



21世纪高等学校计算机科学与技术规划教材

Visual Basic

Chengxu Sheji Jiaocheng

Visual Basic 程序设计教程

主编 刘卫国

第2版



北京邮电大学出版社
www.buptpress.com



21 世纪高等学校计算机科学与技术规划教材

Visual Basic 程序设计教程

(第 2 版)

主编 刘卫国



北京邮电大学出版社
www.buptpress.com

内容提要

本书是在第1版的基础上修订而成的。全书介绍了 Visual Basic 语言基础知识及其程序设计的基本方法,使读者掌握可视化程序设计的基本思想和技术。全书分为 11 章,涵盖了 Visual Basic 程序设计的主要内容。第 1 章主要介绍 Visual Basic 集成开发环境以及 Visual Basic 的编程机制,第 2~5 章分别介绍 Visual Basic 中的数据、程序控制结构、数组及过程,第 6~8 章分别介绍 Visual Basic 控件、一些高级对象及应用程序界面设计,第 9~11 章分别介绍 Visual Basic 文件操作、图形操作和数据库应用。

本书内容丰富,理论与实践相结合,强调程序设计方法与能力的培养。在修订过程中,力求做到概念清晰、取材合理,深入浅出、突出应用,为学生应用 Visual Basic 进行程序设计和软件开发打下良好基础。

本书既可作为高等院校程序设计课程的教材,又可供社会各类软件开发人员阅读参考。

图书在版编目(CIP)数据

Visual Basic 程序设计教程/刘卫国主编. —2 版. —北京:北京邮电大学出版社,2009
ISBN 978-7-5635-2082-4

I. V… II. 刘… III. BASIC 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 163910 号

书 名	Visual Basic 程序设计教程(第 2 版)
主 编	刘卫国
责任编辑	沙一飞
出版发行	北京邮电大学出版社
社 址	北京市海淀区西土城路 10 号(100876)
电话传真	010-62282185(发行部) 010-62283578(传真)
电子信箱	ctrd@buptpress.com
经 销	各地新华书店
印 刷	北京忠信诚胶印厂
开 本	787 mm×1 092 mm 1/16
印 张	21
字 数	456 千字
版 次	2009 年 9 月第 2 版 2009 年 9 月第 1 次印刷

ISBN 978-7-5635-2082-4

定价: 32.00 元

如有质量问题请与发行部联系

版权所有 侵权必究

前 言

计算机是在程序的控制下进行自动工作的,它解决任何实际问题都依赖于解决问题的程序。程序设计是计算机应用人员的一项基本功,也是大学生的一项基本思维方式训练。只有掌握程序设计的基本知识,才能具有较强的计算机应用开发能力。

教育部高等学校非计算机专业计算机基础课程教学指导分委员会于2004年提出“1+X”课程设置模式,即一门“大学计算机基础”和若干门核心课程,“计算机程序设计基础”是其中一门重要的核心课程。通过本课程的学习,使学生了解程序设计语言的基本知识,掌握程序设计的基本方法和常用算法,掌握程序调试的基本技能,具有使用计算机解决实际问题的能力,也为学习后续课程打下良好基础。

“计算机程序设计基础”是从技术的角度学习计算机的主要基础课,已经作为大多数专业的必修课。由于不同学校、不同专业对学生程序设计能力的要求不尽相同,应用背景也不一样,所以程序设计课程通常采用不同的教学语言。但程序语言只是一种载体、一种学习程序设计的工具,无论选用哪种语言,都应掌握程序设计的基础知识与基本编程技术,学习的重点应放在程序设计的基本方法与思路上。

BASIC以其简单易学而一时成为初学者学习程序设计的入门语言。Visual Basic是在原有的结构化BASIC语言基础上发展而来的,一方面它秉承了原来BASIC语言简单易学的特点,入门比较容易;另一方面在其编程系统中采用了面向对象和事件驱动的编程机制,提供了一种所见即所得的可视化程序设计方法,为开发Windows应用程序带来了方便。Visual Basic从1991年诞生以来,其版本不断更新、功能不断增强,在软件开发领域得到广泛应用。由于Visual Basic具有简单易学且功能强大的特点,所以很适合作为人们学习程序设计的入门语言,也是许多高校首选的教学用程序设计语言。

本书是在第1版的基础上修订而成的。全书分为11章,涵盖了Visual Basic程序设计的主要内容。第1章主要介绍Visual Basic集成开发环境以及Visual Basic的编程机制,立足于建立对Visual Basic的初步认识、理解面向对象程序设计的概念和事件驱动的编程思想、能够设计简单的Visual Basic程序。第2~5章分别介绍Visual Basic中的数据、程序控制结构、数组及过程,这些内容是程序设计语言的经典内容,立足于掌握程序设计的方法与技巧,特别是要熟悉一些典型的算法。第6~8章分别介绍Visual Basic控件、一些高级对象及应用程序界面设计,这些内容是可视化程序设计的特色内容,是对传统程序设计语言的发展,立足于掌握图形用户界面的设计方法与技巧。第9~11章分别介绍Visual Basic文件操作、图形操作和数据库应用,文件操作是一种输入输出的方式,也是程序设计语言的传统内容,绘图操作和数据库应用体现了Visual Basic功能强大,这一部分立足于掌握Visual Basic的应用技能。书后的3个附录提供了一些常用Visual Basic语言资料,可供学习时查阅使用。

本书抓住算法设计和用户界面设计两大主题,内容丰富,重点突出,着力于Windows应用程序基本设计方法的训练和开发能力的培养。在修订过程中总结了第1版的使用经验,对部

分章节内容做了补充、更新,对部分内容作了适当精简,降低了难度,使得本书结构编排更加合理,内容组织更加符合教学实际需要。全书力求做到概念清晰、深入浅出、突出应用,为学生应用 Visual Basic 进行程序设计和软件开发打下良好基础。

本书既可作为高等院校程序设计课程的教材,又可供社会各类软件开发人员阅读参考。

为了方便教学和读者上机操作练习,作者还组织编写了《Visual Basic 程序设计实践教程》(第2版)一书,作为与本书配套的实验教材。另外,还有与本书配套的教学课件,供教师教学参考。

本书由刘卫国任主编,蔡旭晖任副主编。第1、9、10章由刘卫国编写,第2~5章由蔡旭晖编写,第6~8章及附录由蔡立燕编写,第11章由熊拥军编写。参与程序调试与资料整理的有刘国强、罗福星、蒋丽娟等。在本书编写过程中,得到了许多同行专家、教师的支持与帮助,在此表示衷心的感谢。

由于作者学识水平有限,书中的疏漏或不足之处在所难免,恳请广大读者批评指正。

编 者

目 录

第 1 章 Visual Basic 的初步认识	1
1.1 从 BASIC 到 Visual Basic 的演变	1
1.2 Visual Basic 集成开发环境	2
1.2.1 Visual Basic 的启动与退出	2
1.2.2 Visual Basic 的工作界面	4
1.3 事件驱动编程机制	6
1.3.1 对象和类的概念	6
1.3.2 对象的属性、方法和事件	7
1.3.3 事件驱动编程的基本要点	9
1.4 窗体及基本控件	11
1.4.1 窗体对象	11
1.4.2 标签对象	13
1.4.3 文本框对象	15
1.4.4 命令按钮对象	17
1.5 用 Visual Basic 开发应用程序	19
1.5.1 Visual Basic 程序的书写规则	19
1.5.2 创建 Visual Basic 应用程序的主要步骤	19
1.6 Visual Basic 工程的管理及环境的设置	20
1.6.1 工程的组成	20
1.6.2 工程的操作	21
1.6.3 集成开发环境的设置	24
习题	24
第 2 章 Visual Basic 的数据类型及运算	27
2.1 数据类型	27
2.1.1 基本数据类型	27
2.1.2 枚举类型	30
2.1.3 用户自定义类型	31
2.2 常量和变量	32
2.2.1 常量	32
2.2.2 变量	34
2.3 常用内部函数	37
2.3.1 数学函数	37

2.3.2	字符函数	38
2.3.3	类型转换函数	39
2.3.4	日期和时间函数	40
2.3.5	其他函数	41
2.4	表达式	43
2.4.1	算术表达式	43
2.4.2	字符串表达式	45
2.4.3	关系表达式	45
2.4.4	逻辑表达式	46
2.4.5	表达式求值和运算符的优先级	47
	习题	47
第3章	程序控制结构	49
3.1	顺序结构程序设计	49
3.1.1	赋值语句	49
3.1.2	数据输入	50
3.1.3	数据输出	53
3.1.4	暂停与结束语句	59
3.1.5	顺序结构程序应用举例	59
3.2	选择结构程序设计	62
3.2.1	简单分支结构	62
3.2.2	多分支选择结构	64
3.2.3	条件函数	67
3.2.4	选择结构程序应用举例	68
3.3	循环结构程序设计	71
3.3.1	用 For 语句实现循环	71
3.3.2	用 Do 语句实现循环	75
3.3.3	循环的嵌套	77
3.3.4	辅助控制语句	79
3.3.5	循环结构程序应用举例	81
	习题	84
第4章	数组	87
4.1	数组的概念	87
4.1.1	为什么要使用数组	87
4.1.2	数组元素与维数	88
4.1.3	数组的逻辑结构	88
4.2	数组的定义与引用	89
4.2.1	数组的定义	89
4.2.2	数组的引用	92
4.3	数组的基本操作	92

4.3.1 数组的输入	92
4.3.2 数组的其他操作	93
4.4 控件数组	95
4.4.1 设计时建立控件数组	95
4.4.2 运行时建立控件数组	97
4.5 数组的应用	98
4.5.1 一维数组的应用	98
4.5.2 二维数组的应用	105
习题	107
第5章 过程	110
5.1 Sub 过程	110
5.1.1 Sub 过程的创建	110
5.1.2 Sub 过程的调用	112
5.1.3 通用过程和事件过程	113
5.2 Function 过程	114
5.2.1 Function 过程的定义	114
5.2.2 Function 过程的调用	115
5.3 参数传递	115
5.3.1 参数传递方式	115
5.3.2 数组参数的传递	118
5.3.3 可选参数和可变参数	119
5.4 过程的递归调用	120
5.4.1 递归的概念	120
5.4.2 Visual Basic 递归过程	121
5.5 过程的应用	122
5.6 变量的作用域和生存期	125
5.6.1 变量的作用域	125
5.6.2 变量的生存期	127
5.6.3 过程的作用域	128
5.7 Visual Basic 程序结构	129
5.7.1 Visual Basic 的模块	129
5.7.2 Sub Main 过程	132
习题	132
第6章 常用标准控件	135
6.1 Visual Basic 控件概述	135
6.1.1 Visual Basic 控件的分类	135
6.1.2 默认属性与公共属性	137
6.2 选择性控件	139
6.2.1 复选框	139

6.2.2 单选按钮	140
6.2.3 框架	140
6.2.4 列表框	144
6.2.5 组合框	147
6.3 图形控件	149
6.3.1 图片框	150
6.3.2 图像框	152
6.3.3 直线与形状	152
6.4 滚动条与定时器	154
6.4.1 滚动条	154
6.4.2 定时器	155
6.5 多重窗体	158
6.5.1 多重窗体的操作	158
6.5.2 窗体操作的语句和方法	160
习题	162
第7章 高级事件与对象	165
7.1 鼠标事件	165
7.1.1 鼠标指针	165
7.1.2 鼠标事件	167
7.1.3 拖放	169
7.2 键盘事件	172
7.2.1 KeyPress 事件	172
7.2.2 KeyDown 事件和 KeyUp 事件	174
7.3 剪贴板对象及其应用	175
7.3.1 剪贴板对象的方法	176
7.3.2 文本数据的操作	177
7.3.3 其他格式数据的操作	178
7.4 RichTextBox 控件	178
7.4.1 RichTextBox 控件的常用属性	179
7.4.2 RichTextBox 控件的常用方法	180
7.5 常用 ActiveX 控件	181
7.5.1 滑动器	182
7.5.2 进度条	183
7.5.3 UpDown 控件	184
7.5.4 动画控件	186
7.5.5 多媒体控件	187
习题	190
第8章 用户界面设计	192
8.1 菜单设计	192

8.1.1 菜单设计概述	192
8.1.2 下拉式菜单设计	192
8.1.3 弹出式菜单设计	198
8.2 对话框设计	200
8.2.1 通用对话框	200
8.2.2 自定义对话框	205
8.3 多文档界面设计	207
8.3.1 多文档界面概述	207
8.3.2 建立多文档界面	207
8.4 工具栏和状态栏设计	211
8.4.1 工具栏设计	211
8.4.2 状态栏设计	216
习题	218
第9章 文件	220
9.1 文件的分类	220
9.2 文件操作	221
9.2.1 顺序文件的操作	221
9.2.2 随机文件的操作	225
9.2.3 二进制文件的操作	229
9.3 文件系统控件	230
9.3.1 驱动器列表框	230
9.3.2 目录列表框	231
9.3.3 文件列表框	231
9.4 文件基本操作	233
习题	234
第10章 图形操作	238
10.1 Visual Basic 的坐标系统	238
10.1.1 标准坐标系统	238
10.1.2 自定义坐标系统	239
10.2 绘图属性	240
10.3 绘图方法	243
10.3.1 画点方法	243
10.3.2 画直线或画矩形方法	244
10.3.3 画圆方法	247
10.4 图形处理	250
10.4.1 图形的重绘	250
10.4.2 图形的变换	251
10.4.3 图形的保存	252
习题	253

第 11 章 Visual Basic 数据库应用开发	255
11.1 数据库概述	255
11.1.1 数据库的概念	255
11.1.2 关系数据库	256
11.1.3 SQL 数据查询	257
11.2 Visual Basic 数据管理器	259
11.2.1 启动可视化数据管理器	259
11.2.2 新建数据库	260
11.2.3 打开数据库	260
11.2.4 添加数据库表	260
11.2.5 添加表索引	262
11.2.6 数据库维护	263
11.3 Visual Basic 中的数据访问方法	264
11.4 Data 控件	266
11.4.1 Data 控件的属性	266
11.4.2 数据绑定控件	267
11.4.3 Data 控件的事件与方法	269
11.4.4 记录集对象	270
11.4.5 使用代码操作数据库记录	273
11.5 ADO 数据控件	276
11.5.1 ADO 模型	276
11.5.2 ADO 数据控件	280
11.6 数据报表	283
11.6.1 数据环境设计器	283
11.6.2 建立数据报表	285
11.7 数据库应用开发实例	286
11.7.1 系统需求分析	286
11.7.2 系统设计	287
11.7.3 系统实现	288
习题	307
附录 1 Visual Basic 常用内部函数	309
附录 2 Visual Basic 常用属性和方法	317
附录 3 常见颜色多种等价表达对照表	325
参考文献	326

第 1 章 Visual Basic 的初步认识

Visual Basic 是在原有的结构化 BASIC 语言基础上发展而来的, Visual 指的是开发图形用户界面的方法, Basic 指的是广为流行的 BASIC 语言。Visual Basic 继承了原来 BASIC 语言简单易学的特点, 入门比较容易。可视化编程环境的“所见即所得”功能, 使界面设计非常直观方便, 把程序设计人员从繁琐复杂的界面设计中解脱出来。本章将介绍 Visual Basic 的演变过程、Visual Basic 集成开发环境、事件驱动的编程思想, 配合简单程序设计的需要, 介绍 Visual Basic 中几个最常用的对象以及建立一个 Visual Basic 程序的方法, 从而帮助读者建立对 Visual Basic 的初步认识。

1.1 从 BASIC 到 Visual Basic 的演变

提及 Visual Basic 语言, 首先要简要介绍 BASIC 语言。BASIC 是 Beginner's All-purpose Symbolic Instruction Code(初学者通用符号指令代码)的缩写, 是国际上广泛使用的一种计算机高级语言。

BASIC 的发展大致经历了基本 BASIC、高级 BASIC 和结构化 BASIC 等阶段。BASIC 诞生于 1964 年, 它只有十几条语句, 简单易学, 成为计算机入门的主要学习语言之一。20 世纪 70 年代中期, 随着计算机硬件技术的发展, 微型计算机诞生并得到普及, BASIC 语言也随之得到了完善, 功能有了较大的扩充, 出现了高级 BASIC, 代表性的版本有 GWBASIC、BASICA 等。20 世纪 80 年代中期, 随着结构化程序设计的需要, 结构化的 BASIC 应运而生, 代表性的版本有 True BASIC、Quick BASIC 和 Turbo BASIC 等。

1988 年 Microsoft 公司推出 Windows 操作系统, 开创了操作系统从 DOS 平台到 Windows 平台的革命性转变。与此同时用户也自然希望设计出具有 Windows 图形用户界面 (Graphical User Interface, GUI) 的应用程序。可视化程序设计语言很好地适应了这种需要。在可视化开发环境中, 不需要编写大量代码去描述界面元素的外观和位置, 而只要把预先建立的对象添加到屏幕上即可。

1991 年 Microsoft 公司推出了 Visual Basic。Visual Basic 由结构化 BASIC 语言发展而来。从 BASIC 到 Visual Basic 的变化是质的飞跃, 这种变化不仅仅是语言功能的大大增强, 更主要的是程序设计方式以及程序界面类型的改变。BASIC 语言是基于过程的程序设计语言, 而 Visual Basic 是基于对象的事件驱动机制的程序设计语言。BASIC 语言的编程界面是字符界面, 设计的程序是基于 DOS 平台的字符界面程序, 而 Visual Basic 的程序开发界面是可视化的图形界面, 开发的应用程序也是 Windows 图形界面程序。

1991 年推出的 Visual Basic 1.0 存在一些缺陷,功能也相对少一些。经过 Microsoft 公司的不断努力,于 1992 年、1993 年、1995 年、1997 年和 1998 年相继推出了 Visual Basic 2.0、Visual Basic 3.0、Visual Basic 4.0、Visual Basic 5.0 和 Visual Basic 6.0 共 5 个版本,功能在不断增强。Visual Basic 6.0 推出之后,深受用户欢迎,目前仍被广泛使用。2002 年 Microsoft 公司正式推出了 Visual Basic . Net,从 Visual Basic 6.0 到 Visual Basic . Net 又一次大的变化,在概念上、框架上和编程方式上都有了变化,但这种变化没有从 BASIC 到 Visual Basic 的变化那么大。掌握 Visual Basic 之后,再学习 Visual Basic . Net 就不是很难的事了,而且绝大部分基于 Visual Basic 6.0 开发的程序,很容易升级成 Visual Basic . Net 程序。由于 Visual Basic 具有简单易学的特点,所以成为人们学习程序设计的入门语言,也是许多高校首选的教学用程序设计语言。基于这些原因,本书仍以 Visual Basic 6.0 为蓝本来介绍程序设计的基本知识和方法。

从功能上来区分,Visual Basic 6.0 有 3 种不同版本,即学习版(Learning)、专业版(Professional)和企业版(Enterprise)。3 种版本各有特点:

①学习版提供了 Visual Basic 6.0 所有的内部控件,具备建立 Windows 应用程序的大部分工具,能够轻松地开发 Windows 应用程序。

②专业版包含了学习版中的所有工具和功能,并提供了开发复杂应用程序时所需的工具,包含了多种 ActiveX 控件等。

③企业版包含了专业版的全部内容,并提供了自动化管理器、部件管理器、数据库管理工具等,适用于开发分布式应用程序。

1.2 Visual Basic 集成开发环境

1.2.1 Visual Basic 的启动与退出

1. Visual Basic 的安装

在启动 Visual Basic 之前,首先要安装 Visual Basic。Visual Basic 6.0 的安装方法和其他 Windows 应用程序的安装方法类似。将 Visual Basic 系统安装盘放入光驱,一般情况下系统能自动运行安装程序,否则运行安装盘中的 setup.exe 文件。启动安装程序后,根据屏幕提示依次回答有关内容,便可完成系统安装。

2. Visual Basic 的启动

与启动 Windows 下的其他应用程序相似,启动 Visual Basic 6.0 的过程也十分简单。常用的方法是,在 Windows 桌面选择“开始”→“程序”→“Microsoft Visual Basic 6.0 中文版”命令。

在启动 Visual Basic 的过程中,会出现如图 1-1 所示的“新建工程”对话框。“新建工程”对话框有 3 个选项卡:新建、现存和最新。“新建”表示建立一个新的工程(程序),“现存”表示打

开一个现有的工程(程序)，“最新”表示打开一个最近所编写并存盘的工程(程序)。

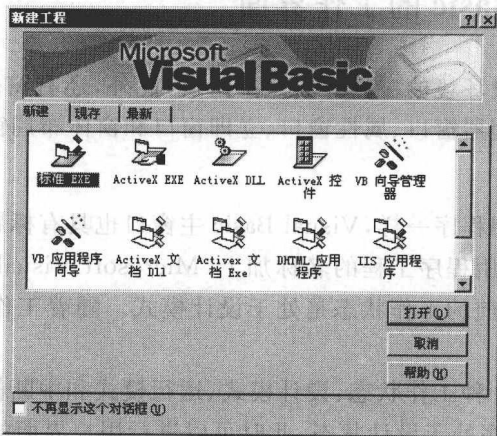


图 1-1 Visual Basic 6.0“新建工程”对话框

在“新建”选项卡下有多种程序类型,可以满足不同开发的需要,一般选择“标准 EXE”类型。在其他两个选项卡下则会出现打开程序的对话框,要求选择要打开的程序文件名。最后都是单击“打开”按钮,即可进入 Visual Basic 6.0 的集成开发环境界面,如图 1-2 所示。

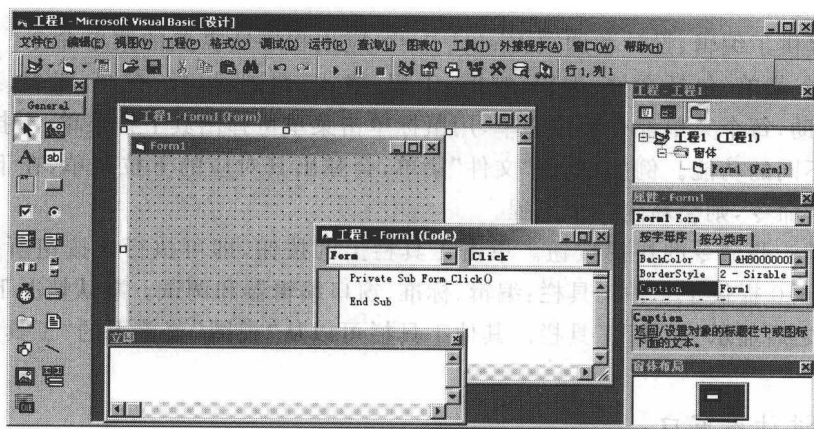


图 1-2 Visual Basic 6.0 集成开发环境

3. Visual Basic 的退出

退出 Visual Basic 有以下 3 种方法:

- ①单击 Visual Basic 6.0 主窗口的“关闭”按钮.
- ②选择“文件”→“退出”命令。
- ③按组合键 Alt + Q。

在退出 Visual Basic 6.0 集成开发环境之前,如果用户对当前程序已进行过修改且没有存盘,Visual Basic 将显示一个对话框,询问用户是否将其存盘,如果单击“是”按钮则对当前程序进行存盘,如果单击“否”按钮,则放弃存盘。

1.2.2 Visual Basic 的工作界面

Visual Basic 集成开发环境的主体界面是主窗口,此外,还有窗体设计器窗口、代码窗口、工具箱窗口、工程资源管理器窗口、属性窗口、立即窗口和窗体布局窗口等。

1. 主窗口

和其他 Windows 应用程序一样,Visual Basic 主窗口也具有标题栏、菜单栏和工具栏。

标题栏的内容就是应用程序工程的名称加上“Microsoft Visual Basic [设计]”字样内容。方括号中的“设计”表明当前的工作状态是处于设计模式。随着工作状态的不同,方括号中的文字也随之改变。

Visual Basic 6.0 有 3 种工作状态:设计模式、运行模式和中断模式。

- 设计模式:应用程序处于设计状态,此时可以进行用户界面的设计和代码的编写。
- 运行模式:程序处于运行状态,此时不可以编辑代码,也不可以编辑界面。
- 中断模式:应用程序运行暂时中断,此时可以编辑代码,但不可以编辑界面。按 F5 功能键或单击工具栏上的“继续”按钮可以使程序继续运行。单击工具栏上的“结束”按钮可以结束程序的运行。在此模式下,会打开立即窗口,在立即窗口内可以输入简单的命令,并立即执行。

菜单栏提供了编辑、设计和调试 Visual Basic 应用程序所需要的菜单命令。Visual Basic 6.0 共有 13 个菜单,包括文件、编辑、视图、工程、格式、调试、运行、查询、图表、工具、外接程序、窗口和帮助,每个菜单都有下拉菜单,用鼠标单击菜单可弹出其下拉菜单,下拉菜单中的每个命令执行不同的功能。例如,单击“文件”菜单,会弹出其对应的下拉菜单,在下拉菜单中选择“新建工程”命令,则会新建一个工程。

工具栏是一些命令的快捷按钮。单击工具栏上的按钮,即可执行该按钮所代表的操作。Visual Basic 6.0 提供了 4 种工具栏:编辑、标准、窗口编辑器和调试。默认情况下,启动 Visual Basic 6.0 之后显示“标准”工具栏。其他工具栏可以从“视图”菜单下的“工具栏”命令中打开或关闭。

2. 窗体设计器窗口

窗体设计器窗口是图 1-2 中标题栏内容为“工程 1-Form1(Form)”的窗口,它用于设计应用程序界面。每当新建一个工程时,Visual Basic 自动为该工程建立一个窗体(名称默认为 Form1),并为该窗体提供一个窗体设计器。从图 1-2 可以看出,当前被设计的窗体为 Form1,在设计程序时,用户通过更改该窗体窗口的属性、添加其他控件对象到窗体窗口上并设计好各控件的属性,就基本上“画出”了应用程序的界面。以后运行应用程序时,用户看到的界面就是这个窗体窗口,并通过其中的对象与程序进行交互对话,得到交互结果。每个窗体窗口必须有一个唯一的名字,建立窗口时 Visual Basic 默认给窗体取名为 Form1、Form2 等。

3. 代码窗口

代码窗口是图 1-2 中标题栏内容为“工程 1-Form1(Code)”的窗口。通过选择 Visual Basic 集成开发环境中的“视图”→“代码窗口”命令就可打开代码窗口。代码窗口是用来输入程序代码的地方,它显示当前窗体中的程序代码,并可对代码进行编辑修改。应用程序

的每个窗体或模块都有一个单独的代码窗口与之对应。可以打开多个代码窗口查看不同窗体、模块中的代码,并且可以在各个窗口间复制代码。

代码窗口左边的下拉列表框是“对象”下拉列表框。单击该下拉列表框会弹出下拉列表,列表中列出的项目有:(通用)、当前窗体的类名 Form、当前窗体中的所有控件名称和各级自定义菜单项名称。其中,无论窗体的名称改为什么,这里显示的都是固定的当前窗体类名 Form。

代码窗口右边的下拉列表框是“事件”下拉列表框。单击该下拉列表框会弹出下拉列表,列表中列出所选对象的所有事件名。当左边下拉列表框选定的项目是“(通用)”时,右边下拉列表框中列出的就是“(声明)”。

下拉列表框下方的空白区就是代码区,在其上可以输入和编辑程序语句代码,有以下两类代码:

- 通用声明代码:即左边下拉列表框选“(通用)”时,下面的代码区称为通用代码区,在这里编写的代码在整个窗体范围内有效。通常在这里编写一些窗体或模块级变量的声明语句、用户的自定义过程代码。

- 对象事件过程代码:选择一个对象和一个事件,下面就对应一个对象的事件过程,过程的首末两句 Visual Basic 已自动给出,程序员只需填写中间的操作性语句。

通用代码区和事件过程代码区之间、各事件过程代码区之间都用横线分隔,这样,结构化的分块更清晰。

4. 工具箱窗口

工具箱窗口存放了建立应用程序所需要的标准控件(也叫内部控件)。标准控件共有 20 个,另外还有一个指针,它不算控件,仅用于移动窗体和其他控件以及调整它们的大小。用户还可向工具箱添加 Windows 中已注册的其他外部控件。

工具箱窗口中显示的控件只是代表各控件的类,是空控件。利用工具箱提供的控件类,用户可以很方便地在程序窗体上添加一个具体控件(即控件的实例)。

要隐藏工具箱窗口,单击工具箱窗口的“关闭”按钮即可;要重新显示工具箱窗口,可以通过选择“视图”→“工具箱”命令来完成。

5. 工程资源管理器窗口

一个工程由多种类型的文件组成,如工程文件、窗体文件、标准模块文件等。在工程资源管理器窗口中,以树型目录结构的形式列出了当前工程中包含的所有文件。该窗口有 3 个按钮,自左至右分别为“查看代码”按钮、“查看对象”按钮和“切换文件夹”按钮,其功能分别如下:

- “查看代码”按钮:显示代码窗口。
- “查看对象”按钮:显示程序的窗体窗口。
- “切换文件夹”按钮:以文件夹形式或不以文件夹形式显示当前工程的所有文件。

如果关闭了工程资源管理器窗口,选择“视图”→“工程资源管理器”命令,或单击工具栏上的“工程资源管理器”按钮,可以打开该窗口。

6. 属性窗口

属性窗口通常在工程资源管理器窗口下面,由一个下拉列表框和一个两栏的表格组成。下拉列表框中列出当前工程的所有控件对象(包括窗体)的名称和所属的类别名(类名),下面

的两栏表格列出了所选对象的所有属性名、属性值。通过属性窗口可以对对象的某些属性值进行修改。

如果属性窗口被关闭,可以选择“视图”→“属性窗口”命令来打开。

7. 其他窗口

(1) 立即窗口

立即窗口常用于程序在调试期间输出中间结果,帮助用户在中断模式下测试表达式的值等。用户也可以在立即窗口中直接输入 Visual Basic 的命令并按 Enter 键,Visual Basic 会实时解释该命令,并立即响应。例如,用户可直接在该窗口中利用 Print 方法或直接在程序中用 Debug.Print 输出表达式的值。

(2) 窗体布局窗口

窗体布局窗口用于指定程序运行时各窗体的初始位置。在窗体布局窗口内,有一虚拟的显示器屏幕,在屏幕上有当前窗体的大小和位置的一个镜像,将鼠标定位到该镜像窗体上,按下鼠标左键,将其拖到显示器的所需位置即可,以后程序运行时,该窗体就会显示在真实屏幕的对应位置。

1.3 事件驱动编程机制

1.3.1 对象和类的概念

Visual Basic 在其编程系统中采用了面向对象和事件驱动的编程机制,提供了一种所见即所得的可视化程序设计方法。利用 Visual Basic 进行程序设计,首先要准确地理解对象和类的概念。

1. 对象

从一般意义上说,对象(Object)是指现实世界中具体存在的实体,如一个人、一件物品等。每一个对象都有反映自身特征的属性,如“人”作为对象有名字、身高、体重等属性。对象还有附属于它的行为(称为方法),如“人”的行为有演讲、听课、打球等。在对象身上可能会发生某种动作(称为事件),该动作发生时可能会触发相应的动作行为,如“电话铃”响起时“人”会去接电话。

在面向对象程序设计中,借助现实世界中对象的概念来表示组成程序的具体部件,这时,对象是一个将数据和处理该数据的过程(函数或子程序)“封装”在一起的一个程序部件。Visual Basic 中的对象与面向对象程序设计中的对象在概念上是一样的,但在使用上有很大区别。在面向对象程序设计中,对象由程序员自己设计,而在 Visual Basic 中,对象一般是由系统设计好的,可以直接使用和对其进行操作。

前面介绍了窗体窗口和工具箱窗口,用工具箱窗口中的控件图标可以在窗体上设计界面,窗体和控件就是 Visual Basic 中的对象。这些对象也具有描述其特征的属性、反映其动作行