

致谢

本书的完成,得益于许多人的帮助,没有他们,单独完成这项工作,是很困难的,首先而且是最重要的,是要感谢 Osborne/McGraw-Hill,感谢他们的耐心、帮助和鼓励——这已经超出了他们的职责范围,感谢 Bill Pollock, Emily Rader, Cindy Clinon, Pat Mannion 和 Helen Worsley。

接下来,要感谢 Microsoft 的工作人员(很遗憾我不知道他们的名字),是他们创造了 Visual Basic 并升级到 3.0,这是一个伟大的产品,Dee Dee Walsh 负责 Visual Basic 的出版业务,她也竭尽所能满足各种奇怪的要求,并提供了各种帮助,Waggener-Edstrom 和 Tricia Bull 给了我许多我需要的信息。

Fred Mosher 和 David Schneider 仔细阅读了本书的原稿,并给了许多有益的建议以使本书更具特色,John Tralka 帮我检查了本书的上百个程序,Keith Pleas 出色地完成了本书的校对工作,非常感谢他为我所做的一切!

最后,感谢所有支持和鼓励我的朋友,尤其要感谢理解和支持我的朋友 Fran。

目 录

致谢	1
简介	1
0.1 关于本书	1
0.2 本书的组成	2
0.3 本书的约定	3
第一章 入门	4
1.1 什么是 Windows 和 Visual Basic	4
1.1.1 Visual Basic 应用程序的开发	5
1.2 安装 Visual Basic	7
1.2.1 运行安装程序	7
1.2.2 运行 Visual Basic	10
1.2.3 教学程序	11
第二章 Visual Basic 用户环境	14
2.1 主屏幕概述	14
2.1.1 标题条	14
2.1.2 菜单条	15
2.1.3 工具条	16
2.1.4 工具框	18
2.1.5 初始的窗体	18
2.1.6 工程文件窗口	18
2.2 在线帮助系统	19
2.2.1 帮助系统菜单	20
2.2.2 进入帮助系统	20
2.3 文件菜单的进一步介绍	24
2.3.1 编辑	27
2.3.2 “编辑”菜单	28
2.4 任选项 (Option) 菜单	29
2.5 装入并运行程序	30
第三章 定制窗体	32
3.1 启动一个新的工程文件	32
3.1.1 修改窗体	32
3.2 窗体属性	33
3.2.1 Properties 窗口	34
3.3 窗体的最一般属性	36

3.3.1	度量属性	39
3.3.2	Icons(图标)	40
3.4	颜色属性	40
3.4.1	通过调色板访问 BackColor 及 ForeColor	41
3.4.2	十六进制表示的颜色	42
3.5	让窗体响应事件	44
3.5.1	Code 窗口	44
3.6	打印窗体	49
3.6.1	打字错误	49
3.7	存储用户的工作	50
3.7.1	在 File 菜单中进行存储	51
3.7.2	只存储代码	52
3.8	建立独立的 Windows 程序	53
第四章	建立用户界面	55
4.1	工具框	55
4.2	生成控件	57
4.2.1	操作已存在的控件	58
4.2.2	生成控件的快捷键	59
4.2.3	操作多个控件	59
4.2.4	删除控件	60
4.3	命令按钮的属性	60
4.3.1	标题属性(Caption)	60
4.3.2	控件名字属性	62
4.3.3	命令按钮其他有用的属性	62
4.3.4	设置属性的快捷键方法	63
4.4	为命令按钮编写一些简单的事件过程	64
4.4.1	命令按钮的其他事件	65
4.4.2	关于命令按钮的最后几个问题	66
4.5	加速键(Access Keys)	67
4.6	图像控件	67
4.6.1	图像控件的属性	68
4.7	画线及形状控件	68
4.7.1	形状控件	68
4.7.2	画线控件	70
4.8	文字框和文字标注	70
4.8.1	文字框的标准属性	70
4.8.2	文字框的几个特殊属性	71
4.8.3	文字框的事件过程	72
4.8.4	文字标注(Lobels)	72

4.8.5	文字标注(Label)几个有用的属性	73
4.8.6	文字标注的事件过程	73
4.9	如何选中需要的控件	74
4.9.1	文字框的加速键	74
4.10	消息框	74
4.11	网格(Grid)	76
4.12	Visual Basic 应用程序的运行情况	77
4.13	窗体的 AutoRedrow 属性	78
4.13.1	Refresh 函数	79
4.14	窗体的 ASCII 码方式表示	79
第五章	编程初步	82
5.1	Visual Basic 程序剖析	82
5.2	代码(Code)窗口	82
5.3	Visual Basic 语句	83
5.3.1	注释语句(Remark)	84
5.3.2	End 语句	84
5.4	赋值语句和属性设置	84
5.4.1	为属性赋值	85
5.5	变量	86
5.5.1	变量类型	87
5.5.2	变量细则	88
5.5.3	类型的 Dim 语句	89
5.5.4	改变类型的缺省值	89
5.5.5	变量声明	90
5.5.6	变量作用域	91
5.6	过程中值的共享	91
5.6.1	保存变量	92
5.7	字符串	92
5.7.1	ASCII/ANSI 代码	93
5.7.2	固定长度的字符串	94
5.8	数值	95
5.8.1	数值操作	96
5.8.2	有关 Visual Basic 中数值的进一步了解	97
5.9	程序举例:抵押契据计算	98
5.9.1	抵押契约计算程序的改进	100
5.10	常量	101
5.10.1	提供的常量文件	102
5.11	多窗体的工程文件	103
5.11.1	为多窗体编写代码	104

5.11.2	如何在运行期处理多重窗体	105
5.11.3	使用多窗体的一例	106
5.11.4	在窗体(模式)中保存焦点	106
5.11.5	输入框	107
5.12	在窗体中打印	108
5.12.1	程序举例:在窗体中对中文字	109
5.12.2	代码中的字体属性	109
5.12.3	表	110
5.13	打印机	111
5.13.1	打印机的有用属性的方法	112
5.14	Format \$ 命令	112
5.15	日历信息	114
5.15.1	Dete \$ 函数	114
5.15.2	数值型日历函数	114
5.15.3	日期函数举例	115
5.15.4	其它日历函数	116
第六章	控制程序流程	118
6.1	重复操作	118
6.1.1	确定循环	118
6.1.2	非确定循环	124
6.2	做出决定	130
6.2.1	If - Then 模块	133
6.2.2	它是什么	134
6.2.3	一个例子:Dir \$ 命令	134
6.2.4	组合 If - Then 和循环	135
6.2.5	Select Case	136
6.2.6	结束 If - Then	137
6.2.7	一个例子:KeyPress	139
第七章	内部函数	140
7.1	字符串函数	140
7.1.1	分析字符串	140
7.1.2	一个例子:检查输入	146
7.1.3	Like 操作符和模糊搜索	147
7.1.4	其它的字符串函数	148
7.2	Rnd 函数	149
7.2.1	一个使用随机发生器的例子	152
7.3	位处理	154
7.4	比特层的逻辑操作符	155
7.4.1	KeyUp 和 KeyDown	157

7.5	金融函数	159
7.6	其余的金融函数	161
7.6.1	剩下的金融函数	162
7.7	数字函数	163
第八章	过程和错误陷阱	165
8.1	Function 过程	165
8.1.1	带多个参数的函数	167
8.1.2	函数的进一步说明	168
8.1.3	提前退出函数	169
8.1.4	字符串函数的例子	169
8.1.5	文字分析	172
8.1.6	应用举例:Pig Latin 生成器	173
8.2	子过程	176
8.2.1	一些简单但实用的过程	179
8.3	过程和函数的高级技术——传指针和传值	181
8.3.1	举例:洗牌	185
8.4	传递控件和窗体信息	186
8.4.1	对象变量	187
8.5	访问 Windows 函数	189
8.6	代码模块:全局过程和全局变量	191
8.6.1	添加或删除代码模块	193
8.6.2	过程作用域的讨论	193
8.7	DoEvents 函数	193
8.8	错误陷阱	195
8.9	程序设计方面的建议	197
第九章	数组、网格和记录	200
9.1	控件数组	200
9.1.1	增加和删除控件数组中的控件	202
9.1.2	一个例子:标号的一个方阵控件数组	204
9.1.3	控件数组和对象变量	207
9.2	列表:一维数组	207
9.2.1	固定和动态列表	207
9.2.2	使用列表的一些方法	209
9.2.3	一个例子:改善嵌入式的随机数字发生器	210
9.2.4	带有索引范围的列表	211
9.2.5	Erase 语句	212
9.3	多维数组	212
9.3.1	一个例子:魔方	213
9.4	过程中使用列表和数组	215

9.4.1	使用对象变量数组	216
9.5	网格控件	217
9.5.1	网格控件的一般属性	218
9.5.2	网格控件的事件和方法	220
9.6	排序和查找	221
9.6.1	查找	221
9.6.2	排序	223
9.7	记录	228
第十章	最终完成用户界面	231
10.1	工具框	231
10.1.1	框架控件	233
10.1.2	快速生成控件的另一种方法—— Sticky Click Method	234
10.1.3	图片框	234
10.1.4	图形与图片框	235
10.1.5	往工程文件中加入工具条和状态条	236
10.1.6	单选按钮	238
10.1.7	复选框	239
10.1.8	列表框和组合框	239
10.1.9	滚动条	245
10.1.10	定时器	247
10.2	菜单	250
10.2.1	菜单设计窗口(Menu Design Window)应用举例	253
10.2.2	弹出式(Pop-Up)菜单	253
10.2.3	运行程序时操作菜单的范例	254
10.3	常用对话框	256
10.3.1	使用常用对话框	257
10.4	MDI窗体	260
10.4.1	Windows 菜单的内部函数 Arrange	261
10.5	关于 Visual Basic 显示控件方法的进一步介绍:Z 顺序	262
10.6	关于窗口设计的一些建议	262
第十一章	测试、调试工具和技术	264
11.1	调试工具及其功能	264
11.2	测试程序	265
11.3	设计程序使测试简单	266
11.3.1	逻辑错误	267
11.4	Debug 窗口	267
11.4.1	在 Debug 窗口内进一步调试程序	268
11.4.2	单步执行	269
11.5	暂停程序	270

11.5.1	使用断点测试程序	271
11.5.2	设置 Next Statement(下一条语句)	274
11.6	程序调试综述	274
11.6.1	事件驱动错误和问题	275
11.7	文档和程序风格	275
第十二章	绘图操作	278
12.1	绘图基础	279
12.2	屏幕标度	281
12.2.1	示例:简单动画	282
12.2.2	定制标度	283
12.2.3	设置标度的另一种方法	284
12.3	色彩	285
12.4	像素控制	286
12.4.1	示例程序:“Visual Basic 素描”	288
12.5	线和框	291
12.5.1	参考终点	292
12.5.2	示例:X-Y 平面	293
12.5.3	网格图	294
12.5.4	DrawWidth 和 DrawStyle 属性	294
12.5.5	方框	296
12.5.6	示例:润饰一个控件	301
12.5.7	动画和 DRAWMODE 属性	302
12.6	圆、椭圆和饼图	304
12.6.1	示例:饼图	306
12.6.2	椭圆和宽高比(Aspect Ratio)	307
12.7	曲线	307
12.7.1	简单公式描述的图形	308
12.7.2	极坐标	311
12.7.3	极坐标演示例程	315
第十三章	监视鼠标活动	320
13.1	鼠标事件过程	320
13.1.1	MouseUp/MouseDown 事件	321
13.1.2	MouseMove 事件	323
13.2	拖放操作	325
13.2.1	手工拖动	326
13.2.2	DragOver 事件	327
13.2.3	示例:以拖/放方式删除文件	327
第十四章	文件操作	333
14.1	与 DOS 相关的文件命令	333

14.1.1	示例:如何重设已登录的驱动器	334
14.1.2	Shell 函数	334
14.1.3	命令行信息	335
14.1.4	文件操作函数	336
14.2	文件系统控件	337
14.2.1	文件列表框	337
14.2.2	目录列表框	339
14.2.3	驱动器列表框	340
14.2.4	协同所有文件控件	341
14.3	顺序文件	342
14.3.1	从文件中读取信息	344
14.3.2	给已有文件添加信息	347
14.3.3	通用顺序文件	348
14.3.4	送入特殊字符至文件	350
14.3.5	修改顺序文件内容	351
14.4	让文件程序更有力:错误处理	352
14.5	随机存取文件	354
14.5.1	随机存取文件的头信息和索引	356
14.6	二进制文件	357
14.6.1	谁使用的程序:App 对象	360
14.7	加密文件信息	363
14.7.1	更复杂的密码	364
14.7.2	一种更安全的密码	367
第十五章	与其他 Windows 应用程序通信	370
15.1	剪贴板	370
15.1.1	在 Visual Basic 中选择文字	371
15.1.2	剪贴板格式与图形传输	372
15.1.3	剪贴板使用示例程序	372
15.2	激活 Windows 应用程序	374
15.2.1	往应用程序发送击键事件	375
15.3	动态数据交换(DDE)	376
15.3.1	在设计程序时建立 DDE 链接	377
15.3.2	DDE 属性	379
15.3.3	DDE 事件	380
15.3.4	DDE 内部方法	381
15.3.5	通过代码建立 DDE 链接	382
15.4	OLE 2.0	386
15.4.1	使用 OLE	386
15.5	在设计程序时使用 OLE 2.0	387

15.5.1	Paste Special 对话框	388
15.5.2	OLE 属性	389
第十六章	递归	392
16.1	递归初步.....	393
16.1.1	递归函数.....	394
16.1.2	简单递归过程.....	397
16.1.3	一个示例——汉诺(Hanoi)塔	398
16.2	递归排序.....	402
16.2.1	合并排序.....	402
16.2.2	快速排序(Quicksort)	405
16.2.3	使排序更稳定.....	407
16.3	二叉树.....	407
16.3.1	树排序.....	412
16.4	分形(Fractals)	413
16.4.1	其它分形曲线.....	416
16.5	什么时候不用递归.....	420
第十七章	数据访问	423
17.1	现代数据库概述.....	423
17.2	数据管理器(Data Manager)	425
17.2.1	使用 Data Manager 检查现有数据库	425
17.2.2	使用 Data Manager 创建新的数据库	429
17.2.3	Data Manager 的另外一些操作	432
17.3	使用数据控件.....	432
17.3.1	查看工作中的数据控件.....	433
17.4	数据控件编程.....	435
17.4.1	数据控件的其它有用方法和事件.....	436
17.4.2	通过代码设置属性.....	439
17.5	监视数据库的变化.....	440
17.5.1	事务控件.....	441
17.6	结构化查询语言(SQL)基础.....	442
17.6.1	SELECT 语句的深入了解	442
17.6.2	通过 SQL 修改表结构	443
17.7	数据库对象.....	444
第十八章	支票簿管理程序	447
18.1	设计程序.....	447
18.2	用户接口.....	448
18.3	启动窗体和设置目录窗体.....	449
18.3.1	用于帐号统计窗体的图片框.....	452
18.3.2	保留 Finance 窗体代码	452

18.4	Accounts 窗体	453
18.5	Estats(Edit Statistics)窗体	456
18.6	用于输入支票和储蓄单的窗体.....	456
18.7	Reports 窗体	457
18.8	Results 窗体	459
18.9	程序的通用代码.....	459
18.9.1	通用过程模块.....	460
18.10	Finance 窗体.....	466
18.11	SetDir 窗体	469
18.12	Accounts 窗体	471
18.13	EStats 窗体	472
18.14	Check 窗体	475
18.15	Deposits 窗体	478
18.16	Reports 窗体.....	480
18.17	Results 窗体	487
附录 A	Visual Basic 专业版编辑器	488
A.1	Visual Basic 专业版编辑器概述.....	488
A.2	帮助编译器	489
A.2.1	编写 Help 标题	490
A.2.2	构造上下文相关的帮助	492
A.3	构造和编译 Help 文件	493
A.4	Setup 工具箱	494
A.4.1	如何使用 Setup 工具箱	494
A.4.2	改写 SETUP1.FRM	495
A.5	概述定制控件	498
A.5.1	动画按钮	499
A.5.2	通信	501
A.5.3	Crystal 报告控件.....	501
A.5.4	标准规格控件	501
A.5.5	图形	501
A.5.6	键状态	503
A.5.7	MAPI 控件	503
A.5.8	屏蔽编辑	504
A.5.9	多媒体的 MCI 设备控件	504
A.5.10	框架控件	504
A.5.11	画笔编辑控件	504
A.5.12	图片剪贴控件	504
A.5.13	旋转按钮	504
A.5.14	3-D 控件	505

A.6	API 工具	506
A.7	Visual Basic 在线知识基础	506
附录 B	Setup Wizard	508
B.1	使用 Wizard	509
附录 C	从 QuickBASIC 到 Visual Basic	512
C.1	一些普通信息	512
C.2	关键字的变化	513
附录 D	第三方工具和其他资源	517
D.1	工具	517
D.1.1	设计和调试工具	517
D.1.2	提供速度和便利的工具	517
D.1.3	定制控件	518
D.2	资源	519
附录 E	Microsoft 的二十个 Visual Basic 技术支持问题	520
E.1	问题:在使用 ODBC 和 Microsoft Visual Basic for Windows 连接数据库时出现了问题,错在何处	521
E.1.1	.INI 文件设置	521
E.1.2	ODBC 及驱动器 DLL	522
E.1.3	连接一数据源所需的服务器信息	523
E.1.4	Microsoft 及 Sybase SQL 服务器特有的信息	523
E.2	问题:在 Microsoft Windows for Workgroups 下怎样使用 Microsoft Visual Basic 来建立一个网络 DDE 链接	524
E.2.1	第一步:调用 NDdeShareAdd()函数加入 DDE 共享	524
E.2.2	第二步:建立 DDE 源应用程序	525
E.2.3	第三步:建立 DDE 的应用程序	528
E.3	问题:在 Microsoft Visual Basic 目的应用程序和 Visual Basic 源应用程序间初始化 DDE 需要些什么步骤	529
E.3.1	第一步:建立 Visual Basic 源应用程序	529
E.3.2	第二步:在 Visual Basic 中建立目的应用程序	529
E.3.3	第三步:试验程序运行	531
E.3.4	第四步:试验	531
E.4	问题:在 Microsoft Visual Basic 目的应用程序和 Microsoft Excel 源应用程序间初始化 DDE 需要些什么步骤	531
E.5	问题:在 Microsoft Visual Basic 目的应用程序和 Microsoft Word for Windows (WINWORD.EXE)源文档之间初始化 DDE 需要些什么步骤	534
E.5.1	第一步:在 Word for Windows 中建立源文档	534
E.5.2	第二步:在 Visual Basic 中建立目的应用程序	535
E.5.3	第三步:试验程序运行	536
E.5.4	第四步:试验	536

E. 6	问题:在 Microsoft Visual for Windows 中怎样向 Microsoft Windows Program Manager 发送 DDE 接口命令	537
E. 7	问题:在 Visual Basic for Windows 里怎样将全屏幕拷贝进图画框	538
E. 8	问题:怎样使用 Microsoft Visual Basic for Windows 将图画框中的 控件的位图送到打印机上	540
E. 9	问题:怎样在 Microsoft Visual for Windows 内打印控件或窗体	542
E. 10	问题:怎样在 Visual Basic 应用程序内把打印机 定向到横向模式或纵向模式	550
E. 11	问题:怎样从 Visual Basic 应用程序里访问 Windows 初始化文件的信息 ..	552
E. 12	问题:怎样用 Visual Basic for Windows 创建带透明背景的位图	554
E. 13	问题:怎样创建浮动窗——总在桌面上的其他窗口之上的窗口	558
E. 14	问题:在 Microsoft Visual Basic for Windows 内 怎样确定外壳程序是否结束	559
E. 15	问题:怎样利用 API 函数调用返回的 LPSTR 值 得到它所指向的字符串数据	560
E. 16	问题:怎样用 Microsoft Visual Basic for Windows 创建可滚动的视窗	561
E. 17	问题:怎样知道给定的应用程序正在 Windows 下运行	563
E. 18	问题:怎样使用 Visual Basic for Windows 的数据控件在 记录设置(recordset)中滚动	564
E. 19	问题:如果已经有 VC++ 和 VBWIN Pro, 还需要 Visual Control Pack(VCP)吗	566
E. 20	问题:如果有多个 .DLL 和 .VBX 文件,应该选用哪一个	567

简介

Visual Basic 1.0 版问世时,Microsoft 公司的总裁 Bill Gates 称它为“可怕的”,在《信息世界》杂志上,Steve Gibson 认为 Visual Basic 是“令人震惊的奇迹”,Stewart Alsop 在接受《纽约时报》的采访时说它是“九十年代的完美的编程环境”。

Visual Basic 的优点在哪呢?它能为用户做些什么呢?对此,Bill Gates 描述它为“使用 Basic 语言的非常容易但强有力的开发 Windows 应用程序的工具”,只有当读者了解到 Microsoft Windows 被上百万的用户使用,而且,在过去,开发 Windows 应用程序需要专门的 C 程序员他必须有 20 磅重的资料和至少 20 兆(很可能 30 兆)的硬盘空间以容纳 C 编译器时,才会觉得这句话的份量,正如 Charles Petzold(标准的用 C 编写 Windows 程序设计的作者在纽约时报上所说“对于那些试图让我们相信编写 Windows 应用程序是如何困难的人来说,Visual Basic 对他们的生存之道产生了威胁。”

Visual Basic 2.0 版比 1.0 版更快,更强大,更容易使用,Visual Basic 3.0 版增加了一些对可利用资源的控制,同时也增加了对公共的对话框的标准编译的支持,并且使它更方便地和其他 Windows 应用程序的通信。

欢迎使用九十年代的编程工具,Visual Basic 的最新版本。

0.1 关于本书

本书是一本容易理解的,对所有的 Visual Basic 3.0 的输入输出的详尽的手册,并且假定读者以前从未用 Visual Basic 编过程序(对于那些以前使用过 Quick BASIC 或其他结构化程序语言的读者来说,本书当然会更容易掌握),读者要从头开始学习,相信很快就能熟练掌握,不久读者就能充分利用 Visual Basic 的强有力的事件驱动特性去编写精彩的 Windows 应用程序,读完了这本书后,读者也就掌握了 Visual Basic。

我尽量采用新的方法来介绍 Visual Basic 编程,因此,本书即使对那些用传统编程语言的专家来说,也是有益的,我之所以这样做,是因为我认为把 Visual Basic 强迫纳入旧的编程语言去思考,就不能充分得到 Visual Basic 的强有力的支持。

为了使这本书更有益于读者,书中还包括了一些对其他大多数书中所遗漏的一些重要内容的讨论,书中用了一整章的篇幅介绍递归,书中还有对文件信息的加密办法和及时对用户的程序进行保护的讨论,书中还包括了对排序和查找算法的补充讨论,和许多的关键点和窍门,我的目的不仅仅是介绍 Visual Basic,而是使读者最终熟练的掌握它。

本书的最后是一个高级的支票管理程序。之所以选择这个程序,是因为我认为没有其他的途径能比这个程序更能显示 Visual Basic 的威力,这个程序要花不到一星期的时间设计并完成,可以想像,一些能产生上百万美元效益的金融管理软件,用 Visual Basic 来编写,仅需要几个星期,这已不仅仅是我个人的想像——Microsoft 的极其成功的会计软件“Profit”能够保持编写时间最短的记录的原因是因为它大部分由 Visual Basic 编写。

0.2 本书的组成

读者可根据自己的背景和需要使用这本书,熟悉结构化编程技术的读者可跳过由 Quick BASIC 继承来的诸如循环,子例程,函数等的全部讨论,这些在本书的第五至第九章,初学者应仔细地学习这些材料。

下面是本书各章的内容简介:

第一章和第二章告诉读者怎样安装 Visual Basic 以及帮助读者熟悉 Visual Basic 的环境。

第三章和第四章使读者迅速建立可定制窗口(称为窗体窗口)的概念,它是所有 Microsoft Windows 应用程序的基石,读者还可学会怎样添加基本的控件到窗体——如命令钮,文字框等等。

第五至第九章讨论了有关 Visual Basic 核心的编程技术,读者将学会怎样充分利用 Visual Basic 的变量及其内部函数,同时读者也将学会怎样添加自己的函数到 Visual Basic。另外,还将学习怎样排序和查找,网格控制,以及模块化的编程技术以使程序更灵活,更强大,更容易调试。

第十章告诉读者怎样完成用户界面,使读者学会怎样将剩下的控件添加到读者的窗体(form)中,其中包括添加列表框、单选按钮、复选框、滚动条和其他所有的 Microsoft Windows 用户希望在他的 Windows 应用程序中使用的控件,正是由于这些,使得 Windows 应用程序比 DOS 应用程序容易使用。

第十一章显示了 Visual Basic 3.0 版的强大的调试工具,读者将学会怎样隔离程序中的错误并最后改正它。

第十二章将把读者带入图形的世界,由于 Microsoft Windows 是一个基于图形的环境,Visual Basic 在这方面的能力也是强大的。

第十三章教读者怎样熟练地对鼠标编程。

第十四章教读者怎样在 Visual Basic 中对文件操作,包括对文件加密的高级技术(即保证文件内容的安全性)。

第十五章介绍动态数据交换(DDE)和对象链接与嵌入(DLE),DDE 就是在 Windows 应用软件中彼此交换数据,用户还能使 Visual Basic 配合进行传输,OLE 允许用户在 Visual Basic 中嵌入别的 Windows 应用程序,使用户能充分享用其他的 Windows 应用程序。

第十六章是关于递归的进一步说明,递归是强有力的编程工具,但常常被别的教材所忽略,本章包括排序算法以及有关递归图形的简介,这是在图形方面的一个很有用的工具,已被用在诸如 Star Trek II: The Wrath of Khan 的 Genesis 序列中。

第十七章介绍了数据管理以及 Visual Basic 3.0 版新添加的功能,读者将学会使用新的数据控制方法,用几行代码实现 PC 数据库的管理。

第十八章是关于支票管理,尽管程序很长,但当读者仔细阅读了前面的章节以后,已不会对它感到陌生。

附录 A 介绍了 Visual Basic 的专业版。重点介绍了使用 Help 编辑器建立完善的帮助系统,还介绍了安装(Setup)工具,它能使用户的程序与别人共享,同时,它还能帮用户决定是

否对自己的 Visual Basic 系统进行升级以使其能进行专业级的编辑。

附录 B 告诉读者怎样使用 Setup Wizard。它能使用户发行文件更容易。

附录 C 介绍了 Visual Basic 与 QuikBasic 之间的区别。

附录 D 介绍了第三方工具以及资源。

附录 E 包括 Microsoft 公司对 Visual Basic 的一些常见疑问的回答,增加这个附录的目的是为了给用户节省一些寻求帮助的时间。

0.3 本书的约定

键用大写字母表示,例如“CTRL”及“HOME”键箭头及其他的方向键用大写字母或拼写出来,例如,如果用户需要按鼠标的右键,本书表示为“按右键”。

如果用户要用组合键激活菜单条,这些键的头两个将用大写字母表示,中间用“+”号分隔,如,“按 CTRL+A,B”表示按下 Ctrl 键的同时,按下“A”键,然后按“B”键(此时 Ctrl 键仍被按下)。而 ALT+F+P 则表示先按 Alt 键,然后按“F”键,最后按“P”键,此时,Alt 键不需要被按下。

DOS 命令,文件名以及扩展名全用大写字母表示,如(COMMAND.COM, .TXT 等等, Microsoft Windows 大部分都参照 Windows,关键字的第一个字母大写如:Print, DeBug, FontSize 等等。

最后,程序之间用单空格隔开,如下所示:

```
Sub Form_Click()  
    Print "Hello world!"  
End Sub
```

在编程时,程序代码应在一行之内,但在本书中,由于排版的原因可能需要数行,这时需要加下划线,例如:

```
Print "This line is much too long to fit on one line  
but is outdented to indicate it should be entered on one line"
```

```
M$ = "This line is also too long to fit but it " + _  
"has an underscore to indicate it should be entered on one line"
```


第一章 入门

本章对 Visual Basic 作一简介,描述了运行 Visual Basic 所需要的环境,怎样安装,以及怎样使用内带的教程,尽管本书不需要读者是使用 Microsoft Windows 的专家,也不需要 Visual Basic 的基础,但读者应会熟练使用鼠标,会操作一个窗口,会使用文件管理器以及程序管理器。

要掌握 Microsoft Windows 最好的办法就是编写 Windows 应用程序,这正是 Visual Basic 所要做的。

注意: 本章介绍的是 Visual Basic 的标准版,附录 A 包含了 Visual Basic 专业版的有关信息。

1.1 什么是 Windows 和 Visual Basic

图形用户接口(GUI),已引起了微机工业的革命,“一幅图片值千言万语”已被证实,用图形界面面对的是充满了图标的桌面系统,从而取代了以前的 C:>提示符,这就给用户提供了一个很直观的感受,而且操纵计算机也变得很容易,图 1.1 展示了一个典型的 Windows 桌面系统。

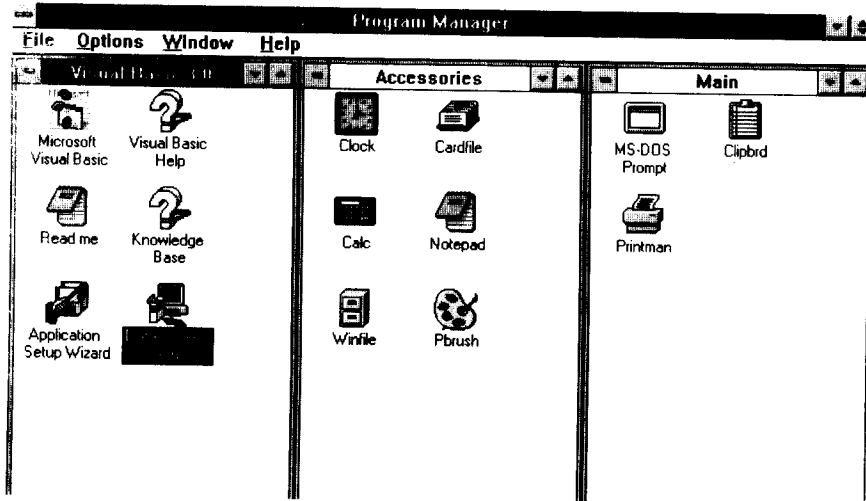


图 1.1 Windows 桌面系统

从长远的观点看,比 Windows 应用程序的外观更重要的是应用程序本身,Windows 应用程序通常有固定的用户界面,这就使得用户有更多的时间关心应用程序本身,而不用担心在菜单和对话框上花费更多的时间。

要做到这一点,是要付出代价的,在 Visual Basic 前,开发 Windows 应用程序要比开发 DOS 应用程序困难得多,程序员需要考虑的事太多,诸如鼠标在干什么,用户在哪儿加一菜