

高校计算机教学系列教材

Visual Basic 程序设计方法

邓文新 张桂香 主编

主 审 曲伟建

北京航空航天大学出版社

内 容 简 介

本书以 Visual Basic 6.0 版为蓝本,并结合作者多年讲授程序设计理论课与实验课的经验,对 Visual Basic 的基础部分作了详细介绍,而对于诸如菜单设计及窗口界面、多媒体编程等则以实例的形式展示给读者,使读者能够在逐渐深入、循序渐进中接受 Visual Basic,了解 Visual Basic,并最终达到灵活运用 Visual Basic 的目的。本书既可作为大学教材,也可作为计算机应用、开发技术人员的参考书。

图书在版编目(CIP)数据

Visual Basic 程序设计方法/邓文新等主编. —北京:
北京航空航天大学出版社,2005.4

(高校计算机教学系列教材)

ISBN 7-81077-617-7

I. V… II. 邓… 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(2005)第 011020 号

Visual Basic 程序设计方法

邓文新 张桂香 主编

责任编辑 许传安

责任校对 陈 坤

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:(010)82317024 传真:(010)82328026

<http://www.buaapress.com.cn> E-mail:bhpress@263.net

北京市松源印刷有限公司印装 各地书店经销

*

开本:787×1092 1/16 印张:16.5 字数:422 千字

2005 年 4 月第 1 版 2005 年 4 月第 1 次印刷 印数:5 000 册

ISBN 7-81077-617-7 定价:23.00 元



总 前 言

科教兴国,教育先行,在全国上下已形成共识。在教育改革过程中,出现了多渠道、多形式、多层次办学的局面。同时,政府逐年加大教育的投入力度。教育发展了,才能有效地提高全民族的文化、科学素质,使我们中华民族屹立于世界民族之林。

计算机科学与技术的发展日新月异,其应用领域迅速扩展,几乎无处不在。社会发展的需求,促使计算机教育生气蓬勃。从普通高校的系统性教学,到远距离的电视、网上教学;从全面讲述,到不同应用领域的、星罗棋布的培训班;从公办的到民办的;从纸介教材到电子教材等等,可以说计算机教学异彩纷呈。要进行教学,就必须有教材。

面对我们这么大的国家和教学形势,在保证国家教学基本要求的前提下,应当提倡教材多样化,才能满足各教学单位的需求,使他们形成各自的办学风格和特色。为此,我们组织北京工业大学、北京航空航天大学、北京理工大学、南开大学、天津工业大学等高校的有丰富教学经验的教师编写了计算机教学系列教材,将陆续与师生见面。

系列教材包括以下各项。

(一) 基础理论:离散数学等。

(二) 技术基础:电路基础与模拟电子技术;数字逻辑基础;计算机组成与体系结构;计算机语言(拼盘,选择使用),包括 C++ 程序设计基础、Visual Basic 程序设计基础、Matlab 程序设计基础、Java 程序设计基础、Delphi 语言基础、汇编语言基础等;数据结构;计算机操作系统基础;计算方法基础;微机与接口技术;数据库技术基础等。

(三) 应用基础:计算机控制技术;网络技术;软件工程;多媒体技术等。

(四) 技术基础扩展:编译原理与编译构造;知识工程——网络计算机环境下的知识处理等。

(五) 应用基础扩展:计算机辅助设计;单片机实用基础;图形、图像处理基础;传感器与测试技术;计算机外设与接口技术等。

本系列教材主要是针对计算机教学编写的,供普通高校、社会民办大学、高等职业学校、业余大学等计算机本科或专科选用。其中一部分也适合非计算机专业本科教学使用。在这些教材的内容简介或前言中对使用范围均作了说明。

本系列教材在编写时,注重以下几点:(1) 面对计算机科学与技术动态发展的现实,在内容上应具有前瞻性;(2) 面对学以致用,既有系统的基础知识,又具有应用价值的实用性;(3) 具有科学性、严谨性。另外,力求使有限的版面具有最大的信息量,以使读者得到实惠。

能否实现这些愿望,只有师生在教学实践中评价。我们期望得到师生的批评和指正。

高校计算机教学系列教材编委会



高校计算机教学系列教材编委会成员

主 任:赵沁平

副 主 任(常务):陈炳和

顾 问:麦中凡

委 员(以姓氏笔划为序):

吕景瑜(北工大教授)

乔少杰(社长,研究员)

麦中凡(北航教授,教育部工科计算机基础教学指导委员会副主任、中专计算机
教学指导委员会顾问)

苏开娜(北工大教授)

陈炳和(北工大教授)

张鸿宾(北工大博导)

郑玉明(北工大副教授)

金茂忠(北航博导)

赵沁平(北航博导,国务院学位办主任)



前 言

随着 Visual Basic 在解决问题中的广泛应用,引起了计算机应用开发者和高校师生的学习兴趣和掌握、使用 Visual Basic 的热潮。近年来也有部分院校把 Visual Basic 程序设计作为在校生的首选计算机语言课。

本书从安装开始,向读者详细地介绍了 Visual Basic 比较基础的内容。其中第 1、第 2 章是学习 Visual Basic 的入门知识,如开发环境的介绍;对象、属性、事件和方法的概念等;第 3、第 4、第 5、第 6 章是 Visual Basic 的语言基础,包括各种常用语句、函数等,是 Visual Basic 的最基础、最重要的内容;第 7 章介绍了菜单设计及多窗口界面;第 8 章是 Visual Basic 中关于文件、目录的操作;第 9 章是多媒体编程,主要以实例的形式给读者展示了 Visual Basic 在多媒体方面的强大功能;第 10 章简单地介绍了数据库编程。

一般 Visual Basic 编程书籍往往给读者的感觉像是一本手册,不注重最基础的部分,导致初学者无从下手。本书作为教材侧重于 Visual Basic 语言的入门,由浅入深,具有叙述详细、浅显易懂、注重基础之特点,特别适于计算机编程的初学者使用。

本书所有实例均以 Visual Basic 6.0 版为蓝本,从安装开始,一步步使读者从不会编程到能够独立使用 Visual Basic 语言编制优良的、具有 Windows 风格的程序;对于计算机专业的读者,本书又可以作为学习 Visual Basic 多媒体及数据库编程的教材。

全书由邓文新、张桂香主编,杨波、李晶、刘艳菊、李敬有等参编。齐齐哈尔大学曲伟建副校长审校了全书,并提出了宝贵意见。由于编者水平所限,书中疏漏之处,敬请读者批评指正。

编 者

2004 年 8 月



目 录

第 1 章 Visual Basic 简介及开发环境	1
1.1 Visual Basic 的主要特性	1
1.2 Visual Basic 安装、进入及退出	2
1.2.1 Visual Basic 的系统要求及安装	2
1.2.2 进入 Visual Basic	3
1.2.3 退出 Visual Basic	4
1.3 Visual Basic 的开发环境及其组成元素	5
1.3.1 主窗口	5
1.3.2 窗体设计器窗口及程序代码窗口	8
1.3.3 属性窗口	9
1.3.4 工具箱及工程资源管理器窗口	10
1.4 帮助文件的使用	12
1.4.1 使用 MSDN Library 获得帮助	12
1.4.2 上下文相关帮助	12
1.5 VB 应用程序的开发步骤及实例分析	13
1.5.1 VB 应用程序的开发步骤	13
1.5.2 实例分析	14
习题 1	17
第 2 章 对象与常用控件	18
2.1 对 象	18
2.1.1 Visual Basic 的对象	18
2.1.2 对象的属性	18
2.1.3 对象的事件	19
2.1.4 对象的方法	20
2.2 窗 体	20
2.2.1 窗体的结构	20
2.2.2 窗体的属性	21
2.2.3 窗体的事件	22
2.3 控 件	24
2.3.1 认识内部控件	24
2.3.2 窗体和控件的基本属性	25
2.3.3 标 签	31
2.3.4 文本框	31
2.3.5 命令按钮	31



2.4 常用事件及方法	34
2.4.1 与鼠标有关的事件	34
2.4.2 与键盘有关的事件	35
2.5 程序调试	37
2.5.1 错误类型	37
2.5.2 调试和纠错	38
习题 2	39
第 3 章 Visual Basic 语言基础	40
3.1 编码规则	40
3.1.1 语句书写自由	40
3.1.2 在 VB 代码中,字母不区分大小写	40
3.1.3 在 VB 中可以使用注释来增加程序的可维护性	41
3.1.4 保留行号与标号	41
3.1.5 VB 中的标识符	41
3.2 变量和常量	41
3.2.1 变量	41
3.2.2 常量	48
3.3 运算符与表达式	50
3.3.1 运算符	50
3.3.2 表达式	54
3.4 常用内部函数	55
3.4.1 数学函数	55
3.4.2 转换函数	56
3.4.3 字符串函数	57
3.4.4 日期时间函数	58
3.4.5 Shell()函数	59
习题 3	60
第 4 章 基本的流程控制	62
4.1 顺序结构	62
4.1.1 赋值语句	62
4.1.2 输入函数	64
4.1.3 MsgBox()函数和 MsgBox 过程	66
4.1.4 输出语句	68
4.2 选择结构	74
4.2.1 If...Then 语句(单分支结构)	74
4.2.2 If...Then...Else 语句(双分支结构)	76
4.2.3 If...Then...ElseIf 语句(多分支结构)	77
4.2.4 Select Case 语句	79
4.2.5 条件函数	83



4.3 循环结构.....	83
4.3.1 For 循环语句.....	83
4.3.2 Do...Loop 循环语句.....	86
4.3.3 当循环结构.....	88
4.3.4 循环的嵌套.....	89
4.4 辅助的控制语句.....	91
4.4.1 GoTo 语句.....	91
4.4.2 End 语句.....	92
4.4.3 Stop 语句.....	92
4.4.4 With 语句.....	92
习题 4.....	93
第 5 章 数 组	97
5.1 数组的概念.....	97
5.2 静态数组与动态数组的定义.....	97
5.2.1 静态数组的定义.....	98
5.2.2 动态数组的定义.....	99
5.3 数组的基本操作.....	100
5.3.1 数组的引用.....	100
5.3.2 数组的输入、输出.....	100
5.3.3 数组的初始化.....	101
5.3.4 For Each...Next 语句.....	101
5.3.5 数组的清除和释放.....	102
5.4 控件数组.....	103
5.4.1 控件数组的概念.....	103
5.4.2 建立控件数组.....	104
5.5 列表框和组合框.....	106
5.5.1 列表框.....	106
5.5.2 组合框.....	110
5.6 数组应用举例.....	111
5.6.1 自定义类型数组的应用.....	111
5.6.2 数组在数学及统计中的应用.....	113
5.6.3 数组在排序中的应用.....	114
习题 5.....	117
第 6 章 过 程	120
6.1 事件过程.....	120
6.2 通用过程.....	122
6.2.1 子程序过程.....	122
6.2.2 函数(Function)过程.....	125
6.3 参数传递.....	127



6.3.1	虚实结合原则	127
6.3.2	虚实结合方式	128
6.3.3	数组参数	130
6.4	变量的作用域和生存期	132
6.4.1	局部变量	133
6.4.2	窗体级或模块级变量	134
6.4.3	全局变量	135
6.5	递归	136
6.5.1	递归的概念	136
6.5.2	递归调用	136
	习题 6	138
第 7 章 菜单设计及多窗体界面		141
7.1	菜单设计	141
7.1.1	菜单编辑器	141
7.1.2	简单的菜单设计	143
7.1.3	建立菜单应用程序	145
7.1.4	敏感式菜单	148
7.1.5	创建弹出式菜单	149
7.2	多窗体界面的设计	151
7.2.1	有关窗体的语句和方法	152
7.2.2	多窗体的程序设计	152
7.3	多文档界面的设计	160
7.3.1	多文档界面的特点	161
7.3.2	与 MDI 有关的属性、方法和事件	161
7.3.3	建立 MDI 应用程序	162
	习题 7	165
第 8 章 文件及文件管理		167
8.1	文件系统的 3 个控件	167
8.1.1	驱动器列表框控件	167
8.1.2	目录列表框控件	168
8.1.3	文件列表框控件	171
8.1.4	目录和文件的操作语句	173
8.2	文件的存取方式	175
8.2.1	顺序文件的使用	175
8.2.2	随机文件的使用	181
8.2.3	二进制文件的使用	186
8.3	文件系统对象	189
8.3.1	创建文件系统对象	189
8.3.2	获取驱动器信息	190



8.3.3 管理和操作文件夹	191
习题 8	197
第 9 章 图形与多媒体程序设计	199
9.1 图形与图片	199
9.1.1 图 形	199
9.1.2 图 像	201
9.2 多媒体 MCI 控件	205
9.2.1 播放 WAV, AVI 文件	205
9.2.2 播放音频 CD 中的音乐	209
9.3 使用外来多媒体控件	211
习题 9	212
第 10 章 数据库编程	213
10.1 数据库基本概念	213
10.2 数据访问方法	214
10.3 Data 控件	215
10.3.1 Data 控件简介	215
10.3.2 Data 控件的属性	215
10.3.3 数据绑定控件	218
10.3.4 Data 控件的方法	223
10.3.5 Data 控件的事件	230
10.4 ADO 数据控件	232
10.4.1 什么是 ADO	232
10.4.2 ADO 数据控件简介	233
10.4.3 ADO 数据控件的属性和事件	233
10.4.4 ADO 数据控件的使用	237
10.5 数据报表	239
习题 10	244
附录一 参考答案	245
附录二 VB 常用函数索引表	247
参考文献	250



第 1 章

Visual Basic 简介及开发环境

Visual Basic 是一种功能强大的程序设计语言。它提供了基于对象(对象的概念在第 2 章介绍)的可视化程序开发工具,用户无须编写大量代码去描述界面的外观,就能够以迅速、简捷的方法开发设计应用程序。本章着重介绍 Visual Basic 的主要特点、安装、进入和退出,并将对 Visual Basic 的集成开发环境进行详细讲述。



1.1 Visual Basic 的主要特性

Basic 语言是在 20 世纪 60 年代诞生于美国。由于 Basic 语言具有简单易学、效率高的特点,所以在国际上广泛流行,并得到了不断的发展。20 世纪 80 年代,以 Windows 操作系统为代表的图形用户界面逐渐被广大用户接受。为了简化 Windows 应用程序的开发,微软公司于 1991 年推出了 Visual Basic。它是在 Basic 语言的基础上研制而成的很有发展前途的高级程序设计语言,继承了 Basic 语言的特点,并且功能强大,可用于开发 Windows 环境下的各类应用程序,是一种可视化的、面向对象和采用事件驱动的方式,利用 Windows 环境下丰富的图形界面的程序设计语言。

随着版本的提高,Visual Basic 的功能也越来越强大。5.0 版以后,Visual Basic 推出了中文版,与以前各版相比,其功能有了质的飞跃,成为 32 位、全面支持面向对象的大型程序设计语言。在推出 6.0 版时,Visual Basic 又在数据访问、控件、语言、向导及 Internet 支持等方面增加了许多新的功能。

Visual Basic 作为目前最简单、最容易使用的 Windows 应用程序开发语言,有以下主要特点。

1) 可视化编程设计工具 Visual Basic 提供了可视化设计工具,把 Windows 界面设计的复杂性“封装”起来,开发人员不必为开发界面设计而编写大量程序代码,只需要按设计要求的屏幕布局,用系统提供的工具,在屏幕上画出各种“部件”,即图形对象,并设置这些图形对象的属性。Visual Basic 自动产生界面设计代码,程序设计人员只须要编写实现程序功能的那部分代码,从而可以大大提高程序设计的效率。

2) 面向对象的程序设计 Visual Basic 4.0 版,特别是 5.0 版以后支持面向对象的程序设计。Visual Basic 是应用面向对象的程序设计方法(OOP),把程序和数据封装起来作为一个对象,并为每个对象赋予应有的属性,使对象成为实在的东西。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 Visual Basic 工具,在界面上自动生成对象的程序代码,并封装起来。每个对象以图形方式显示在界面上,都是可视的。

3) 结构化程序设计语言 Visual Basic 是在 Basic 和 Quick Basic 语言的基础上发展起来



的,具有高级程序设计语言的语言结构,接近于自然语言和人类的逻辑思维方式,其语句简单易懂;其编译器支持彩色代码,可以自动进行语法检查,同时具有功能强,且使用灵活的调式器和编译器。

Visual Basic 是解释型语言,在输入代码的同时,解释系统将高级语言分解翻译成计算机可以识别的机器指令,并判断每个语法的语法错误。在设计 Visual Basic 程序的过程中,随时可以运行程序,而在整个应用程序设计好之后,Visual Basic 又呈现其编译型语言的特点,可以编译生成可执行文件(.EXE),脱离 Visual Basic 环境,直接在 Windows 环境下运行。

4) 事件驱动编程机制 Visual Basic 通过事件来执行对象的操作。一个对象可能会产生多个事件,每个事件都可以通过一段程序来响应,例如,命令按钮是一个对象,将会产生一个“单击”(Click)事件,而在产生该事件时将执行一段程序,用来实现指定的操作。

在用 Visual Basic 设计大型应用软件时,不必建立具有明显开始和结束的程序,而是编写若干个微小的子程序,即过程。这些过程分别面向不同的对象,由用户操作引发某个事件来驱动执行某种特定的功能,或者由事件驱动程序调用通用过程来执行指定的操作。这样可以方便编程人员,提高效率。

5) 访问数据库 Visual Basic 系统具有很强的数据库管理功能。利用数据控件和数据库管理窗口,可以直接建立或处理 Microsoft Access 格式的数据库,并提供了强大的数据存储和检索功能。同时,Visual Basic 还能直接编辑和访问其它外部数据库,如 dBASE, FoxPro, Paradox 等。这些数据库格式都可以用 Visual Basic 编辑和处理。

Visual Basic 提供开放式数据库连接(open database connectivity),即 ODBC 功能。它可通过直接访问或建立连接的方式使用,并操作后台大型网络数据库,如 SQL Server, Oracle 等。在应用程序中可以使用结构化查询语言 SQL 数据标准,直接访问 Server 上的数据库,并提供了简单的面向对象的库操作指令、多用户数据库访问的加锁机制和网络数据库的 SQL 的编程技术,为单机上运行的数据库提供了 SQL 网络接口,以便在分布式环境中快速而有效地实现客户服务器(Client/Server)方案。

以上介绍了 Visual Basic 的一些主要特点。除上述特点外,Visual Basic 还提供了其它一些功能,包 Active 技术,使开发人员摆脱了特定语言的束缚,可以方便地使用其它应用程序提供的功能;网络功能可以使 Web 页面设计者动态地创建和编辑页面,使用户开发多功能的网络软件;应用程序向导功能,可以快捷地创建不同类型、不同功能的程序;Help 联机帮助功能,提供了大量用户所需的帮助信息,方便用户学习和使用。

1.2 Visual Basic 安装、进入及退出

1.2.1 Visual Basic 的系统要求及安装

Visual Basic 可以在多种操作系统下运行,包括 Windows 9x, Windows 2000, Windows NT 4.0 和 Windows XP 等。

Visual Basic 有三种版本,即专业版、学习版和企业版。企业版大约占用 140 MB 硬盘空



第三种方法

使用【开始】菜单中的【运行】命令。操作如下：

单击【开始】按钮，弹出一个菜单，然后单击【运行】命令，将弹出一个对话框。

在【打开】栏内输入 Visual Basic 启动文件的名称(包括路径)，例如 c:\vc98\VB6.exe。

单击【确定】按钮，即可启动 Visual Basic，看到如图 1.1 所示的窗口。

第四种方法

建立启动 Visual Basic 的快捷方式(具体操作见有关 Windows 操作资料)。

进入 Visual Basic 后，在图 1.1 所示的窗口中列出了建立 Visual Basic 的应用程序类型，初学者只要选择默认的【标准.EXE】即可。在该窗口中有三个选项卡：

- 【新建】：建立新工程；
- 【现存】：选择和打开现有的工程；
- 【最新】：列出最近使用过的工程。

单击【新建】选项卡后，就可创建该类型的应用程序，进入 Visual Basic 应用程序集成开发环境，如图 1.2 所示。



图 1.2 Visual Basic 应用程序集成开发环境

1.2.3 退出 Visual Basic

为了退出 Visual Basic，可先打开【文件】菜单，并执行其中的【退出】命令(或按 Alt+Q)。如果当前程序已修改过，并且没有存盘，系统将显示一个对话框，询问用户是否将其存盘。此时选择【是】按钮则存盘，选择【否】按钮则不存盘。在上述两种情况下，都将退出 Visual Basic，回到 Windows。



1.3 Visual Basic 的开发环境及其组成元素

用上述所介绍的任何方法启动 Visual Basic 后,将首先显示版权屏幕,说明此份拷贝的使用权属于谁。稍后,显示【新建工程】对话框,如图 1.1 所示。【新建】选项卡对话框显示了可以在 Visual Basic 中使用的工程类型,即可以建立的应用程序。其中【标准 EXE】用来建立标准的 EXE 工程,本书将只讨论这种工程类型。

在对话框中选择建立的工程类型(例如【标准 EXE】),然后单击【打开】按钮,即进入 Visual Basic 集成环境,如图 1.2 所示。

Visual Basic 提供了许多种工程类型以满足不同的需要。每次启动 Visual Basic 时,都要显示【新建工程】对话框。在一般情况下,可能主要使用【标准 EXE】工程,因此,没有必要在每次启动 Visual Basic 时显示该对话框。这可以通过选择(如图 1.1 左下角)【不再显示这个对话框】选项来实现(单击小方框,使框内有【√】)。选择该项后,再选择【标准 EXE】,然后单击【打开】按钮,进入 Visual Basic 开发环境。以后启动 Visual Basic 时,即不再显示【新建工程】对话框,直接进入开发环境。

在默认情况下,Visual Basic 的集成开发环境为传统的 Windows MDI(多文档界面)方式;此外,也可以用 SDI(单文档界面)方式启动 Visual Basic 设计。在多数情况下,使用 SDI 方式会更方便。为了把编程环境变成为 SDI 方式,可执行【工具】菜单中的【选项】命令,打开【选项】对话框,选择【高级】选项卡,在对话框中选择【SDI 开发环境】选项,然后单击【确定】按钮。这样设置后,退出 Visual Basic,然后重新启动,即可按 SDI 方式进入 Visual Basic 集成开发环境。

从图 1.2 中可以看出,启动 Visual Basic 后,屏幕上分为若干部分,包括标题栏、工具栏、工具箱、窗体设计器窗口、工程管理器、窗体布局窗口和属性窗口。为了能清楚地看到每个部分,这里对原来的各部分进行了缩放和重新排列。读者在启动自己的 Visual Basic 后所看到的各部分的排列情况可能与图 1.2 所示的有微小差别,一些窗口会重叠。实际上,和 Windows 下的窗口一样,集成开发环境中的每个窗口都可以在屏幕上移动、缩小、放大或关闭。此外,Visual Basic 保存上一次使用时屏幕上各部分最后的排列方式,并作为下一次启动 Visual Basic 后的屏幕布局。

1.3.1 主窗口

主窗口又称设计窗口。启动 Visual Basic 后,主窗口位于集成环境的顶部,该窗口由标题栏、菜单栏和工具栏组成(参见图 1.2)。

1. 标题栏

标题栏是屏幕顶部的部分,其上显示的是应用程序的名字。用户与标题栏之间的交互关系由 Windows 来处理,而不是由应用程序处理。启动 Visual Basic 后,标题栏中显示的信息为:【工程 1—Microsoft Visual Basic[设计]】,方括号中的【设计】表明当前的工作状态是【设计阶段】。随着工作状态的不同,方括号中的信息也随之改变,可能会是【运行】或 Break,分别代表运行阶段或中断阶段。这 3 个阶段也分别称为“设计模式”、“运行模式”和“中断模式”。

2. 菜单栏

在标题栏的下面是集成环境的主菜单。菜单栏中的菜单命令提供了开发、调试和保存应用程序所需要的工具。Visual Basic 中文版的菜单栏共有 13 个菜单项,即【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】(见图 1.2)。每个菜单项含有若干个菜单命令,执行不同的操作。例如,单击【文件】菜单,就可以打开文件菜单,如图 1.3 所示。打开菜单后,如果单击【打开工程】命令,就可以打开已有的工程文件;如果单击【工程另存为】命令,就可以保存文件等等。

菜单中的命令分为两种类型:一类是可以直接执行的命令,这类命令的后面没有任何信息(例如【保存工程】);另一类在命令后面带有省略号(例如【打开工程】),需要通过打开【对话框】来执行。在单击后一类命令后,屏幕上将显示一个对话框,利用该对话框可以执行各种有关的操作。在【文件】菜单中,【新建工程】、【保存工程】等是可以直接执行的命令,而【打开工程】、【工程另存为】等命令必须通过对话框来执行。此外,从【文件】菜单可以看出,在有些命令的后面还带有其它信息,例如:

打开工程... Ctrl+O

保存 Form1 Ctrl+S

其中 Ctrl+O 叫做快捷键。在菜单中,快捷键列在相应的菜单命令之后,与菜单命令具有相同的作用。使用快捷键方式,不必打开菜单就能执行相应的菜单命令。例如,按 Ctrl+O,可以立即执行【打开工程】命令。注意,只有部分菜单命令能通过快捷键执行。



图 1.3 文件菜单

上面介绍了通过鼠标和快捷键执行菜单命令的方法。除使用鼠标外,也可以通过键盘执行菜单命令。只有在打开菜单后,才能选择所需要的命令,执行相应的操作。Visual Basic 提供了多种打开菜单和选择菜单的方法,用户可以根据自己的兴趣或习惯选用其中的一种。

第一种方法,步骤如下。

按 F10 或 ALT 键激活菜单栏,此时第一个菜单项【文件】被加上一个浅色的框。

按菜单项后面括号中的字母键打开菜单,下拉显示该菜单项的命令。菜单被打开后,各菜单命令后面的括号内都有一个字母。

按菜单命令后面括号中的字母键,即可执行相应的命令。

第二种方法,步骤如下。

按 F10 或 ALT 键激活菜单栏,此时第一个菜单项【文件】被加上一个浅色的框。

用“→”键或“←”键把条形光标移到需要打开的菜单项上,按回车键打开该菜单项。

菜单被打开后,条形光标覆盖在第一个或上一次执行的菜单命令上。用“↑”键或“↓”键把条形光标移到所需要的命令上,按回车键即可执行条形光标所在位置的菜单命令。

第三种方法,步骤如下。

按下 Alt 键,不要松开,接着按需要打开的菜单项后面括号中的字母键,然后松开(Alt 键



接着松开),该菜单即被打开。

按菜单命令后面括号中的字母键,即可执行指定的菜单命令。

例如,为了执行【文件】菜单中的【打开工程】命令,按如下操作:按住 Alt 键,不要松开,接着按 F 键,再按 O 键,即可执行【文件】菜单中的【打开工程】命令。我们把上述过程记作:Alt + F|O。

除上面 3 种方法外,有些菜单命令还可以通过快捷键执行(见前)。对于没有快捷键的菜单命令,只能通过上面 3 种方式执行。

菜单被打开后,在屏幕上显示相应的菜单命令。如果打开得不适当或不需要的菜单,或者执行菜单命令是打开了不需要的对话框,可以按 Esc 键取消菜单。

Visual Basic 应用程序的编辑、编译、链接、运行、调试及文件的打开、保存等都可以通过相应的菜单命令来实现,其用法与上面介绍的类似。

3. 工具栏

Visual Basic 提供了 4 种工具栏,包括【编辑】、【标准】、【窗体编辑器】和【调试】,并可根据需要定义用户自己的工具栏。在一般情况下,集成环境中只显示标准工具栏,其它工具栏可以通过【视图】菜单中的【工具栏】命令打开(或关闭)。每种工具栏都有固定和浮动两种形式。把光标移到固定形式工具栏中没有图标的地方,按住鼠标左键向下拖动,或者双击工具栏左端的两条浅色竖线,即可把工具栏变成浮动的;而如果双击浮动工具栏的标题条,则可变为固定工具栏。

固定形式的标准工具栏位于菜单栏的下面,即主窗口的上部。它以图标的形式提供了部分常用菜单命令。只要用单击代表某个命令的图标按钮,就能直接执行相应的菜单命令。标准工具栏中有 21 个图标,代表 21 种操作,如图 1.4 所示。大多数图标都有与之等价的菜单命令。图 1.4 是浮动形式的标准工具栏。



图 1.4 工具栏

在工具栏的右侧还有两个栏,分别用来显示窗体的当前位置和大小,其单位为 twip(缇)。左边一栏显示的是窗体左上角的坐标,右边一栏显示的是窗体的长×宽。

twip 是一种与屏幕分辨率无关的计量单位,无论在什么屏幕上,如果画了一条 1 440 twip 的直线,打印出来是 1 英寸,也就是 1 英寸=1 440 twip,1 英寸=72 磅,1 磅=20 twip。这种计量单位可以确保在不同的屏幕上都能保持正确的相对位置或比例关系。在 Visual Basic 中,twip 是默认单位,可以通过 ScaleMode 属性改变。