



Visual Basic 6.0

应用与提高

计算机
实用
教程



◆ 东岳创作室 主编



人民邮电出版社

计 算 机 实 用 教 程

Visual Basic 6.0 应用与提高

东岳创作室 主编

人 民 邮 电 出 版 社

内 容 提 要

Visual Basic 6.0 是美国 Microsoft 软件公司推出的 Visual Basic 编程语言的最新版本, 其简单易用的程序设计环境、强大的功能、丰富的 ActiveX 控件和向导、极其友好的图形用户界面、简单的数据存取方式、良好的跨平台特性以及真正的可编译性, 使其成为目前广大开发人员十分喜爱的编程语言之一。

本书全面介绍了 Visual Basic 6.0 的操作环境和程序设计方法, 其内容涉及 Visual Basic 操作环境的使用, 程序界面设计, 基本控件用法, Visual Basic 程序设计基本概念, 鼠标和键盘事件的响应, 文本和图形的处理, 数据库访问, ActiveX 控件和 Internet, 程序发布等, 同时给出了多个程序设计实例, 以加深读者对 Visual Basic 程序设计基本概念和方法的理解。

本书内容丰富、条理清晰, 既可供 Visual Basic 初学者阅读, 也可供已有一定基础的用户提高之用。

计算机实用教程

Visual Basic 6.0 应用与提高

◆ 主 编 东岳创作室

责任编辑 须春美

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

北京朝阳隆昌印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 30.25

字数: 749 千字

1999 年 5 月第 1 版

印数: 5 001—9 000 册

2000 年 2 月北京第 2 次印刷

ISBN 7-115-07569-7/TP·953

定价: 41.00 元

编 者 的 话

和以前 VB 版本不同的是, Visual Basic 6.0、Visual FoxPro 6.0、Visual C++ 6.0 等各种编程语言的帮助均被集成到了 MSDN 库中。用户在使用时可单独安装该帮助库,并可定制所要安装的组件。此外, Visual Basic 6.0 还在以下几个方面进行了改进:

(1) 数据访问的改进。增加了 ADO (ActiveX Data Objects) 数据对象 (在所有版本中)、提供了数据环境设计器 (在专业版和企业版中) 等。

(2) 新的 Internet 特性。用户现在可以编写 Internet 服务器程序, 以使 Visual Basic 代码能够响应来自浏览器的用户请求。利用动态 HTML (DHTML) 技术, Web 页面上的任何元素都可具有属性、方法和事件。在 Visual Basic 6.0 中, 允许用户编写代码来响应 HTML 页上的动作, 而不用传送过程给服务器。

(3) Visual Basic 新增了一些控件并对某些控件进行了改进。在 Visual Basic 6.0 中, 新增了 ADO Data 控件 (所有版本)、CoolBar 控件 (专业版和企业版)、DataGrid 控件 (所有版本)、DataList 控件和 DataCombo 控件 (所有版本)、DataRepeater 控件 (专业版和企业版)、DateTimePicker 控件 (专业版和企业版)、Flat Scrollbar 控件 (专业版和企业版)、Hierarchical FlexGrid 控件 (所有版本)、ImageCombo 控件 (所有版本) 和 MonthView 控件 (专业版和企业版) 等。此外, 在 Visual Basic 6.0 中, Intrinsic、ImageList、ListView、MSChart、ProgressBar、Toolbar 等控件被进行了改进或增强。

(4) 在 Visual Basic 6.0 中, 用户可创建自己的 Data Consumer、数据源、工程开始选择、简化的 UserControl 等。

(5) Visual Basic 6.0 还对某些语言特性进行了改进, 例如, 用户定义类型的参数化、返回公共属性和方法的类型、函数可返回数组、可变数组可出现在等式的左侧、通过指定包含属性或方法名的字符串访问属性和方法、CreateObject 函数现在允许用户指定机器名作为可选参数以便访问远程机器上的对象等。

此外, Visual Basic 6.0 还新增了一些字符串函数, 如 Filter、FormatCurrency、FormatDateTime、FormatNumber、FormatPercent、InstrRev、Join、MonthName、Replace、Round、Split、StrReverse、WeekdayName 等。

本书全面介绍了 Visual Basic 6.0 的操作环境和程序设计方法, 内容涉及 Visual Basic 操作环境的使用, 程序界面设计, 基本控件用法, Visual Basic 程序设计基本概念, 鼠标和键盘事件的响应, 文本和图形的处理, 数据库访问, ActiveX 控件和 Internet, 程序发布等有关 Visual Basic 程序开发的所有方面。全书由东岳创作室策划, 参与本书编写工作的主要有何春垣、赵文生、刘先枝、郑永红、徐革、李冬、郭明文、刘珊、张春华、刘贞等, 本书由张海之负责审校。

由于时间仓促, 不当之处在所难免, 尚希读者批评指正。

编 者

1999 年 4 月

目 录

第一章 认识 Visual Basic 6.0.....	1
1.1 Visual Basic 简介.....	1
1.1.1 Visual Basic 语言的特点	1
1.1.2 Visual Basic 的版本	2
1.2 Visual Basic 程序设计基本概念.....	2
1.2.1 事件驱动编程	2
1.2.2 面向对象程序设计	3
1.2.3 Visual Basic 中的窗体、控件和工程	4
1.2.4 属性、方法和事件	5
1.2.5 交互式开发	6
1.3 Visual Basic 集成开发环境使用详解.....	7
1.3.1 启动 Visual Basic 集成开发环境	7
1.3.2 集成开发环境的组成元素	8
1.3.3 集成开发环境设置	10
1.4 工程管理详解.....	15
1.4.1 工程文件概述	15
1.4.2 Visual Basic 工程组成文件	16
1.4.3 创建、打开和保存工程	22
1.4.4 添加、删除和保存文件	23
1.4.5 在工程中添加控件	24
1.4.6 设置工程选项	26
1.4.7 制作和运行可执行文件	27
1.4.8 使用向导、模板和外接程序	31
1.4.9 利用多个工程进行工作	34
1.5 Visual Basic 6.0 的新特点.....	36
1.5.1 数据访问的改进	36
1.5.2 新的 Internet 特性.....	37
1.5.3 新的和改进的控件	38
1.5.4 组件创建增强	39
1.5.5 语言增强	39
1.5.6 向导增强	39
1.5.7 升级应用程序	40

第二章 创建用户界面.....	41
2.1 设计程序界面应该遵循的原则.....	41
2.1.1 界面设计基础	42
2.1.2 请用户参与界面设计	45
2.1.3 系统出错时与用户交互	46
2.1.4 设计用户辅助模式	48
2.1.5 各种显示类型的设计	49
2.2 设计窗体.....	51
2.2.1 设置窗体属性	51
2.2.2 窗体事件和方法	54
2.2.3 关于 Form 对象和 Forms 集合	55
2.2.4 设置启动窗体	56
2.2.5 将 Main 过程作为应用程序入口	56
2.2.6 设计快速显示窗体	57
2.2.7 结束应用程序	57
2.3 设计多文档界面.....	58
2.3.1 多文档界面概述	58
2.3.2 创建 MDI 应用程序.....	60
2.3.3 MDI 窗体运行特性.....	61
2.3.4 MDI 应用程序示例.....	62
2.3.5 使用 MDI 窗体及其子窗体.....	64
2.4 在应用程序中使用菜单.....	66
2.4.1 菜单概述	66
2.4.2 使用菜单编辑器	67
2.4.3 菜单标题与命名准则	71
2.4.4 创建子菜单	72
2.4.5 创建菜单控件数组	72
2.4.6 运行时创建和修改菜单	73
2.4.7 编写菜单控件的代码	75
2.4.8 设计弹出式菜单	75
2.4.9 MDI 应用程序中的菜单.....	77
2.5 设计工具栏.....	80
2.5.1 创建工具栏	80
2.5.2 协调菜单与工具栏的外观	82
2.6 使用对话框.....	83
2.6.1 模式与无模式对话框	83
2.6.2 预定义对话框的使用	84
2.6.3 用窗体作为自定义对话框	85

2.7	为应用程序设计帮助系统.....	88
2.7.1	增加帮助支持	88
2.7.2	使用“这是什么”帮助	89
2.7.3	随应用程序一起发布帮助文件	90
第三章	基本控件的用法.....	91
3.1	控件概述.....	91
3.1.1	控件分类	91
3.1.2	内部控件功能简述	92
3.1.3	标准 ActiveX 控件功能简述	93
3.1.4	控件命名约定	95
3.1.5	使用控件值	95
3.1.6	关于 Tab 键次序.....	96
3.1.7	关于焦点	97
3.1.8	通过约束焦点确认控件数据	97
3.2	使用控件数组.....	97
3.2.1	为何使用控件数组	97
3.2.2	设计时创建控件数组	99
3.2.3	运行时添加控件数组	99
3.2.4	在控件数组中添加和删除控件示例	100
3.3	调整控件布局.....	102
3.3.1	统一控件尺寸	102
3.3.2	调整水平和垂直间距	103
3.3.3	使控件在窗体中水平或垂直居中	103
3.3.4	对齐控件位置	103
3.3.5	锁定控件布局	104
3.4	使用命令按钮执行操作.....	104
3.4.1	使用 CommandButton.....	104
3.4.2	命令按钮与图像按钮比较	107
3.4.3	命令按钮应用程序示例	107
3.5	用于显示和输入文本的控件.....	109
3.5.1	用 Label 标签控件显示文本	109
3.5.2	标签控件应用示例	111
3.5.3	使用 TextBox 文本框输入文本.....	116
3.6	为用户提供选择的控件.....	119
3.6.1	用 CheckBox 复选框选择单个选项	119
3.6.2	用 OptionButton 选项按钮组进行分组选项	122
3.6.3	使用 ListBox 列表框从列表中进行选择	125
3.6.4	使用 ComboBox 组合框进行列表选择	134

3.6.5	使用 HScrollBar 和 VScrollBar 滚动条作为输入设备	138
3.7	显示图片和图形的控件	142
3.7.1	使用 PictureBox 图像框控件	142
3.7.2	使用 Image 图像控件	145
3.7.3	使用 Shape 形状控件	146
3.7.4	利用 Line 线形控件画辅助线	147
3.7.5	Image 应用程序示例	148
3.7.6	动画设计示例	150
3.8	使用 Frame 框架控件	163
3.8.1	在框架中绘制控件	164
3.8.2	在框架中选定多个控件	164
3.9	使用定时器控件 Timer	164
3.9.1	在窗体上放置 Timer 控件	164
3.9.2	初始化 Timer 控件	165
3.9.3	Timer 事件	166
3.9.4	Timer 控件应用示例	166
3.10	使用文件系统控件	168
3.10.1	控件概述	168
3.10.2	检查文件系统	169
3.10.3	使用驱动器列表框控件 DriveListBox	169
3.10.4	使用目录列表框控件 DirListBox	170
3.10.5	使用文件列表框控件 FileListBox	171
3.10.6	使用文件系统控件的组合	172
3.10.7	设计文件搜索器	173
3.11	使用 OLE 容器控件	181
3.11.1	OLE 容器控件的作用	181
3.11.2	OLE 容器控件应用示例	181
3.12	使用数据控件访问数据库	183
3.13	使用 CommonDialog 控件	185
3.13.1	CommonDialog 控件的功能	185
3.13.2	CommonDialog 控件应用示例	193
3.13.3	设计记事本程序	194
第四章	Visual Basic 程序设计基础	219
4.1	Visual Basic 应用程序的结构	219
4.1.1	Visual Basic 应用程序组成	219
4.1.2	事件驱动应用程序的工作方式	220
4.2	Visual Basic 的代码编写机制	220
4.2.1	编码前的准备工作	220

4.2.2	模块的三种类型	222
4.2.3	使用代码编辑器	223
4.2.4	自动完成编码	224
4.2.5	程序续行、合并、注释及其他	225
4.3	声明和使用变量、常数和数据类型	227
4.3.1	变量的声明方法	228
4.3.2	变量的作用范围	230
4.3.3	高级变量主题	231
4.3.4	常数的声明和应用	234
4.3.5	Visual Basic 中的数据类型	236
4.3.6	关于变量的进一步讨论	241
4.3.7	数组的声明和应用	245
4.3.8	使用动态数组	247
4.4	过程的声明和应用	248
4.4.1	Sub 过程	249
4.4.2	Function 过程	250
4.4.3	调用 Sub 过程和 Function 函数的方法	251
4.4.4	向过程传递参数	253
4.5	Visual Basic 程序控制结构	256
4.5.1	判定结构	257
4.5.2	循环结构	260
4.5.3	控制结构应用	262
4.6	在程序中使用对象	264
4.6.1	对象是什么	265
4.6.2	用对象能做什么	266
4.6.3	使用对象初步	267
4.6.4	对象之间的关系	268
4.6.5	创建对象	272
第五章	响应鼠标和键盘事件	278
5.1	响应鼠标事件	278
5.1.1	MouseDown 事件	279
5.1.2	MouseMove 事件	280
5.1.3	MouseUp 事件	281
5.2	检测鼠标按钮	282
5.2.1	结合 MouseDown 事件和 MouseUp 事件使用 Button 参数	283
5.2.2	结合 MouseMove 使用 Button 参数	284
5.2.3	使用 Button 参数增强图形鼠标应用程序	286
5.3	检测 Shift、Ctrl 及 Alt 键的状态	287

5.4	处理拖放.....	288
5.4.1	启动自动拖动模式	289
5.4.2	改变拖动图标	289
5.4.3	放下对象时的响应	290
5.4.4	启动拖动或停止拖动时的控制	290
5.4.5	改变控件的位置	292
5.5	应用 OLE 拖放.....	293
5.5.1	OLE 拖放概述.....	293
5.5.2	启动自动 OLE 拖放.....	295
5.5.3	使用 DataObject 对象	295
5.5.4	OLE 拖放的工作原理.....	297
5.5.5	启动 OLE 拖动操作.....	298
5.5.6	将 OLE 拖动源到 OLE 拖动目标的上方	300
5.5.7	将 OLE 拖动源放到 OLE 拖动目标上.....	303
5.5.8	用鼠标和键盘修改放下效果和用户反馈	306
5.5.9	创建自定义数据格式	307
5.5.10	从 Windows 资源管理器中拖动文件	307
5.5.11	自定义鼠标指针.....	309
5.6	响应键盘事件.....	310
5.6.1	编写低级键盘处理程序	310
5.6.2	KeyPress 事件	311
5.6.3	KeyUp 和 KeyDown 事件	312
5.6.4	编写窗体级键盘处理程序	314
5.7	中断后台处理.....	316
5.7.1	允许用户中断任务	316
5.7.2	使用 DoEvents 函数.....	317
第六章	处理文本和图形.....	320
6.1	设置字体.....	320
6.1.1	为应用程序选择字体	320
6.1.2	检查可用的字体	321
6.1.3	设置字体特征	321
6.2	在窗体和图片框上显示文本.....	323
6.2.1	使用 Print 方法	323
6.2.2	在单行上显示不同的项	324
6.2.3	在指定位置显示 Print 输出.....	324
6.3	格式化数字、日期和时间.....	326
6.3.1	Format 函数的语法	326
6.3.2	已命名的格式	326

6.3.3	数字格式	327
6.3.4	打印格式化的日期和时间	328
6.4	使用剪贴板对象传送文本和图形	328
6.4.1	选定文本	328
6.4.2	使用剪贴板剪切、复制和粘贴文本	329
6.4.3	使用 ActiveControl 属性	330
6.4.4	在剪贴板上使用多种格式	331
6.4.5	检查剪贴板上的数据格式	332
6.5	坐标系统概述	333
6.5.1	坐标系统概述	333
6.5.2	改变对象的坐标系统	334
6.5.3	刻度转换	337
6.6	使用图形控件	338
6.6.1	图形控件的优点	338
6.6.2	图形控件的限制条件	338
6.6.3	给应用程序添加图片	338
6.6.4	窗体和控件的图形属性	340
6.6.5	用 AutoRedraw 创建持久的图形	341
6.6.6	用 ClipControls 裁剪区域	342
6.6.7	用 AutoRedraw 和 ClipControls 使图形分层	344
6.6.8	动态移动控件	345
6.6.9	动态地改变控件大小	348
6.6.10	创建简单动画	349
6.7	使用图形方法绘制图形	350
6.7.1	图形方法的优点	350
6.7.2	图形方法的限制条件	350
6.7.3	图形方法画图的原理	351
6.7.4	画点	351
6.7.5	画直线和方框	352
6.7.6	画圆、圆弧和椭圆	354
6.7.7	设置绘图位置、线宽、线型和显示	356
6.8	使用颜色	359
6.8.1	使用颜色属性	359
6.8.2	使用 256 种颜色	361
6.8.3	管理多个调色板	362
6.9	使用图片对象	364
6.9.1	使用图片对象数组	365
6.9.2	使用图片对象代替 Windows API	365
6.10	处理打印任务	365

6.10.1	从应用程序中打印	366
6.10.2	使用 Printer 对象	367
第七章	访问数据库	371
7.1	使用 Visual Basic 访问数据	371
7.1.1	Visual Basic 数据访问简述	371
7.1.2	窗体和数据识别控件	373
7.1.3	Visual Basic 中的 ADO、DAO 和 RDO	374
7.1.4	OLE DB 提供者	375
7.1.5	Visual Basic 的数据源	375
7.1.6	Visual Basic 和 Microsoft 事务服务器	377
7.1.7	Visual Basic 中的数据绑定	377
7.1.8	Visual Basic 的远程特性	379
7.1.9	DHTML 和 Visual Basic 数据访问	380
7.2	用 Data 控件访问数据库	381
7.2.1	用 Data 控件能做什么	381
7.2.2	创建简单数据库应用程序的步骤	381
7.2.3	数据库结构	383
7.2.4	记录集对象 Recordset	384
7.2.5	使用数据绑定控件	385
7.2.6	绑定控件的类型	386
7.2.7	添加、更新和删除记录	386
7.3	使用 ADO Data 控件	387
7.3.1	安装 ADO Data 控件	388
7.3.2	ADO Data 控件的用途	389
7.3.3	以最少的代码创建前端数据库应用程序	389
7.3.4	利用程序设置 ConnectionString、Source、DataSource 和 DataField	391
7.3.5	设置 ADO Data 控件的数据库关联属性	392
7.4	使用 DataListBox 和 DataComboBox	393
7.4.1	DataListBox 和 DataComboBox 的用途	393
7.4.2	DataListBox 和 DataComboBox 的绑定属性	394
7.4.3	利用 DataCombo 和 DataList 控件连接两个表	394
7.4.4	DataListBox 和 DataComboBox 的其他属性	396
7.5	使用 DataGrid	397
7.5.1	DataGrid 控件的用途	397
7.5.2	创建 OLE DB 数据链接的步骤	397
7.5.3	设计时设置 DataGrid 控件属性	399
7.5.4	运行时改变数据	401
7.5.5	从 DataGrid 中返回值	402

7.5.6	创建链接到 DataList 控件的 DataGrid.....	403
7.5.7	操作列	405
7.5.8	操作 DataGrid 视图.....	406
7.5.9	利用追踪记录	407
7.5.10	显示计算字段	408
7.5.11	使用 DataGrid 控件和类模块.....	408
7.6	使用 MSFlexGrid 控件.....	411
7.6.1	MSFlexGrid 控件的用途	412
7.6.2	绑定数据到 MSFlexGrid 控件	412
7.6.3	在 MSFlexGrid 控件中排序和合并数据	414
7.6.4	利用 MSFlexGrid 控件创建可进行单元编辑的工作表	418
7.6.5	在 MSFlexGrid 控件中设置带有标头的概要显示	422
7.7	用代码操作记录.....	426
7.7.1	记录集的定位	426
7.7.2	查找指定的记录	427
7.7.3	用代码更新记录集	429
7.7.4	控制事务	432
7.7.5	确认数据库的变化	434
7.7.6	数据访问专用的 Update 方法	435
第八章	ActiveX 部件简述	437
8.1	ActiveX 部件概述	437
8.1.1	ActiveX 部件的类型	437
8.1.2	进程内外的服务器	437
8.1.3	使用 ActiveX 部件	438
8.1.4	创建对对象的引用	439
8.1.5	使用 Visual Basic 可以创建的部件	442
8.1.6	关于 ActiveX 文档	442
8.1.7	关于代码部件	446
8.2	Visual Basic 6.0 提供的 ActiveX 控件	446
8.2.1	安装 ActiveX 控件	447
8.2.2	主要的 ActiveX 控件	447
8.2.3	编译带 ActiveX 控件的程序	452
8.3	开发 Internet 程序	452
8.3.1	Internet 开发环境	453
8.3.2	在 Web 上使用 ActiveX 部件	455
第九章	发布应用程序	461
9.1	Package and Deployment 向导简述	461

9.1.1	Package and Deployment 向导的功能	461
9.1.2	Package and Deployment 向导和 Setu Toolkit 工程的关系	461
9.1.3	启动 Package and Deployment 向导	462
9.2	Package and Deloyment 向导使用详解	462
9.2.1	选择安装目录	463
9.2.2	选择驱动程序	464
9.2.3	选择发布文件	464
9.2.4	选择创建单个或多个.cab 文件	465
9.2.5	输入安装标题	465
9.2.6	选择菜单项	466
9.2.7	调整文件安装目录	466
9.2.8	设置文件共享特性	466
9.2.9	设置正本文件名	467
9.3	调整 Setup Toolkit	468

第一章 认识 Visual Basic 6.0

很长一段时间以来，由于 Basic 程序设计语言具有语法简单、学习容易的特点。因此，该语言一直被作为大多数电脑初学者的首选入门编程语言。但是，随着事件驱动、面向对象、可视化等程序设计概念的导入，Basic 语言也被赋予了新的内涵，这就是 Visual Basic 语言的由来。

实际上，Visual Basic 语言的功能已非常强大，用户可用它方便地完成大部分任务，从界面设计、数据库处理到多媒体控制等。

1.1 Visual Basic 简介

无论用户是一个 Windows 编程新手，还是已经具有丰富的 Windows 程序设计经验，利用 Visual Basic 提供的良好开发环境，用户均可快速地完成所需任务。

何谓 Visual Basic？其中，“Visual”中文意义为“可视化”，它提供了一种创建图形用户接口（GUI）的方法。利用这种方法，用户不必书写复杂的描述接口元素外观和位置的程序代码，而只需将系统提供的对象放在屏幕上的适当位置，并通过属性窗口进行适当的设置，即可设计一个优秀的应用程序界面。

此外，用户除了可以使用单独的 Visual Basic 程序设计环境进行程序设计外。Visual Basic 程序设计系统还被包括在 Excel、Access 等众多 Windows 应用软件中，以供用户进行二次开发。此外，目前使用较多的 VBScript(The Visual Basic Scripting Edition)脚本语言实际上是 Visual Basic 语言的子集。因此，无论从哪方面讲，学习 Visual Basic 都是一件非常合算的投资。

1.1.1 Visual Basic 语言的特点

无论用户是编制一个小的实用程序，还是开发一个大型的专业系统，甚至是开发一个跨越 Internet 的分布式应用系统，Visual Basic 都为用户提供了合适的工具。

- 利用 Visual Basic 的数据访问特性，用户可创建数据库、前端应用程序、适用于大多数流行数据库格式（包括 SQL Server 和其他专业级数据库）的服务器组件。
- 利用 ActiveX 技术允许用户使用其他应用程序提供的功能，如 Microsoft Word、Microsoft Excel，甚至可直接使用由 Visual Basic 专业版和企业版创建的应用程序和对象。
- Visual Basic 的 Internet 功能允许用户通过程序方便地通过 Internet 或 Intranet 访问文档和应用程序，甚至创建 Internet 服务器程序。

- 用户最终创建的程序是一个真正的.EXE 文件，该文件可自由发布。

1.1.2 Visual Basic 的版本

Visual Basic 有三种版本，各自满足不同的开发需要。其中：

- Visual Basic 学习版使编程人员轻松开发 Windows 95 和 Windows NT 的应用程序。该版本包括所有的内部控件，连同 Grid、Tab 和 Data_Bound 控件。
- 专业版为专业编程人员提供了一整套进行开发的功能完备的工具。该版本包括学习版的全部功能连同 ActiveX 控件，还包括 Internet Server Application Designer（Internet 服务器应用程序设计器）、Integrated Visual DatabaseTools（集成可视化数据库工具）和 Data Environment（数据环境）、Active Data Objects（活动数据对象，简称 ADO）和 Dynamic HTML Page Designer(动态 HTML 页设计器)。
- 企业版使得专业编程人员能够开发功能强大的组内分布式应用程序。该版本包括专业版的全部功能连同 Back Office 工具，如 SQL Server、Microsoft Transaction Server、Internet Information Server、Visual SourceSafe、SNA Server 等。

1.2 Visual Basic 程序设计基本概念

为了解使用 Visual Basic 应用程序的开发过程，用户有必要先理解 Visual Basic 赖以创建程序的一些关键概念。因为 Visual Basic 是 Windows 开发语言，所以有必要与 Windows 环境保持一定的相似性。如果不熟悉 Windows 编程，就需要明白在 Windows 环境下编程和在其他环境下编程的一些根本性的差别。

1.2.1 事件驱动编程

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和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。

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他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（例如，假设在运行来处理某一输入字段的过程之前，该输入字段必须包含确定的值），应该组织好应用程序的结构，以确保该假设始终有效（例如，在输入字段中有值之前禁止使用启动该处理过程的命令按钮）。

在执行中代码也可以触发事件。例如，在程序中改变文本框中的文本将引发文本框的 Change 事件。如果 Change 事件中包含有代码，则将导致该代码的执行。如果原来

假设该事件仅能由用户的交互操作所触发，则可能会产生意料之外的结果。正因为这一原因，所以在设计应用程序时理解事件驱动模型并牢记在心是非常重要的。

1.2.2 面向对象程序设计

和传统的面向过程的程序设计思想相比，面向对象的程序设计（OOP）是一种程序设计思路的改变，它对传统的面向过程的程序设计思想作了根本性的变革。在传统的程序设计中，人们需要考虑的是代码的全部流程，而在面向对象的程序设计中，需要考虑的则是如何创建对象以及创建什么样的对象。

例如，用户要设计一个口令输入程序，只需新建一个窗体，然后向该窗体中放置一个标签控件、一个编辑框控件和两个命令按钮控件即可（参见图 1-1）。

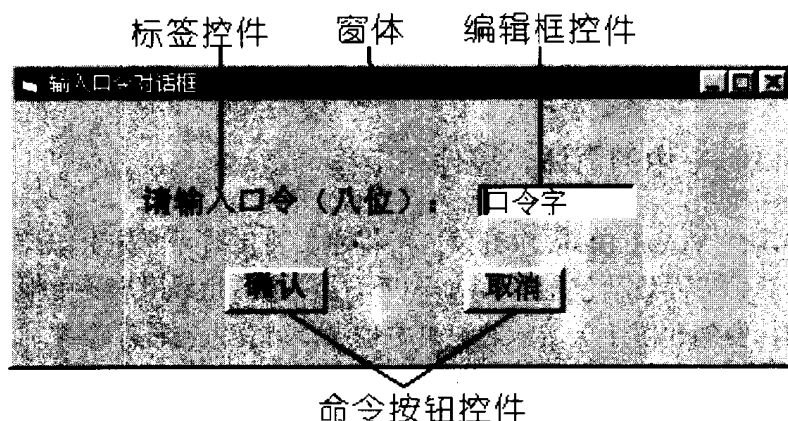


图 1-1 口令输入对话框

在该程序中，窗体、标签控件、编辑框控件和命令按钮控件被称为对象。用户在编程时只需简单地修改这些对象的属性（如提示、字体等），并为两个命令按钮书写简单的事件代码（单击）即可完成任务。

事实上，这些对象均来自系统提供的类。例如，“确定”按钮和“取消”按钮均源自命令按钮类。因此，对象和类密切相关，但是两者并不相同。类包含了有关对象的特征和行为信息，它是对象的蓝图和框架。例如，电话的电路结构和设计布局可以是一个类，而这个类的实例（对象）便是一部电话。

所有对象的属性、事件和方法程序在定义类时被指定，而可在应用时调整。此外，类还有如下特征，这些特征对提高代码的可重用性和易维护性很有用处。

1. 利用类的封装特性简化程序设计

在现实生活中，当用户在办公室内安装一部电话的时候，您也许并不关心这部电话在内部如何接收呼叫，怎样启动或终止与交换台的连接，以及如何将拨号转换为电子信号。您所要知道的全部信息就是您可以拿起听筒，拨打合适的电话号码，然后与您要找的人讲话。在这里，如何建立连接的复杂性被隐藏起来。

对于面向对象的程序设计而言，也同样如此。用户不必具体了解对象在文件的存储结构、工作原理，而只需知道对象的外在特性即可，从而简化了程序设计。因此，所谓封