

最新流行软件轻松上手系列丛书

编著

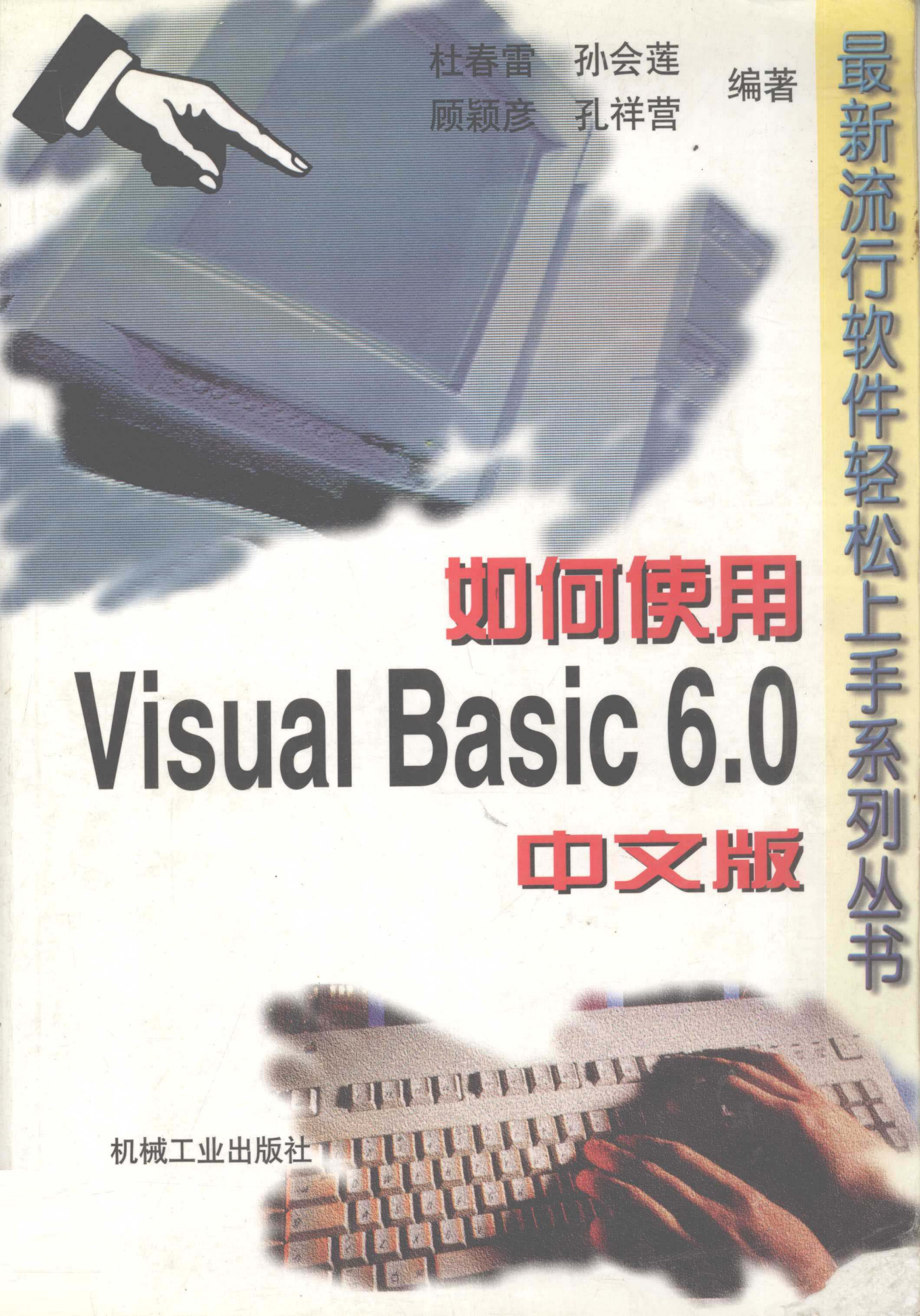
杜春雷 孙会莲
顾颖彦 孔祥营

如何使用

Visual Basic 6.0

中文版

机械工业出版社



最新流行软件轻松上手系列丛书

如何使用 Visual Basic 6.0 中文版

杜春雷 孙会莲
顾颖彦 孔祥营 编著



机械工业出版社

本书叙述了使用 Visual Basic 6.0 进行应用程序开发的各方面知识。全书分为三个部分，分别论述了 Visual Basic 6.0 程序设计基础、Visual Basic 6.0 中的 ActiveX 技术及使用 Visual Basic 6 开发数据处理应用程序。全书在编排上循序渐近、由浅入深，在内容上几乎覆盖了使用 Visual Basic 6.0 进行应用程序开发的所有细节，特别是关于 ActiveX 技术及数据访问技术部分提供了非常详细的参考资料。这些使得本书既适合于想全面学习 VB 的初学者，也可作为 VB 程序员进行应用程序开发的参考手册。

图书在版编目 (CIP) 数据

如何使用 Visual Basic 6.0 中文版/杜春雷等编著.—北京：机械工业出版社，1999.5

(最新流行软件轻松上手系列丛书)

ISBN 7-111-07301-0

I. 如… II. 杜… III. BASIC 语言-基本知识 IV. TP312

中国版本图书馆 CIP 数据核字 (1999) 第 20210 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑：张秀恩 王 龙 封面设计：姚 毅

责任印制：路 琳

北京市密云县印刷厂印刷·新华书店北京发行所发行

2000 年 1 月第 1 版第 2 次印刷

787mm×1092mm 1/16 30.5 印张·757 千字

5 001—8 000 册

定价：49.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

本书购书热线电话 (010) 68993821、68326677-2527

前 言

近几年来, Visual Basic 已经发展成为一种真正专业化的开发工具和编程环境。它不但功能强大,而且简单易用,它在数据库、多媒体、企业级管理系统的开发和一般应用程序的设计等方面,具有其它开发工具不可取代的优越性。

Visual Basic 6.0 在 Visual Basic 5.0 的基础上,又增加了许多新的功能。尤其在数据库访问和 Web 开发方面,提供了更加灵活方便的工具。如增加了 ADO 数据访问控件、数据环境设计器,对很多控件增加了数据绑定功能,使数据访问更加方便。

本书所有描述都针对 VB6 的企业版。涉及数据访问部分,所使用的数据库系统是 SQL Server 6.5。在 ActiveX 技术及数据访问技术部分提供了非常详细的参考资料,旨在为用户编写企业级的客户/服务器、及数据库应用程序提供方便快捷的指南。本书可以作为专业开发人员的参考资料。

全书在编排上循序渐进、由浅入深,在内容上几乎覆盖了使用 Visual Basic 6.0 进行应用程序开发的所有细节,也可以作为想全面学习 VB 的初学者的入门书。

全书由三个部分组成,分别论述了 Visual Basic 6.0 程序设计基础、Visual Basic 6.0 中的 ActiveX 技术及使用 Visual Basic 6.0 开发数据处理应用程序。

第一部分是基础篇,包括本书前 11 章的内容,主要介绍了 Visual Basic 编程的基础概念。通过这一部分的学习,读者就可以编一些 Visual Basic 的应用程序,并为后面更高级编程概念和技术的学习打下基础。下面就简单介绍一下该部分各章节的主要内容:

第 1 章向您简单介绍了 Visual Basic 的基本概念、安装方法及其集成开发环境,使读者对 Visual Basic 6.0 有一个初步的了解。

第 2 章介绍了 Visual Basic 中的工程结构、工程的基本操作以及如何编译生成可执行文件。

第 3 章介绍了一些窗体和控件的基本知识,使读者知道如何设计窗体和控件,具体介绍了 Visual Basic 标准控件的功能和用法,并通过一些小的例程加以说明。

第 4 章具体介绍了 Visual Basic 的代码编写机制、常量、变量、数据类型、过程、控制结构以及如何用对象编写 Visual Basic 应用程序。有一定编程经验的读者看过本章会发现 Visual Basic 的编程结构与其它编程语言类似,只是它的事件驱动的特性与其它语言有些不同。

第 5 章主要介绍了 Visual Basic 提供的调试工具和手段,通过本章读者可以学会如何调试自己的应用程序。

第 6、7、8 章主要介绍在应用程序中创建菜单、工具栏、对话框、多窗体与多文档等用户界面。

第 9 章是应用程序的输入输出处理,帮助读者掌握鼠标和键盘事件编程。

第 10 章简单介绍了 Visual Basic 6.0 中提供的新的文件操作功能——FSO 编程初步。

第 11 章介绍了 Visual Basic 应用程序使用 Windows 本身资源以及调用其它应用程序并与之交换信息的途径、方法。

第二部分是部件应用及设计技术，包括本书的第 12~14 章的内容。该部分的主要目标是使读者掌握使用 VB6.0 中的 ActiveX 技术，并用来开发自己的应用程序。内容包括 ActiveX 控件的使用，创建用户自己的 ActiveX 控件等。这一部分通过一些循序渐进的示例，向用户展示 VB6.0 提供的 ActiveX 控件。通过对这些控件的简单的引用，读者就可以创造出奇妙的具有 Windows 风格的应用程序。各章节主要内容如下：

第 12 章主要介绍了 ActiveX 技术的基本概念、发展历史以及 VB6.0 中增加的 ActiveX 的新技术。

第 13 章介绍了 ActiveX 控件的使用方法。

第 14 章主要介绍了用户如何创建自己的 ActiveX 控件。

第三部分是数据访问技术，包括本书的第 15~18 章的内容，本部分主要目标是使读者掌握使用 VB6.0 中的新的数据访问技术建立数据访问应用程序。它包含了 VB6.0 中数据访问对象的详细参考资料、一些新的工具与向导的介绍以及一些循序渐进的示例。通过本部分学习，读者可以建立起使用 VB6.0 开发数据访问应用程序的一般概念，并充分享受 VB6.0 中新的数据访问技术所带来的快捷、高效的开发方法。每部分都有若干示例，这些示例仅仅为了说明相关技术的使用方法，没有实际意义。有一些示例甚至不满足 VB 中代码的书写规范，这样做主要是考虑这些示例能够引导用户开始其数据访问程序设计。建议读者先浏览各数据访问对象的参考部分，再亲自实践各个示例，在实践示例的过程中加深对对象的理解。各章节主要内容如下：

第 15 章主要介绍 VB 中数据访问技术的总体结构及 VB6.0 中新添加的数据访问技术。从而使读者对 VB6.0 中数据访问技术有一个总的概念，便于以后的学习。

第 16 章介绍 VB 中建立数据连接的方法，主要介绍 VB6.0 中新增加的数据访问技术，ADO 数据访问技术与 VB5.0 中引进的远程数据访问技术 RDO 数据访问技术。

第 17 章介绍 VB6.0 中数据识别控件，主要介绍了 DataGrid、DataList、DataCombo、Hierarchical FlexGrid、Datapeater 等控件个编写过程中所做的一切，没有他的工作，就不可能有这本书。同时，还要感谢中国舰船研究院第七一六研究所计算机专家李向东、方军等。

第 18 章主要介绍了数据环境设计器、用户连接设计器、数据窗体向导及报表生成器。

在这里要特别感谢杨彬在这本书的整的悉心指导。马杰、徐和钦、陈永智、董渊、周世武、徐臻青、陈旭、裴民、汤洪涛、薛松、刘彬、李腾、潘擎宇、赵栋伟、李微、赵瑛、张新明、李岩、高鹏、王平、刘伟峰、孙方、周涛等同志在本书的编写过程中都做了很多有益的工作。最后，应该说明家家是这本书产生的最重要的原因。

由于时间仓促，加之本书涉及的内容较新，疏漏和错误之处在所难免。我们热诚期望广大读者多提宝贵意见。

编 者

1999 年 4 月

目 录

前 言

第一部分 VB6 程序设计基础	1
第 1 章 Visual Basic 6.0 概述	1
1.1 Visual Basic 简介	1
1.2 安装 Visual Basic 6.0	2
1.3 Visual Basic 6.0 集成开发环境	3
第 2 章 工程的管理	6
2.1 Visual Basic 的工程及其结构	6
2.1.1 工程概述	6
2.1.2 工程资源管理器	6
2.1.3 工程文件	7
2.1.4 工程的结构	7
2.2 工程的基本操作	8
2.2.1 创建、打开和保存工程	8
2.2.2 添加、删除和保存文件	10
2.2.3 在工程中操作控件和对象	11
2.3 制作可执行文件	13
2.4 设置工程选项	14
2.5 使用向导和外接程序	16
2.5.1 使用外接程序管理器	16
2.5.2 使用向导	17
第 3 章 窗体和控件	18
3.1 对象的属性、方法和事件	18
3.2 设计窗体	22
3.2.1 设置窗体的属性	22
3.2.2 窗体的事件和方法	24
3.3 标准控件	24
3.3.1 标准控件概述	24
3.3.2 命令按钮控件	25
3.3.3 文本控件	26
3.3.4 选择性控件	28
3.3.5 显示图片和图形的控件	35
3.3.6 数据控件	40
3.3.7 文件系统控件	41
3.3.8 其它控件	43

3.3.9	焦点概述	45
3.3.10	设置 Tab 键顺序	46
第 4 章	Visual Basic 编程基础	47
4.1	应用程序的结构	47
4.2	代码编写机制	48
4.2.1	代码模块	48
4.2.2	代码编辑器	48
4.2.3	自动完成编码	49
4.2.4	编写代码的基础	50
4.3	Visual Basic 中的变量	52
4.3.1	变量声明	52
4.3.2	变量的范围	54
4.3.3	使用多个同名的变量	55
4.4	Visual Basic 中的常量	57
4.5	数据类型	58
4.5.1	数字型变量	59
4.5.2	字符串型变量	59
4.5.3	字节型 (Byte) 变量	60
4.5.4	布尔型 (Boolean) 变量	60
4.5.5	日期型 (Date) 变量	61
4.5.6	对象型 (Object) 变量	61
4.5.7	变体型 (Variant) 变量	61
4.5.8	特定值	63
4.5.9	数据类型转换	64
4.5.10	数组	64
4.5.11	创建用户定义的数据类型	67
4.6	过程概述	68
4.6.1	子过程	68
4.6.2	函数过程	69
4.6.3	使用过程	69
4.6.4	向过程传递参数	72
4.7	程序流程控制	73
4.7.1	判定结构	74
4.7.2	循环结构	76
4.7.3	嵌套控制结构	78
4.7.4	退出控制结构	78
4.8	Visual Basic 中的对象	79
4.8.1	什么是对象	79
4.8.2	使用对象初步	80
4.8.3	对象之间的关系	81

4.8.4	创建对象.....	81
4.8.5	对象浏览器.....	85
第 5 章	调试应用程序代码.....	88
5.1	调试工具.....	88
5.2	设计时、运行时以及中断方式时.....	89
5.2.1	识别当前模式.....	89
5.2.2	用工具栏改变模式.....	89
5.3	调试方法.....	90
5.3.1	调试的几种简单方式.....	90
5.3.2	使用调试窗口.....	92
5.3.3	使用中断模式.....	93
5.3.4	用断点有选择地中断执行.....	95
5.3.5	用监视表达式监视数据.....	96
5.4	错误陷阱.....	99
5.5	错误种类.....	101
5.6	避免错误.....	102
第 6 章	菜单与工具栏设计.....	103
6.1	用菜单编辑器创建菜单.....	103
6.1.1	菜单基础.....	103
6.1.2	菜单编辑器.....	104
6.1.3	使用菜单编辑器中的列表框.....	104
6.1.4	分隔符条.....	105
6.1.5	定义访问键和快捷键.....	106
6.2	菜单标题命名准则.....	107
6.3	创建子菜单.....	108
6.4	创建菜单控件数组.....	108
6.5	运行时创建和修改菜单.....	109
6.5.1	菜单命令有效或无效.....	109
6.5.2	显示菜单控件上的复选标志.....	110
6.5.3	使菜单控件不可见.....	110
6.5.4	运行时添加菜单控件.....	111
6.6	编写菜单控件的代码.....	111
6.7	弹出式菜单.....	112
6.8	MDI 应用程序中的菜单.....	113
6.8.1	创建 MDI 应用程序的菜单.....	114
6.8.2	创建“窗口”菜单.....	114
6.8.3	排列子窗体.....	115
6.9	工具栏.....	116
6.9.1	创建工具栏.....	116
6.9.2	编写工具栏的代码.....	117

第7章 对话框设计	119
7.1 对话框的种类	119
7.2 预定义对话框的使用	120
7.3 用窗体作为自定义对话框	122
7.4 通用对话框的使用	125
7.4.1 通用对话框控件	125
7.4.2 “打开”和“另存为”对话框	127
7.4.3 “颜色”对话框	128
7.4.4 使用“字体”对话框	129
7.4.5 使用“打印”对话框	130
7.4.6 使用 ShowHelp 方法显示帮助文件	132
第8章 多窗体与多文档用户界面	134
8.1 界面样式	134
8.2 多文档界面 (MDI) 应用程序	135
8.3 设计时使用 MDI 子窗体	136
8.4 MDI 窗体运行时的特性	137
8.5 设计 MDI NotePad 应用程序	138
8.6 使用 MDI 窗体及其子窗体	140
8.6.1 指定活动子窗体或控件	140
8.6.2 加载 MDI 窗体及其子窗体	141
8.6.3 维护子窗体的状态信息	142
8.6.4 用 QueryUnload 卸载 MDI 窗体	143
8.7 再论窗体	144
第9章 鼠标与键盘事件	147
9.1 鼠标事件	147
9.1.1 MouseDown 事件	148
9.1.2 MouseMove 事件	149
9.1.3 MouseUp 事件	151
9.2 检测鼠标按钮	152
9.2.1 结合 MouseDown 和 MouseUp 事件使用 Button 参数	153
9.2.2 结合 MouseMove 使用 Button 参数	154
9.2.3 使用 Button 参数增强图形鼠标应用程序	156
9.3 检测 SHIFT、CTRL 及 ALT 键的状态	157
9.4 拖放和 OLE 拖放	158
9.4.1 拖放	158
9.4.2 OLE 拖放	159
9.5 自定义鼠标指针	161
9.6 键盘事件	162
9.6.1 编写低级键盘处理程序	163
9.6.2 KeyPress 事件	164

9.6.3	KeyDown 和 KeyUp 事件	165
9.6.4	编写窗体级键盘处理程序	166
第 10 章	File System Object 编程	169
10.1	File System Object 模型介绍	169
10.2	FSO 对象模型编程	170
10.2.1	创建 FileSystemObject 对象	170
10.2.2	使用适当的方法	171
10.2.3	FileSystemObject 对象的属性	171
10.3	驱动器和文件夹	172
10.3.1	Drive 对象	172
10.3.2	驱动器和目录操作	173
10.3.3	使用文件夹	175
10.4	文件操作	176
第 11 章	操作 Windows 环境	181
11.1	使用剪贴板	181
11.1.1	使用剪贴板剪切、复制和粘贴文本	181
11.1.2	在剪贴板上使用多种格式	182
11.1.3	检查剪贴板上的数据格式	184
11.2	激活其它 Windows 应用程序	185
11.2.1	使用 Shell 函数	185
11.2.2	使用 AppActivate 语句转移焦点	186
11.2.3	使用 SendKeys 发送键击	186
11.3	中断后台处理	189
11.3.1	允许用户中断任务	189
11.3.2	使用 DoEvents 分配处理器时间	190
第二部分	部件应用及设计技术	194
第 12 章	VB6 中 ActiveX 技术概述	194
12.1	COM 概述	194
12.2	ActiveX 与 OLE 比较	195
12.2.1	OLE 发展	195
12.2.2	ActiveX 与 OLE 的比较	195
12.3	VB6 中部件应用新特性	196
12.3.1	控件的新内容	196
12.3.2	增强的控件	197
12.3.3	控件的新特性	198
12.3.4	部件创建的新内容	198
第 13 章	ActiveX 部件应用	200
13.1	使用 ActiveX 控件	200
13.1.1	概述	200
13.1.2	添加和删除 ActiveX 控件	200

13.1.3	更新 VBX 控件.....	201
13.1.4	控件文件名.....	201
13.1.5	数据绑定 ActiveX 控件.....	202
13.1.6	ActiveX 控件类.....	203
13.1.7	发布可执行 (EXE) 文件.....	204
13.2	VB6 所提供的 ActiveX 控件.....	204
13.2.1	Animation 控件.....	204
13.2.2	Communications (M ^S Comm) 控件.....	207
13.2.3	CoolBar 控件.....	214
13.2.4	Datarepeater 控件.....	216
13.2.5	DateTimePicker 控件.....	225
13.2.6	FlatScrollBar 控件.....	228
13.2.7	ImageCombo 控件.....	230
13.2.8	ImageList 控件.....	234
13.2.9	Internet Transfer 控件.....	239
13.2.10	ListView 控件.....	246
13.2.11	MAPI 控件.....	247
13.2.12	MaskedEdit 控件.....	251
13.2.13	MonthView 控件.....	254
13.2.14	MSChart 控件.....	259
13.2.15	Multimedia MCI 控件.....	262
13.2.16	PictureClip 控件.....	266
13.2.17	ProgressBar 控件.....	269
13.2.18	RichTextBox 控件.....	271
13.2.19	Slider 控件.....	274
13.2.20	StatusBar 控件.....	277
13.2.21	SysInfo 控件.....	282
13.2.22	Tabbed Dialog 控件.....	286
13.2.23	TabStrip 控件.....	289
13.2.24	ToolBar 控件.....	293
13.2.25	TreeView 控件.....	298
13.2.26	UpDown 控件.....	301
13.2.27	Winsock 控件.....	304
第 14 章	创建 ActiveX 控件.....	312
14.1	概述.....	312
14.2	创建一个控件.....	313
14.2.1	创建 ActiveX 控件的一般过程.....	313
14.2.2	创建 ActiveX 控件实例.....	313
14.3	用户控件界面向导.....	318
第三部分	VB6 数据访问技术.....	321

第 15 章	VB6 中数据访问技术概述	321
15.1	VB6 中数据访问体系结构	321
15.2	VB6 中数据访问的新特性	322
15.2.1	新的数据访问对象	322
15.2.2	新的数据访问控件	322
15.2.3	新的开发工具	323
15.2.4	其他新的技术	324
第 16 章	VB 中数据访问技术选择	325
16.1	ADO 数据访问技术简介	325
16.2	ADO 对象模型	326
16.2.1	连接(Connection)对象	327
16.2.2	命令(Command)对象	332
16.2.3	结果集(Recordset)对象	335
16.2.4	参数(Parameter)对象	348
16.2.5	字段(Field)对象	350
16.2.6	错误(Error)对象	352
16.2.7	属性(Property)对象	352
16.3	ADO 对象编程基础	353
16.3.1	关于使用 ADO 对象编程的详细讨论	353
16.3.2	一个使用 ADO 对象编程的完整示例	358
16.4	使用 ADO 数据访问控件	372
16.4.1	ADO 控件介绍	372
16.4.2	使用 ADO 控件	375
16.4.3	可绑定到 ADO 上的数据识别控件	377
16.5	使用数据环境设计器创建 ADO 数据访问环境	377
16.5.1	使用数据环境设计器创建 Data Environment	378
16.5.2	一些使用数据环境设计器的完整示例	390
16.6	远程数据访问对象(RDO)模型	395
16.6.1	rdoEngine 对象	396
16.6.2	环境(rdoEnvironment)对象	399
16.6.3	连接(Connection)对象	402
16.6.4	查询(rdoQuery)对象	409
16.6.5	参数(rdoParameter)对象	412
16.6.6	表(rdoTable)对象	414
16.6.7	结果集(rdoResultset)对象	415
16.6.8	列(rdoColumn)对象	423
16.6.9	错误(rdoError)对象	427
16.7	使用用户连接设计器	428
16.7.1	引用相关的库	428
16.7.2	建立一个 UserConnection 对象	428

16.7.3	建立一个 Query 对象	430
16.7.4	在代码中使用 UserConnection 对象	430
16.8	使用远程数据访问控件	430
16.7.1	使用远程数据访问控件几点说明	431
16.7.2	一个使用远程数据访问控件的示例	431
第 17 章	VB 中数据识别控件介绍	434
17.1	使用 DataGrid 控件	434
17.1.1	设置 DataGrid 控件的主要属性	434
17.1.2	一个完整示例	435
17.1.3	DataGrid 控件的高级操作	437
17.2	使用 DataList 控件	443
17.2.1	DataList 控件的使用背景	443
17.2.2	设置 DataList 控件的主要属性	443
17.2.3	一个完整示例	444
17.3	使用 DataCombo 控件	446
17.3.1	DataCombo 控件的使用背景	446
17.3.2	DataCombo 控件的主要属性	446
17.3.3	一个完整的示例	447
17.4	使用 Hierarchical FlexGrid 控件	449
17.4.1	Hierarchical FlexGrid 控件的使用背景	449
17.4.2	使用 Hierarchical FlexGrid 控件	450
17.4.3	Hierarchical FlexGrid 控件的应用举例	452
17.5	使用 Datapeater 控件	454
17.5.1	Datapeater 控件的使用背景	454
17.5.2	Datapeater 控件的使用	454
17.5.3	Datapeater 控件的使用举例	457
第 18 章	VB 中数据访问开发工具介绍	461
18.1	使用数据报表设计器	461
18.1.1	数据报表设计器介绍	461
18.1.2	使用数据报表设计器建立数据报表的示例	463
18.1.3	打印一个数据报表	469
18.2	使用数据窗体向导	469
18.2.1	使用本地 Access 数据库文件的示例	470
18.2.2	使用远程数据库示例	471

第一部分 VB6 程序设计基础

第 1 章 Visual Basic 6.0 概述

1.1 Visual Basic 简介

Visual Basic 是在世界范围内广泛使用的一种强大的编程语言，它为 Windows 应用程序的开发提供了最迅速和便捷的方法。不论是 Windows 应用程序的专业开发人员还是初学者，Visual Basic 都为他们提供了一整套工具用来方便地开发应用程序。如果您现在想学习 Windows 编程，我们建议您先从 Visual Basic 学起，因为它简单易学，功能强大，可以很轻松地开发出 Windows 应用程序。目前 Visual Basic 软件开发团体是世界上最大的软件开发团体，它为全世界数千种工具和控件商所支持，因此选择 Visual Basic 将使您获得很大的优势。

Visual Basic 是在原有 Basic 语言的基础上进一步发展，至今包含了数百条语句、函数及关键词，其中很多和 Windows GUI 有直接关系。专业人员可以用 Visual Basic 实现其它任何 Windows 编程语言所能实现的功能，而初学者只要掌握几个关键词就可以建立实用的 Windows 应用程序。用 Visual Basic 既可以开发出一般的 Windows 应用程序，也可以实现先进的客户机/服务器、浏览器/服务器系统。目前 Visual Basic 6.0 有三种版本，即学习版、专业版和企业版，可以满足不同的开发需要。

为了理解 Visual Basic 应用程序的开发过程，首先要理解 Visual Basic 的一些关键概念。因为 Visual Basic 是基于 Windows 的开发语言，所以有必要与 Windows 环境保持一定的相似性。简单地说就是三个关键的概念：窗口、事件和消息。从这点上说 Visual Basic 6.0 与传统的 Windows 应用程序开发语言没有大的区别，但是由于处理各种窗口、事件和消息的所有可能的组合将有惊人的工作量，运用传统方法开发 Windows 应用程序将耗费程序员许多精力。幸运的是，Visual Basic 使您摆脱所有的低层消息处理。许多消息都由 Visual Basic 自动处理了，其它的作为事件过程由编程者自行处理。这样就可以快速创建强大的应用程序而无需涉及不必要的细节。

在传统的或“过程化”的应用程序中，应用程序自身控制了执行哪一部分代码及按何种顺序执行代码。从第一行代码执行程序并按应用程序中预定的路径执行，必要时调用过程。Visual Basic 是事件驱动的程序设计语言。在事件驱动的应用程序中，代码不是按照预定的路径执行，而是在响应不同的事件时执行不同的代码片段。事件可以由用户操作触发，也可以由来自操作系统或其它应用程序的消息触发，甚至由应用程序本身的消息触发。这些事件的顺序决定了代码执行的顺序，因此应用程序每次运行时所经过的代码的路径都是不同的。因为事件的顺序是无法预测的，所以在代码中必须对执行时的“各种状态”作一定的假设。当作出某些假设时，应该组织好应用程序的结构，以确保该假设始终有效。

传统的应用程序开发过程可以分为三个明显的步骤：编码、编译和测试代码。但是 Visual

Basic 与传统的编程语言不同, 它使用交互式方法开发应用程序, 使得三个步骤之间不再有明显的界限。在大多数语言里, 如果编写代码时发生了错误, 则在开始编译应用程序时该错误就会被编译器捕获, 此时必须查找并改正该错误, 然后再次进行编译, 对每一个发现的错误都要重复这样的过程。Visual Basic 在编程者输入代码时便进行解释, 即时捕获并突出显示大多数语法或拼写错误, 这看起来就像一位专家在实时监视代码的输入。除即时捕获错误外, Visual Basic 也在输入代码时部分地编译该代码。当准备运行和测试应用程序时, 只需极短时间即可完成编译。如果编译器发现了错误, 则将错误突出显示于代码中。这时可以更正错误并继续编译, 而不需从头开始。

由于 Visual Basic 的交互特性, 因此可以发现在开发应用程序时, 您自己正频繁地运行着您的应用程序。通过这种方式, 代码运行的效果可以在开发时进行测试, 而不必等到编译完成以后。

1.2 安装 Visual Basic 6.0

Visual Basic 6.0 必须使用安装程序安装后才能使用。在安装 Visual Basic 6.0 之前, 必须确认计算机能够满足最低的安装要求, 最好先阅读下安装盘根目录下的 Readme 文件。

下面我们将介绍 Visual Basic 企业版安装的系统要求及过程。运行 Visual Basic 6.0 企业版的系统配置要求是:

- (1) Microsoft Windows 95/98 或更高版本, 或 Microsoft Windows NT Workstation 4.0 (推荐 Service Pack 3) 或更高版本。
- (2) 486DX/66 MHz 或更高档的微处理器 (推荐使用 Pentium 或更高档的处理器), 或任何运行于 Microsoft Windows NT Workstation 的 Alpha 处理器。
- (3) 有一个 CD-ROM 驱动器。
- (4) Microsoft Windows 支持的 VGA 或分辨率更高的监视器。
- (5) 如果操作系统为 Windows 95/98 至少需要 16MB 内存, 如果操作系统为 Windows NT Workstation 4.0 则至少需要 32MB 内存
- (6) 最小安装至少需要 60MB 硬盘空间。
- (7) Microsoft 鼠标或者其它可以兼容的设备。

机器配置好之后, 最好先阅读一下 Readme 文件。Readme 文件列举了自 Visual Basic 文档出版发行以来的更改。可在最初的安装屏幕上选取 “Read Me First” 来读取该文件, 也可在 CD-ROM 的根目录下查找到这个文件。

运行安装程序时将会为 Visual Basic 创建目录, 从而可选择要安装的 Visual Basic 部件。如果要从 CD 盘上安装, 可以按照以下步骤执行:

- (1) 将安装光盘插入 CD-ROM 驱动器中。
- (2) 安装程序在 Disk 1 的根目录下, 可用操作系统中的适当命令来运行。如果您的计算机能够在系统中运行 AutoPlay, 则在插入 CD 盘时, 安装程序将被自动加载。
- (3) 选取 “安装 Visual Basic 6.0”。
- (4) 按照屏幕上的安装指令行事。

安装成功之前, 安装程序会让用户先重新启动计算机, 才能最后完成安装。一旦完成安装过程, 就可单击 Windows 任务栏中的开始按钮, 启动 Visual Basic。

如果想添加或删除 Visual Basic 部件，可以按照以下步骤进行：

- (1) 将 CD 盘插入 CD-ROM 驱动器中。
- (2) 安装程序在 CD 盘的根目录下，可用操作系统中的适当命令来运行。如果您的计算机能够在系统中运行 AutoPlay，则在插入 CD 盘时，安装程序将被自动加载。
- (3) 在“Microsoft Visual Basic 6.0 安装”对话框中选取“自定义”按钮。
- (4) 在“自定义”对话框的“选项”列表框中选定要安装的部件（或撤消选定要删除的部件）。
- (5) 按照屏幕上的安装指令行事。

1.3 Visual Basic 6.0 集成开发环境

Visual Basic 6.0 的工作环境常常是指集成开发环境（IDE），这是因为它在一个公共环境里集成了许多不同的功能，例如设计、编辑、编译和调试。这使用户在一个界面中可以完成多种任务。而在大多传统开发工具中，每个功能都是以一个独立的程序运行，并都有自己的界面。本节将简单介绍 Visual Basic 的用户集成开发环境，使大家对 Visual Basic 的开发环境有个大概的了解。

当运行 Visual Basic 安装程序时，允许将程序项置于已存在的程序组中，或在 Windows 中为 Visual Basic 创建一个新的程序组和程序项。这时可准备从 Windows 启动 Visual Basic。首先让我们启动 Visual Basic 6.0 的工作环境，点击 Windows 桌面的开始菜单，选择“程序”，接着选取“Microsoft Visual Basic 6.0”程序组，会弹出一个子菜单，然后用户在该子菜单选择 Visual Basic 6.0 项，即可启动 Visual Basic 6.0。

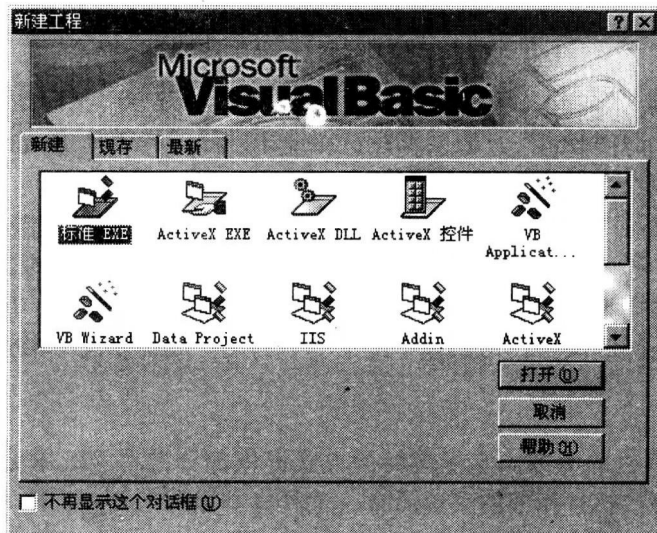


图 1-1 “新建工程”对话框

启动 Visual Basic 6.0 后，我们会看见如图 1-1 的对话框。我们现在暂时不管对话框中的选项，按缺省情况点击“打开”按钮，就会出现如图 1-2 所示的集成开发环境。

从图 1-2 中我们可以看到，进入 Visual Basic 集成开发环境后，屏幕中会出现：菜单栏、工具条、工具箱、工程资源管理器窗口、属性窗口和窗体布局窗口。

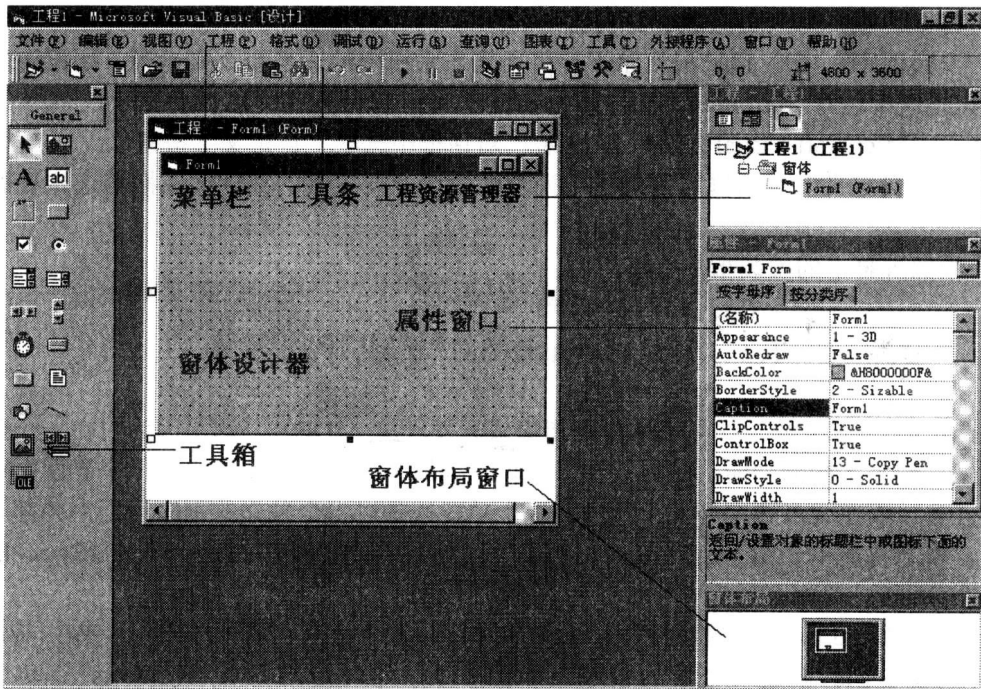


图 1-2 Visual Basic 6.0 集成开发环境

Visual Basic 集成开发环境 (IDE) 主要由以下元素组成 (注意下面有一些是图 1-2 中没有显示的) :

1. 菜单栏

进入 Visual Basic 集成开发环境后, 主界面的标题栏中显示“工程 1 — Microsoft Visual Basic [设计] — Form1[Form]”, 表明现在正在设计工程 1 的表单 Form1。随着工作状态的变化, 标题栏的内容也会发生变化。

标题栏的下面就是 Visual Basic 集成开发环境的菜单栏, 它除了提供“文件”、“编辑”、“视图”、“窗口”和“帮助”这些标准菜单外, 还提供了编程专用的功能菜单, 如: “工程”、“格式”、“调试”、“运行”、“工具”、“外接程序”等。每个菜单项有若干个菜单命令, 可执行不同的操作。用鼠标单击菜单中的某一项就能执行相应的菜单命令。

2. 工具栏

工具栏位于菜单栏的下方, 它以图标的方式提供部分常用的菜单命令。单击某个命令的图标按钮, 就能直接执行相应的菜单命令。工具栏有标准工具栏、窗体设计工具栏、编辑工具栏和调试工具栏。按照缺省规定, 启动 Visual Basic 之后显示标准工具栏。附加的编辑、窗体设计和调试的工具栏可以使用“视图”菜单上的“工具栏”命令移进或移出。在工具栏上单击鼠标右键, 可在弹出菜单中对工具栏上的图标根据个人喜好进行定制。

在工具栏的右方还有两个栏, 分别用来显示当前选中的控件的大小的位置, 其单位为 twip, 1 in (英寸) 等于 1400 twip。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, 画一条 1400twip 的直线, 打印出来都是 1 in。