说,Windows 的图形资源和控件在实质上说的是同一事物,只不过不同的人对它的叫法不同而已!

2.1.2 属性(Properties)

Visual Basic 赋予每个对象以不同的属性,我们可以通过修改属性来定制我们所需要的对象。属性是对象的外部可见数据,具体说来,就是对象的颜色、字体、内容、大小等等,表现为一系列的值,比如说:

```
Command1.Fontsize = 12
```

Command1 是按钮对象,FontSize 是Command1 的字号属性,12 是字号属性的值。这行代码表示Command1 按钮采用12号字。属性的赋值语句格式为:对象.属性=属性的值

有些属性可以在Visual Basic 提供的属性框(图2.2)中直接修改,而有些属性则必须在程序运行时才能修改。如列表框(List)的Listcount 属性,只有在程序运行时才会出现。

2.1.3 方法(Methods)

方法,是对象所知道的如何完成的工作,相当于用来表示动作的函数与过程。比如说,滚动条有拖拉(Drag)、移动(Move)、聚集(SetFocus)等动作,这些动作统称为滚动条这个对象的方法。举个例子来说,Clipboard.clear 该语句的功能是清除剪贴板上的内容。其中,Clipboard 为对象,clear 是Clipboard 本身知道如何完成的工作,称作为方法。调用方法的固定格式为:对象.方法

请注意,当执行一个方法时,该方法只能处理对象所包含的属性。一个对象的方法不能直接改变另一个对象的属性的值,但一个对象的方法可以请求其他对象自行改变它们自己的属性的值。

2.1.4 事件(Events)、事件驱动

事件是指在对象上所进行的操作。比如对按钮进行单击(Click)、按键(Keypress)操作。

Visual Basic 在接收到事件发生的信号后,马上传递给与之对应的对象,开始执行事先写好的子程序,这样就形成了事件驱动。事件驱动也许是Visual Basic 最吸引人的地方了!还记得C 与C++的时代吗?那时候,我们需要写出长长的循环来捕获哪一个按键被按下,在Visual Basic 中,我们唯一要做的只是告诉按钮:如果被按下后应该干什么。

事件的一般表示格式为:

对象 事件(参数)

如:

```
Sub Form—Load( )
```

```
Sub Text1—GotFocus( )
```

2.2 数据类型、变量、变量作用范围

2.2.1 数据类型(Data Type)

数据类型是指数据在内存中存储的方式。我们知道,要在固定字节数的内存中放入一个数,可以采用许多办法。

如果我们用一个整数来占据该内存的所有字节,那么这个整数就具有最长的有效位,但这种整数既没有小数位也不可能表示非常大的数值。

为了解决这个问题,我们必须使用一定的字节量来表示该数的小数点位置,余下的字节量来存放该数的数值部分。这样做,也就意味着牺牲一定的有效位来换取小数与大数的表示能力。

更进一步,我们可以使用指数形式存储相同有效位的数,如果这样做的话,就将不得不占用更多的内存,降低程序运行的效率。

Visual Basic 有11种内部数据类型,参见表2.1

表2.1

数据类型	有效位	数据大小 (字节)	范 围
整型(Integer)	5	2	-32768 到 32767
长整型(Long)	10	4	-2147483648 到 2147483647
单精度浮点数(Single)	7	4	-3.402823E38 到 -1.401298E-45
		0	-1.401298E-45 到 3.402823E38
双精度浮点数(Double)	15	8	-1.797693134862315D308 到 -4.94066D-324
		0	-4.94066D-324 到 -1.797693134862314D308
货币型(Currency)	19	8	-992337203685477.5808 到 922337203685477.5807
字符串(String)		字符总数+1	0 到 65535 个字符
布尔型(Boolean)	1	2	真或假
日期型(Date)		8	1/1/100 到 12/31/9999