

教育部考试中心制订的考试大纲配套用书



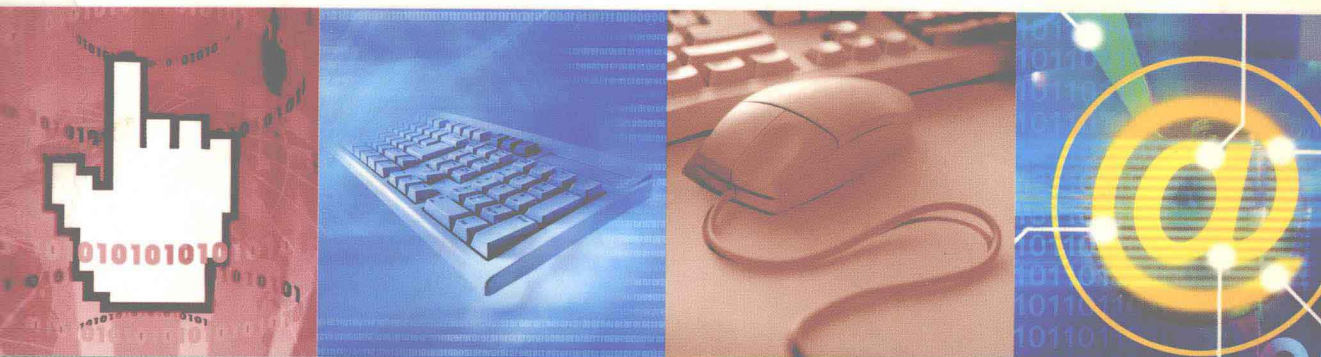
全国计算机 等级考试用书

二级教程

—Visual Basic语言程序设计

全国计算机等级考试用书编委会 组编

高延武 主编



世界知识出版社

根据教育部考试中心最新版《考试大纲》编写

全国计算机等级考试用书

二级教程

——Visual Basic 语言程序设计

全国计算机等级考试用书编委会 组编

主 编	高延武		
副主编	刘爱国	冯春辉	郭 爽
编 者	高延武	刘爱国	冯春辉
	郭 爽	梁海丽	李志梅
	赵丙辰	丁春芳	隋春荣

世界知识出版社

内容提要

本书是根据教育部考试中心制定的《全国计算机等级考试二级考试大纲(Visual Basic 语言程序设计)》的要求而编写,主要包括 Visual Basic 6.0 的编程环境、常用内部控件的功能和用法、控制结构、数组、过程、菜单、对话框、多重窗体程序设计技术和鼠标、键盘事件以及文件处理等。

本书体现了大纲的要求,讲解简明,概念准确清晰,例题丰富。可供各层面学生、教师、自学应试者阅读参考,对于应试人员复习应考有切实的指导意义。

图书在版编目(CIP)数据

二级教程. Visual Basic 语言程序设计 / 高延武等
编著. —北京:世界知识出版社, 2003. 7

全国计算机等级考试用书

ISBN 7-5012-2072-7

I. 二... II. 高... III. ①电子计算机—水平考试
—教材②BASIC 语言—程序设计—水平考试—教材
IV. TP3

中国版本图书馆 CIP 数据核字 (2003) 第 055004 号

责任编辑 / 郑志国

封面设计 / 易 红

责任出版 / 夏凤仙

出版发行 / 世界知识出版社

地 址 / 北京东城区干面胡同 51 号

邮政编码 / 100010

经 销 / 新华书店

开本印张 / 787×1092 毫米 16 开本 24 印张 567 千字

印 刷 / 北京大中印刷厂

版 次 / 2003 年 8 月第 1 版 2003 年 8 月第 1 次印刷

定 价 / 35.50 元

版权所有 翻印必究

前 言

为了适应计算机技术及应用的发展,教育部考试中心 2002 年全国计算机等级考试大纲进行了全面修订,全国计算机等级考试委员会审议通过了《全国计算机等级考试大纲(2002 年版)》。教育部考试中心推出的全国计算机等级考试是一种重视应试人员对计算机实际掌握能力的考试,为各单位工作人员计算机知识和能力水平提供了一种全国统一的、客观公正的等级标准。许多单位都把计算机等级考试水平作为干部录用、职务晋升、职称评定、上岗资格的重要依据之一。

随着计算机软硬件的发展和操作系统平台的升级,教育部考试中心决定在原有考试门类的基础上,增加 Windows 环境下可视化程序设计的内容,其中包括 Visual Basic 语言程序设计。鉴于社会的客观需求,作者根据《全国计算机等级考试二级大纲(Visual Basic 语言程序设计)》,结合多年从事计算机教学的经验,编写了这本计算机等级考试用书。

全书共分 12 章,主要内容包括:Visual Basic 程序开发环境,对象和事件驱动的概念,常用内部函数和表达式,数据输入输出,常用标准控件,基本控制结构,数组、过程调用,键盘和鼠标事件过程,菜单与对话框程序设计,多窗体程序设计,文件处理等。

全书内容紧扣考试大纲,适用于初学者学习,书中大量的例题有利于学习者对概念的深刻理解,只有理解了这些基本概念,才能灵活运用 Visual Basic 设计复杂的应用程序。本书既可以作为参加等级考试人员的应试教材,又可作为高等院校 Visual Basic 语言程序设计课程的教材。

本书第 1 章、第 2 章、第 9 章、附录由高延武、梁海丽编写,第 3 章、第 4 章、第 10 章由冯春辉、赵丙辰编写,第 5 章、第 6 章由刘爱国、李志梅编写,第 7 章、第 8 章、第 11 章、第 12 章由郭爽、丁春芳、隋春荣编写,高延武对本书进行了全面的审阅。

由于时间仓促,书中难免有不足之处,欢迎广大读者批评指正。

编 者

2003 年 8 月

目 录

第1章 Visual Basic 程序开发环境	1
1.1 Visual Basic 的版本和特点	1
1.1.1 Visual Basic 的版本	1
1.1.2 Visual Basic 的特点	2
1.2 Visual Basic 的启动与退出	3
1.2.1 Visual Basic 的启动	3
1.2.2 Visual Basic 的退出	5
1.3 主窗口	5
1.3.1 标题栏	5
1.3.2 菜单栏	5
1.3.3 工具栏	6
1.4 其它窗口	7
1.4.1 窗体设计器窗口与工具箱窗口	7
1.4.2 属性窗口与代码窗口	8
1.4.3 工程资源管理器窗口	8
习题	8
第2章 对象及操作	10
2.1 对象	10
2.1.1 Visual Basic 的对象	10
2.1.2 对象属性的设置	11
2.2 窗体	12
2.2.1 窗体的结构与属性	12
2.2.2 窗体的主要方法	16
2.2.3 窗体的事件	17
2.3 控件	18
2.3.1 标准控件	18
2.3.2 控件的命名和控件值	20
2.4 控件的画法和基本操作	22
2.4.1 控件绘制方法	22
2.4.2 控件的基本操作	23
2.5 事件驱动	24
2.5.1 过程驱动	24
2.5.2 事件驱动	25
2.5.3 Visual Basic 程序设计的一般过程	26
2.5.4 Visual Basic 应用程序的构成	29
2.5.5 程序的保存和装入	30

习题	31
第3章 数据类型及运算符	34
3.1 Visual Basic 标识符	34
3.2 基本数据类型	34
3.2.1 字符串(String)	35
3.2.2 数值(Numeric)	35
3.2.3 其他数据类型	36
3.3 常量和变量	37
3.3.1 常量	37
3.3.2 变量	38
3.3.3 局部变量与全局变量	40
3.3.4 缺省声明	42
3.4 用户定义的数据类型	43
3.4.1 记录类型	43
3.4.2 枚举类型	44
3.5 常用内部函数	46
3.5.1 数学函数	46
3.5.2 转换函数	46
3.5.3 字符处理与字符串函数	47
3.5.4 日期函数	51
3.5.5 随机函数	52
3.6 运算符与表达式	53
3.6.1 算术运算符	53
3.6.2 字符串连接运算符	54
3.6.3 关系运算符	55
3.6.4 逻辑运算符	55
3.6.5 表达式的执行顺序	56
习题	56
第4章 数据输入与输出	63
4.1 数据输出	63
4.1.1 Print 方法	63
4.1.2 与 Print 方法有关的函数(Tab、Spc、Space\$)	65
4.1.3 格式输出(Format\$)	68
4.1.4 其他方法和属性	70
4.2 InputBox\$函数	73
4.3 MsgBox 函数和 MsgBox 语句	74
4.3.1 MsgBox 函数	75
4.3.2 MsgBox 语句	77
4.4 字形	77
4.4.1 字体与大小	78

4.4.2 其他属性	80
4.5 打印机输出	81
4.5.1 直接输出	81
4.5.2 窗体输出	83
习题	84
第5章 标准控件	90
5.1 文本控件	90
5.1.1 标签(Label)	90
5.1.2 文本框(TextBox)	91
5.2 图形控件	94
5.2.1 图片框(PictureBox)、图像框(Image)控件	94
5.2.2 线段(Line)、形状(Shape)控件	97
5.3 按钮控件	100
5.3.1 属性和事件	100
5.3.2 应用举例	101
5.4 选择控件:复选框和单选按钮	103
5.4.1 属性与事件	103
5.4.2 应用举例	104
5.5 选择控件:列表框和组合框	106
5.5.1 列表框(ListBox)	106
5.5.2 组合框(ComboBox)	107
5.5.3 应用举例	108
5.6 滚动条	110
5.6.1 滚动条控件的常用属性和事件	111
5.6.2 应用举例	111
5.7 计时器	114
5.7.1 计时器控件的主要属性和事件	114
5.7.2 应用举例	115
5.8 框架	115
5.8.1 框架(Frame)及使用	115
5.8.2 应用举例	116
5.9 焦点与Tab顺序	117
5.9.1 设置焦点	117
5.9.2 Tab次序	118
习题	120
第6章 控制结构	125
6.1 算法、控制结构和语句	125
6.1.1 算法	125
6.1.2 三种基本控制结构	126
6.1.3 Visual Basic 基本语句	128

6.2 选择结构程序设计	130
6.2.1 单行结构条件语句	130
6.2.2 块结构条件语句	131
6.2.3 IIF 函数	133
6.2.4 应用举例	134
6.3 多分支结构程序设计	135
6.3.1 多分支结构	135
6.3.2 应用举例	138
6.4 For 循环结构程序设计	139
6.4.1 For 循环结构	139
6.4.2 应用举例	142
6.5 当循环控制结构	143
6.5.1 While...Wend 语句块	143
6.5.2 应用举例	145
6.6 Do 循环控制结构	146
6.6.1 Do 循环控制结构	146
6.6.2 应用举例	149
6.7 多重循环应用举例	150
6.8 GoTo 型控制	154
6.8.1 GoTo 语句	154
6.8.2 ON - GoTo 语句	155
习题	155
第7章 数组	162
7.1 数组的概念	162
7.1.1 数组的定义	162
7.1.2 LBound 和 UBound 函数	165
7.1.3 默认数组	165
7.2 静态数组与动态数组	166
7.2.1 动态数组	168
7.2.2 数组的清除	169
7.3 数组的基本操作	170
7.3.1 数组元素的输入、输出和复制	170
7.3.2 For Each...Next 语句	173
7.4 数组的初始化	174
7.5 控件数组	175
7.5.1 创建控件数组	175
7.5.2 控件数组中控件的引用	176
7.5.3 控件数组事件	177
习题	178

第8章 过程	187
8.1 Sub 过程	187
8.1.1 Sub 过程的建立	187
8.1.2 调用 Sub 过程	189
8.1.3 通用过程和事件过程	191
8.2 Function 过程	193
8.2.1 Function 过程的建立	193
8.2.2 调用 Function 过程	195
8.3 参数传递	197
8.3.1 形参与实参	197
8.3.2 引用	198
8.3.3 传值	200
8.3.4 数组参数的传递	201
8.3.5 传送记录	203
8.4 可选参数与可变参数	204
8.4.1 可选参数	204
8.4.2 可变参数	205
8.5 对象参数	206
8.5.1 窗体参数	206
8.5.2 控件参数	207
习题	209
第9章 菜单与对话框	219
9.1 用菜单编辑器建立菜单	219
9.1.1 菜单概述	219
9.1.2 菜单编辑器	220
9.1.3 用菜单编辑器建立菜单	222
9.2 菜单项的控制	226
9.2.1 有效性控制	227
9.2.2 菜单项标记	228
9.2.3 键盘选择	231
9.3 菜单项的增减	232
9.4 弹出式菜单	234
9.5 通用对话框	237
9.5.1 概述	237
9.5.2 自定义对话框	237
9.5.3 通用对话框	240
9.6 文件对话框	242
9.6.1 文件对话框的结构	242
9.6.2 文件对话框的属性	243
9.6.3 文件对话框举例	245

9.7 其他对话框	248
9.7.1 颜色对话框	248
9.7.2 字体对话框	249
9.7.3 打印对话框	252
习题.....	254
第10章 多重窗体与环境应用	260
10.1 建立多重窗体应用程序.....	260
10.1.1 窗体的创建、选择与移出	260
10.1.2 启动窗体的设置.....	261
10.1.3 窗体加载、显示、隐藏与卸载.....	261
10.1.4 应用举例.....	263
10.2 多重窗体程序的保存、装入与编译	267
10.3 Visual Basic 工程结构	267
10.3.1 标准模块.....	267
10.3.2 Sub Main 过程.....	269
10.3.3 窗体模块.....	269
10.4 闲置循环与 DoEvents 语句	269
10.4.1 DoEvents 的功能.....	269
10.4.2 闲置循环.....	271
习题.....	273
第11章 键盘和鼠标事件过程	275
11.1 KeyPress 事件	275
11.2 KeyDown 和 KeyUp 事件	276
11.3 鼠标事件.....	280
11.4 鼠标光标.....	283
11.5 拖放.....	284
11.5.1 属性.....	284
11.5.2 事件.....	285
11.5.3 方法.....	286
11.5.4 应用举例.....	286
习题.....	290
第12章 数据文件	294
12.1 文件的结构与分类.....	294
12.2 文件操作语句和函数.....	296
12.2.1 文件的打开与关闭.....	296
12.2.2 文件指针、Seek 语句和函数	298
12.2.3 其他语句和函数.....	299
12.3 顺序文件.....	300
12.3.1 顺序文件的写操作.....	300
12.3.2 顺序文件的读操作.....	304

12.4 随机文件.....	307
12.4.1 随机文件的打开与读写操作.....	308
12.4.2 随机文件中记录的添加与删除	312
12.5 文件系统控件.....	313
12.5.1 驱动器列表框和目录列表框.....	313
12.5.2 文件列表框.....	316
12.6 文件基本操作.....	318
习题.....	318
附录 I 综合上机练习	322
附录 II 笔试模拟试卷.....	344
附录 III 二级考试大纲(Visual Basic 语言程序设计)	366

第 1 章 Visual Basic 程序开发环境

Visual Basic 是一种可视化的、面向对象和采用事件驱动方式的结构化高级程序设计语言,可以开发 Windows 环境下的各类应用程序,已经成为风靡全球的快速软件开发工具之一。

1.1 Visual Basic 的版本和特点

Visual Basic 是在 BASIC 语言基础上研制而成的,Microsoft 公司对其进行了一次次的完善和不断地更新,它的功能越来越强大。

1.1.1 Visual Basic 的版本

Visual Basic 的版本时间表如下:

1991 年,Visual Basic 1.0 版。

1992 年,Visual Basic 2.0 版。

1993 年,Visual Basic 3.0 版。

1995 年,Visual Basic 4.0 版。

1997 年,Visual Basic 5.0 版。

1998 年,Visual Basic 6.0 版。

2002 年,Visual Basic. net 版。

从 1.0 版到 4.0 版的 Visual Basic 只有英文版,为 16 位版本。而从 5.0 以后,Visual Basic 在推出英文版本的同时,又推出了中文版,并升级为 32 位、全面支持面向对象的大型程序设计语言。6.0 版本在数据库访问、控件、语言、向导及 Internet 支持等方面增加了许多新功能。Visual Basic. net 是 Visual Basic 的最新版本。

本书介绍的是 Visual Basic 6.0,它是专门为 Microsoft 的 32 位操作系统设计的,分为学习版、专业版和企业版 3 个版本。这 3 个版本是在相同的基础上建立起来的,适用于不同的用户层次。

(1)学习版:Visual Basic 6.0 的基本版本,可用来开发 Windows 应用程序。该版本包括所有的内部控件(标准控件)、网格(Grid)控件、Tab 对象以及数据绑定控件等。

(2)专业版:该版本为专业编程人员提供一整套用于软件开发、功能完善的工具。它包括学习版的全部功能,同时包括 ActiveX 控件、Internet 控件、Crystal Report Writer 和报表控件等。

(3)企业版:该版本可供专业编程人员开发功能强大的分布式应用程序。它包括专业版的全部功能,同时具有自动化管理器、部件管理器、数据库管理器、Microsoft Visual SourceSafe 面向工程的控制系统等。

这 3 个版本中,企业版的功能最全,其次是专业版,再次是学习版。本书使用的是 Visual

Basic 6.0 企业版,但其内容也适合于专业版和学习版,所有程序可在专业版和学习版中运行。

1.1.2 Visual Basic 的特点

作为 Windows 环境下的快速集成开发工具,Visual Basic 主要有如下特点:

1. 可视化编程

传统的程序设计都是通过编写程序代码来设计用户界面的,在设计过程中看不到界面的实际显示效果,必须编译后运行程序时才能观察。如果对界面的效果不满意,还要回到程序中修改。有时会反复多次,大大影响了软件开发效率。Visual Basic 提供了可视化设计工具,开发人员不必编写大量程序代码,只需进行必要的屏幕布局设计。不仅在程序设计阶段就可及时观察到界面效果,而且能自动生成界面设计代码,减小了程序设计的难度,提高了程序设计的效率。

2. 面向对象的程序设计

Visual Basic 4.0 及以后的版本,均支持面向对象的程序设计。所谓对象,就是客观世界的实体。映射到程序世界,对象就是功能和数据的独立单元。对象之间只能通过它们认可的方式进行通信和调用,而有关对象功能及数据的具体描述是封装于对象内部的。Visual Basic 支持面向对象的程序设计思想,它提供的用于构造用户界面的可视化控件都可视为对象。每个对象都有一些属性(对应于数据)和方法(对应于程序代码),并且能够接收事件消息。Visual Basic 程序设计的相当一部分内容就是设置对象的属性值,调用对象的各种方法,编写对象对应的事件处理程序。

3. 结构化程序设计

所谓结构化程序设计,就是指以“自顶向下,逐步细化,模块化”思想作指导而进行的程序设计。从语言的角度来讲,就是对顺序、分支和循环三种基本结构的支持。Visual Basic 具有结构化程序设计的语句结构,简单易懂,很好地支持了结构化程序设计的思想。同时,提供了很友好的程序编码环境和功能强大的调试器和编译器。Visual Basic 既是一个解释型语言,又可以将设计的应用程序编译生成可执行文件(.EXE)。

4. 事件驱动编程机制

Visual Basic 采用了事件驱动的编程机制。一个对象可以感知或接收多个不同类型的事件,如单击鼠标、双击鼠标等。在 Visual Basic 编程中程序员要做的就是面向各对象的不同事件,编写事件过程代码。Visual Basic 整个应用程序的执行没有一预先的执行顺序,而是根据事件发生的顺序,分别触发对应的事件过程。

5. 数据库功能

Visual Basic 提供了强大的数据库管理、存取操作和数据检索功能。利用数据控件可以直接建立或处理 Microsoft Access 格式的数据库,也可直接编辑或访问 dBase、FoxPro、Paradox、Btrieve 等外部数据库。Visual Basic 提供了开放式数据连接(ODBC)、数据访问对象(DAO)、ActiveX 数据对象(ADO)等,可以方便地以 C/S 模式访问各种大型数据库系统,支持 SQL 数据标准。是一个很好的数据库应用系统前端开发工具。

1.2 Visual Basic 的启动与退出

Visual Basic 6.0 可以在 Windows 9x、Windows NT 4.0、Windows 2000、Windows XP 等操作系统下运行。它是 Visual Studio 6.0 套装软件的一个成员,可以随 Visual Studio 6.0 一起安装,也可以单独安装。这里假定 Visual Basic 6.0 是单独安装的。

1.2.1 Visual Basic 的启动

在 Windows 操作系统环境下,可以由多种方式启动 Visual Basic。

第一种方式:使用“开始”菜单中的“程序”命令。具体操作如下:

(1)在 Windows 环境下单击“开始”按钮,弹出“开始菜单”,把光标移动到“程序”命令项上,则弹出相应的程序级联菜单。

(2)在程序级联菜单中,把光标移动到“Microsoft Visual Basic 6.0 中文版”,则弹出 Microsoft Visual Basic 6.0 程序组的级联菜单。

(3)单击级联菜单中的“Microsoft Visual Basic 6.0 中文版”,即可启动 Visual Basic 6.0。

第二种方式:使用“我的电脑”或者“资源管理器”。具体操作如下:

(1)打开“我的电脑”或“资源管理器”窗口,找到安装 Visual Basic 6.0 的硬盘驱动器,双击此驱动器图标,弹出对应的窗口。

(2)找到安装 Visual Basic 6.0 的文件夹,并打开此文件。

(3)在安装 Visual Basic 6.0 的文件夹窗口中,找到“vb60. exe”文件,并双击此文件的图标,即可启动 Visual Basic 6.0。

第三种方式:使用“开始”菜单的“运行”命令。具体操作如下:

(1)在 Windows 环境下单击“开始”按钮,弹出“开始菜单”,单击此菜单中的“运行”命令,弹出“运行”对话框。

(2)单击对话框的“浏览”命令按钮,弹出“浏览”窗口。在此窗口中找到“vb60. exe”文件,并打开此文件。或直接输入 vb60. exe 的路径及文件名,如“C: \VB98\VB6. EXE”。

(3)单击“运行”对话框“确定”按钮,即可启动 Visual Basic 6.0。

第四种方式:使用“vb60. exe”快捷方式。在适当的位置(通常是桌面上)建立“vb60. exe”的快捷方式,单击此快捷方式,即可启动 Visual Basic 6.0。

不管用那种启动方式启动 Visual Basic,总是首先显示版权屏幕,随后会自动弹出如图 1.1 所示的“新建工程”对话框,其中有新建、现存、最新 3 个选项卡,通过它们可建立各种类型的 Visual Basic 工程(本书只涉及“标准 EXE 工程”)、打开已经建立过的工程、打开新近建立或打开过的工程。

如果你要新建一个“标准 EXE”工程,可选择“新建工程”对话框的“新建”选项卡,选中其中的“标准 EXE”图标后,单击“打开”按钮,即进入 Visual Basic 6.0 集成环境,如图 1.2 所示。此环境可分成若干部分,包括主窗口、窗体设计器窗口、工程资源管理器窗口、窗体布局窗口和属性窗口等,它们的大小、位置都是可以调整的。所以你的 Visual Basic 6.0 开发环境的屏幕界面与图 1.2 不一定完全相同。

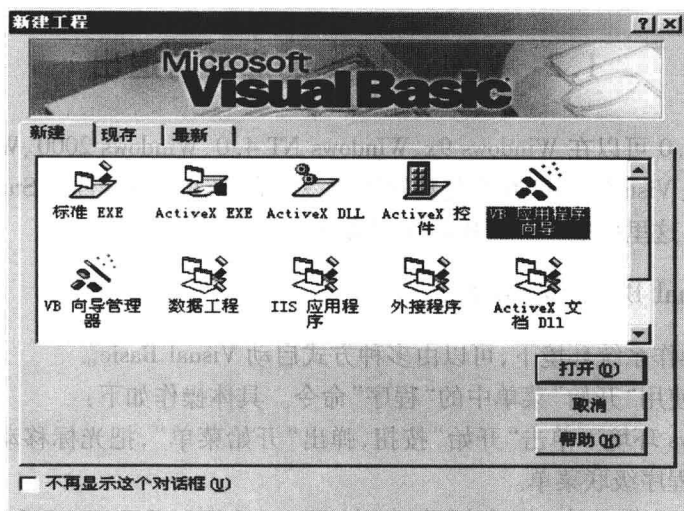


图 1.1 “新建工程”对话框

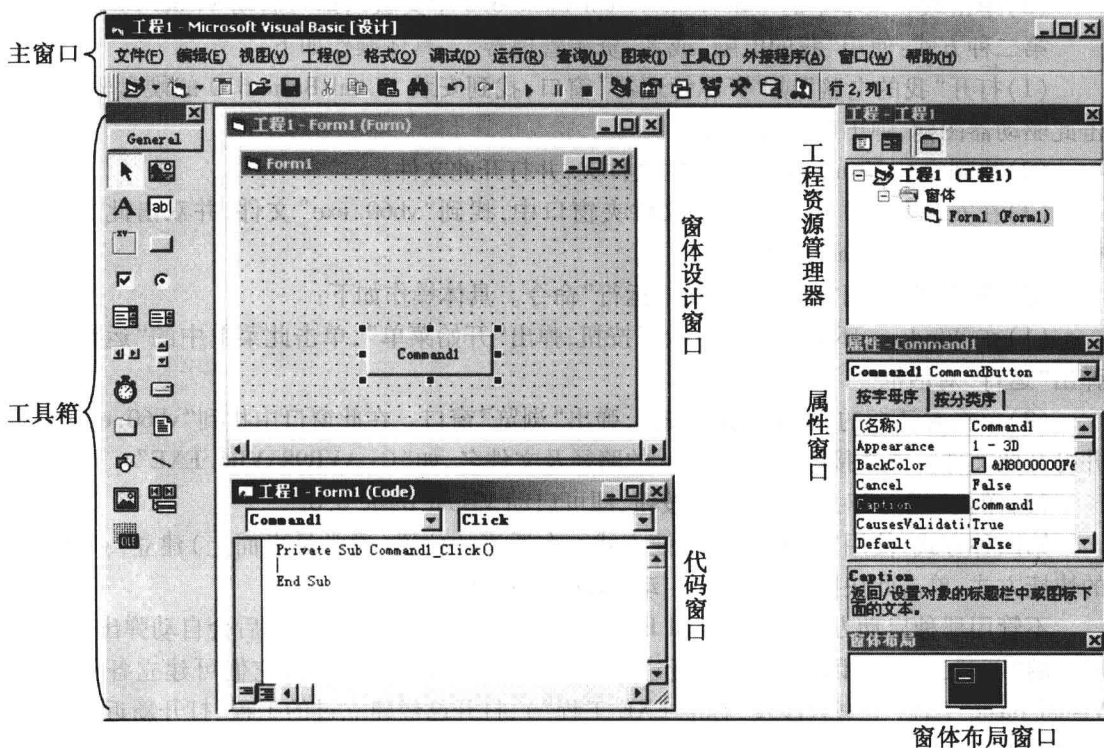


图 1.2 Visual Basic 6.0 开发环境的屏幕界面

在“新建工程”对话框的下端有一个“不再显示这个对话框”的复选框选项,启动 Visual Basic 6.0 时,如果选中了这种选项,以后再启动时则不再显示“新建工程”对话框,而直接进入开发环境。

另外,Visual Basic 6.0 开发环境分为 Windows MDI(多文档界面)和 SDI(单文档界面)两种界面。在默认情况下 Visual Basic 6.0 开发环境是传统的 Windows MDI 界面,可以通过

Visual Basic 6.0 开发环境的“工具”菜单中“选项”命令,打开“选项”对话框,选择“高级”选项卡,在其中选择(或者取消)“SDI 开发环境”选项,便可改变开发环境的界面方式,这种改变只有到重新启动 Visual Basic 时才能生效。

1.2.2 Visual Basic 的退出

退出 Visual Basic 6.0 的方式与退出其它应用程序类似,可以通过 Visual Basic 6.0 开发环境的“文件”菜单中的“退出”命令(或按 Alt + Q)退出,也可以通过关闭 Visual Basic 6.0 开发环境窗口的方式退出。

这里要注意的是,在退出 Visual Basic 6.0 前,一定要考虑是否对工程及相关内容进行“保存”,以防丢失自己的劳动成果。如果在你对工程的相关内容修改后,还没有存过盘,退出 Visual Basic 6.0 时会弹出是否存盘的提示对话框,此时可根据具体情况进行选择。

1.3 主 窗 口

主窗口位于 Visual Basic 6.0 开发环境的屏幕界面的顶部,又称为设计窗口。它包括标题栏、菜单栏和工具栏(如图 1.3 所示)。

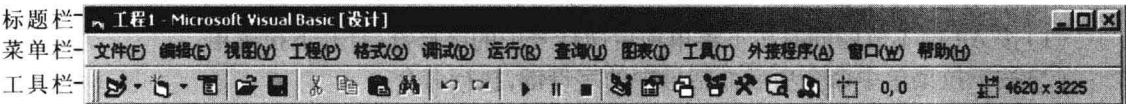


图 1.3 Visual Basic 6.0 开发环境的屏幕界面主窗口

1.3.1 标题栏

标题栏是 Visual Basic 6.0 开发环境屏幕界面顶部的水平条,从左到右包括系统菜单图标、应用程序标题、控制按钮(最大化、最小化、还原、关闭)。见图 1.3 所示。

Visual Basic 6.0 开发环境,有“设计”、“运行”和“中断”三种工作模式。当前工作模式将显示在应用程序标题后面的方括号中。

1.3.2 菜单栏

标题栏的下边是菜单栏。Visual Basic 6.0 的菜单栏有文件、编辑、视图、工程、格式、调试、运行、查询、图表、工具、外接程序、窗口和帮助等 13 个菜单,每个菜单对应一个下拉菜单,其中包括若干个命令项,如图 1.4 所示的是“文件”菜单对应的下拉菜单。Visual Basic 通过菜单中的命令完成编辑、编译、连接、运行、调试及文件的打开、保存等操作。

下拉菜单中的命令分为两类,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,如文件菜单中的“保存工程”命令项。另一类命令项后带有省略号,选择这类命令,总是要打开一个对话框,如文件菜单中的“打开工程”命令项。

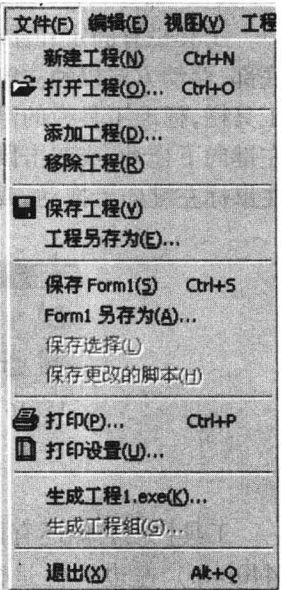


图 1.4 文件菜单

在很多命令项中,都有热键提示,如在文件菜单中有:

打开工程...	Ctrl+O
保存 FORM1	Ctrl+S

这里的“Ctrl + O”、“Ctrl + S”都是对应命令项的热键。在键盘上按下“Ctrl + O”则立即打开“打开工程”对话框。

Visual Basic 6.0 提供了多种打开菜单和选择菜单的方法:

第一种方法,用鼠标打开,并选择菜单(详略)。

第二种方法,具体如下:

(1)按 F10 或 Alt 键激活菜单栏。

(2)按菜单项后面括号中的字母键,打开下拉菜单。如按 F 键打开“文件”菜单。

(3)按下拉菜单中命令项后面括号中的字母键,执行命令。如按 O 键,执行“文件”菜单中的“打开工程”命令项。

第三种方法,具体如下:

(1)按下 F10 或 Alt 键激活菜单栏。

(2)按“←”或“→”键,选择菜单项,按回车打开下拉菜单。

(3)按“↑”、“↓”键,在下拉菜单中选择命令项,按回车执行命令。

第四种方法,具体如下:

(1)按 Alt 键不放的同时按下菜单项后面括号中的字母键,打开下拉菜单。

(2)按下拉菜单中命令项后面括号中的字母键,执行命令。

第五种方法,通过热键直接执行命令(详略)。

另外,如果打开了不适当或不需要的菜单,可按 Esc 键关闭。

1.3.3 工具栏

工具栏以图标形式提供部分常用菜单命令功能。Visual Basic 6.0 提供了标准、编辑、窗口编辑器和调试四个工具栏,用户可根据自己的需要自定义工具栏。可以通过“视图”菜单的“工具栏”命令,打开或关闭某个工具栏。

工具栏有固定和浮动两种显示方式,固定方式的菜单通常停靠于菜单栏的下方即主窗口底部,浮动方式以独立窗口的方式,浮动于主窗口的下方。图 1.3 中工具栏为固定方式的标准工具栏,标准工具栏的浮动方式如图 1.5 所示。在固定工具栏没有图标的任何位置按住鼠标左键向下拖动,或双击鼠标左键,可将固定工具栏变为浮动工具栏。在浮动工具栏的标题栏按住鼠标左键向上拖动,或双击鼠标左键,都可将浮动工具栏变为固定工具栏。

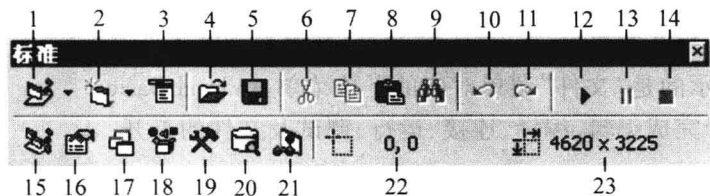


图 1.5 浮动式标准工具栏

工具栏的右侧还有两个栏,分别显示窗体的当前位置和大小,其单位为 twip(1 英寸 = 1440twip),左边一栏是窗体的位置(左上角的坐标),右边一栏是窗体的大小(长 × 宽)。