

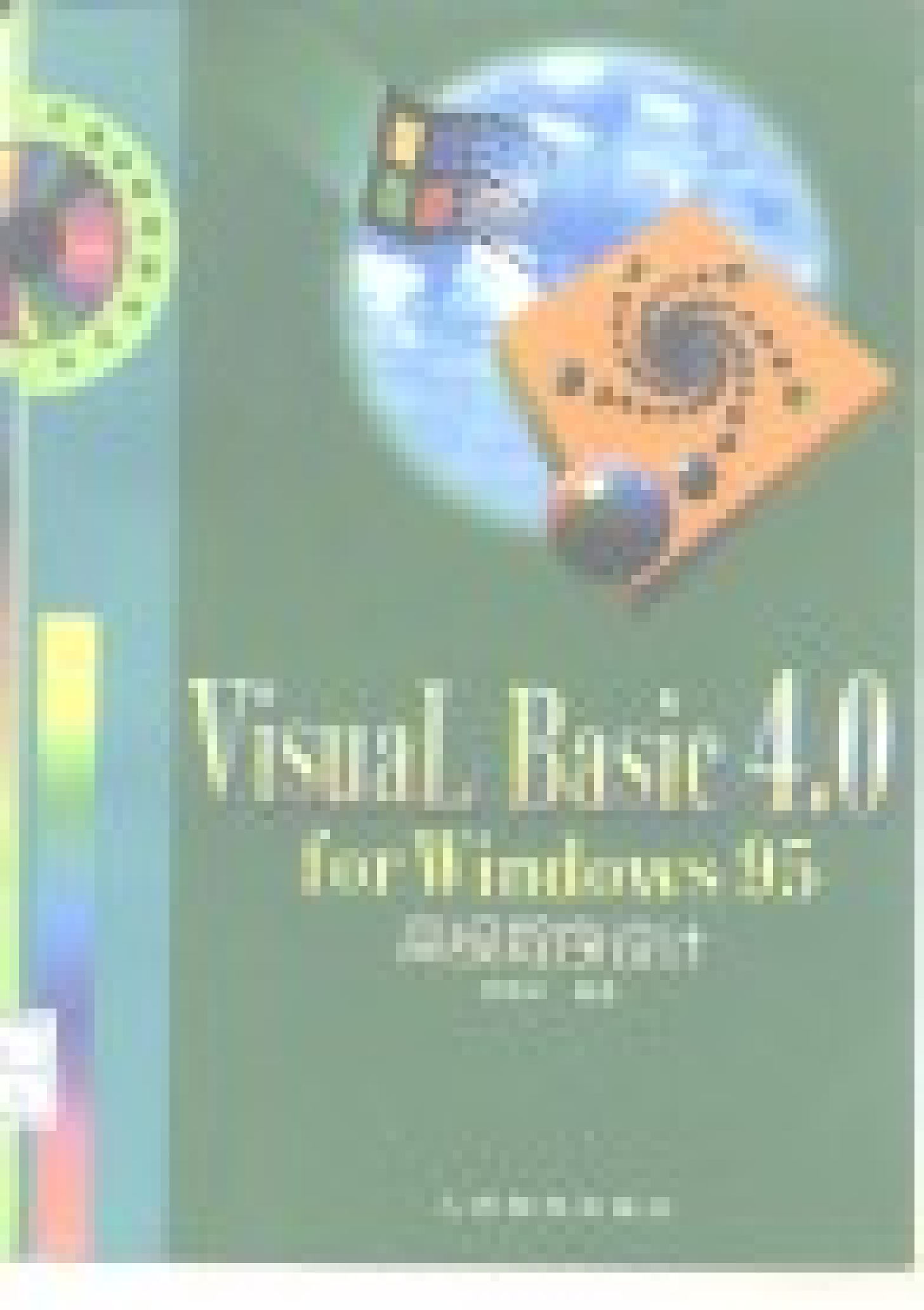
Visual Basic 4.0 for Windows 95

高级程序设计

刘炳文 编著

2
/1

人民邮电出版社



Visual Perception

How We Understand the World

Second Edition

2008

© 2008

© 2008

© 2008

© 2008

© 2008

© 2008

© 2008

© 2008

TP312
LBW/1

计算机实用软件丛书

Visual Basic 4.0 for Windows 95 高级程序设计

刘炳文 编著

人民邮电出版社

图书在版编目(CIP)数据

Visual Basic 4.0 for Windows 95 高级程序设计/刘炳文编著. - 北京:人民邮电出版社,
1998.4

(计算机实用软件丛书)

ISBN 7-115-06846-1

I.V... II.刘... III.Basic 语言-程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(98)第 00257 号

内 容 提 要

Microsoft Visual Basic 4.0 支持 32 位编程和面向对象程序设计,可用于大型应用程序的研制,并能在较短的时间内开发出功能强、质量高的应用软件。

本书是为具有 Visual Basic 基础知识和初步编程经验,并想进一步学习 Visual Basic 程序设计的读者编写的。书中通过大量程序实例,介绍了 Visual Basic 4.0 专业版较为深入的一些功能,包括:面向对象程序设计、常用定制控件、图形程序设计、多文档界面(MDI)、Windows 应用程序接口(API)、数据库的建立和访问、动态数据交换(DDE)、对象的链接与嵌入(OLE)及 Visual Basic 应用程序的发行。

本书可供具有 Visual Basic 初步知识的读者学习使用,也可作为大专院校师生和应用程序开发人员深入学习 Visual Basic 程序设计的参考书。

人民邮电出版社出版的《Visual Basic 4.0 for Windows 95 程序设计基础》介绍了 Visual Basic 程序设计的基础知识,可与本书配套使用。

计算机实用软件丛书

Visual Basic 4.0 for Windows 95 高级程序设计

Visual Basic 4.0 for Windows 95 Gaoji Chengxu Sheji

◆ 编 著 刘炳文

责任编辑 赵桂珍

◆ 人民邮电出版社出版发行

北京崇文区夕照寺街 14 号

北京顺义向阳胶印厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:28.75

字数:714 千字

印数:1-5 000 册

1998 年 4 月第 1 版

1998 年 4 月北京第 1 次印刷

ISBN7-115-06846-1/TP·573

定价:37.00 元

“计算机实用软件丛书”编委会

高级顾问 张效祥 胡启恒

主 任 牛田佳

副主任 李树岭 罗晓沛

特约编委 谭浩强 陈树楷

编 委 (按姓氏笔画排序)

毛 波 方 裕 史美林 孙中臣

孙家骥 刘炳文 刘德贵 吴文虎

张国锋 周山芙 周堤基 钟玉琢

柳克俊 侯炳辉 赵桂珍 聂元铭

徐国平 徐修存 寇国华 戴国忠

丛书前言

JS266/07

随着计算机、通信和信息技术的迅速发展与广泛应用,人类正在进入信息化社会。计算机技术的应用与推广,将直接推动社会信息化的发展;而计算机技术的应用与推广,实质上取决于计算机软件的应用和推广,可以说,没有软件,就没有计算机的应用;学习、使用计算机,从根本上讲就是学习和掌握软件的使用。

为了适应当前计算机技术发展的需要,满足读者学习、使用计算机软件的需求,人民邮电出版社约请有关专家编写出版了这套“计算机实用软件丛书”。

这套丛书的特点是:普及兼顾提高,应用兼顾开发,各书独立成册形成系列,并注重其相关性,使丛书成为广大计算机应用和开发人员学习使用计算机的必备用书。

这套丛书的内容包括:程序设计语言、操作系统技术、数据库技术、软件开发技术及工具、网络技术、多媒体技术等。

在计算机技术飞速发展的今天,软件产品更新快,经常有新产品或新版本问世,因此我们不但介绍当前流行和优秀的软件,而且力求尽快把国内外最新的软件产品也介绍给读者。

我们将全心全意为读者服务,也热切期待广大读者对丛书提出宝贵意见,以进一步提高丛书的质量。让我们共同努力,为提高我国的计算机开发、应用水平做出贡献。

“计算机实用软件丛书”编委会



前 言

1991年,Microsoft公司推出了它的第一个 Visual Basic 版本: Visual Basic 1.0。与目前的版本相比,尽管它还很不完善,但已经引起了人们极大的兴趣。一些专家把 Visual Basic 的出现看作是“令人震惊的新奇迹”,认为它会“极大地改变人们对 Microsoft Windows 的看法和使用 Windows 的方式”,是“90年代完美的编程环境”。

此后,随着机器性能的提高和操作平台的改变,Microsoft公司在四年内先后推出了 Visual Basic 的 2.0、3.0 和 4.0 版。版本的更新,使得 Visual Basic 的功能越来越强,而且更容易使用。特别是 Visual Basic 4.0 版,不仅可以用来为 Windows 95 和 Windows NT 开发 32 位的应用程序,而且使 Visual Basic 成为一种真正的面向对象的程序设计语言。此外,它还允许用户方便地使用第三方提供的工具,从而可以大大扩充 Visual Basic 本身的功能,并给应用软件的开发带来极大的方便。在 Windows 应用程序的开发中,越来越多的用户选择了 Visual Basic 语言。

Visual Basic 之所以受到人们的青睐,原因是多方面的,但主要有两点:一是功能强大,二是容易掌握。Microsoft 公司总裁 Bill Gates 把 Visual Basic 描述为“使用 BASIC 语言的、非常容易使用但功能强大的开发 Windows 应用程序的工具”。Visual Basic 的出现,打破了 Windows 应用程序的开发由专业的 C 程序员一统天下的局面,即使是非专业人员也能胜任,并可在较短的时间内开发出功能强、质量高、界面好的应用程序。

对于几乎所有的 Visual Basic 程序设计人员来说,都希望在初步掌握了 Visual Basic 的程序设计方法之后,能够登堂入室,深入了解 Visual Basic 的功能,开发出更具 Visual Basic 特色、功能更加强大的应用程序,本书正是为了满足这些读者的需求编写的。也就是说,本书所介绍的不是 Visual Basic 的基础知识,而是供有一定 Visual Basic 程序设计经验的读者使用的。对于初学者来说,可以阅读拙作《Visual Basic 4.0 for Windows 95 程序设计基础》(已由人民邮电出版社出版),该书介绍 Visual Basic 的基础知识,主要内容如下:

- Visual Basic 4.0 的新增内容和程序开发环境
- 建立简单的应用程序
- Visual Basic 语法基础
- 窗体与输入输出
- 常用标准控件的使用
- 流程控制和数组

- 过程
- 对话框程序设计
- 菜单程序设计
- 键盘与鼠标事件过程
- 多重窗体与环境应用
- 程序调试与错误处理
- 数据文件
- 文件系统控件及其应用

本书是在《Visual Basic 4.0 for Windows 95 程序设计基础》的基础上编写的,读者可以把两本书放在一起,作为上、下册阅读。

本书共分九章,各章内容如下:

第一章介绍了 Visual Basic 4.0 面向对象的程序设计。以前的版本支持面向对象程序设计,但不是面向对象程序设计语言。Visual Basic 4.0 增加了类模块,直接支持面向对象的程序设计,成为真正的面向对象的程序设计语言。

第二章介绍了常用定制控件的使用。Visual Basic 4.0 提供了大量的定制控件,这些控件具有很强的功能。

第三章讲述 Visual Basic 4.0 的图形程序设计功能。介绍了绘图中使用的坐标系及直线、圆、椭圆及弧的画法,同时介绍了颜色的使用和图形的复制。

第四章介绍了多重文档界面(MDI)。主要讲述如何用 Visual Basic 编写具有 Windows 风格的 MDI 应用程序。

第五章讲述 Windows 应用程序接口,即 API。对于大多数程序设计人员来说,通过 API 来学习 Windows 程序设计是一个好方法。API 提供了 600 多个函数,可大大扩充 Visual Basic 本身的功能。

第六章介绍 Visual Basic 的数据库功能。讲述如何在 Visual Basic 中通过多种方式建立和访问数据库,以及如何建立和输出报表。

第七章和第八章分别介绍了 Visual Basic 与 Windows 中其它应用程序的通信功能,即动态数据交换(DDE)和对象的链接与嵌入(OLE)。

第九章介绍应用程序的发行。

Visual Basic 4.0 分为标准版、专业版和企业版,每种版本都有 16 位和 32 位版本。考虑到大多数读者的需要,本书介绍的是 32 位 Visual Basic 4.0 专业版,要求所使用的操作平台必须是 Windows 95 或 Windows NT 3.51 以上版本。

本书文稿的录入、程序的编写和运行、插图的截取是在 Windows 95 环境下同步进行的,所有程序都已在 Visual Basic 4.0 专业版顺利通过。

在我国,Visual Basic 已被越来越多的用户接受,并在应用程序的开发中日益显示出其优越性。但愿本书能满足广大读者深入学习 Visual Basic 的强烈愿望,对 Visual Basic 在我国的应用和提高能起到一些作用。由于笔者水平有限,时间较紧,可供参考的资料不多,错谬之处在所难免,诚望专家和广大读者批评指正。

目
录

第一章 对象、类和集合	1
1.1 程序设计发展简况	1
1.1.1 程序设计发展的三个阶段	1
1.1.2 从结构化到面向对象	2
1.2 面向对象的程序设计	3
1.2.1 基本概念	3
1.2.2 面向对象方法与程序设计语言	6
1.3 Visual Basic 中预定义的和对象	7
1.3.1 预定义对象和类	7
1.3.2 对象的操作	8
1.4 对象变量	11
1.4.1 对象变量的声明和赋值	11
1.4.2 继承	15
1.4.3 对象变量的释放	19
1.5 对象数组	20
1.5.1 窗体数组	20
1.5.2 控件数组	22
1.6 属性(Property)过程	26
1.6.1 属性过程的定义	26
1.6.2 属性过程举例	30
1.7 初始化事件和终止处理事件	33
1.7.1 Initialize 事件和 Terminate 事件	34
1.7.2 程序举例	34
1.8 建立自己的对象	35
1.8.1 类模块的基本概念	36
1.8.2 类的建立	37
1.9 对象浏览器	44
1.9.1 对象浏览器的功能和构成	44
1.9.2 把注释信息加到对象浏览器中	46
1.10 集合	48
1.10.1 集合的建立	49
1.10.2 程序举例	53

第二章 常用定制控件	59
2.1 定制控件与可插入对象	59
2.1.1 定制控件的插入和删除	59
2.1.2 可插入对象	64
2.2 网格	67
2.2.1 网格的属性	67
2.2.2 网格事件	73
2.3 三维控件	76
2.3.1 三维命令按钮控件(3D Command Button Control)	76
2.3.2 三维组按钮控件(3D Group Push Button Control)	78
2.3.3 三维面板控件(3D Panel Control)	80
2.4 量度表控件	85
2.4.1 量度表控件属性	85
2.4.2 程序举例	87
2.5 旋钮控件	90
2.5.1 事件和属性	91
2.5.2 程序举例	92
2.6 格式文本框控件	93
2.6.1 文件的装入和保存	94
2.6.2 字体设置	98
2.6.3 颜色设置	100
2.6.4 对齐与缩进	101
2.6.5 打印	103
2.7 滑动器	104
2.7.1 设置滑动器的外观	105
2.7.2 滑动器的操作	107
2.8 状态条	111
2.8.1 建立状态条	111
2.8.2 面板对象的操作	117
2.9 进度条	120
2.9.2 进度条的显示位置	120
2.9.2 程序举例	121
2.10 工具条	123
2.10.1 按钮对象和按钮集合	123
2.10.2 工具条的位置	123
2.10.3 在工具条上添加按钮对象	124
2.10.4 按钮对象的大小和操作	128
2.10.5 显示 ToolTips	129

第三章 图形程序设计	133
3.1 对象坐标系统	133
3.1.1 缺省坐标系	133
3.1.2 标准规格	134
3.1.3 自定义规格	136
3.2 对象的动态调整	140
3.2.1 位置调整	140
3.2.2 大小调整与隐现	142
3.3 点与直线	143
3.3.1 清屏与画点	143
3.3.2 画直线	147
3.3.3 线型与线宽	151
3.4 矩形	156
3.4.1 画矩形	156
3.4.2 图案填充	157
3.5 颜色	159
3.5.1 调色板	159
3.5.2 颜色参数	161
3.5.3 前景、背景与图案填充	168
3.6 圆、椭圆和弧	172
3.6.1 画圆	172
3.6.2 画椭圆	175
3.6.3 画弧	177
3.7 画图模式	179
3.7.1 设置绘图模式	179
3.7.2 程序举例	180
3.8 图形的复制	182
3.8.1 PaintPicture 方法	182
3.8.2 程序举例	184
第四章 多重文档界面(MDI)	187
4.1 概述	187
4.1.1 MDI 子窗体	188
4.1.2 引例	189
4.1.3 几点说明	192
4.2 MDI 特性	194
4.2.1 属性、方法和事件	194

4.2.2	Dim 语句和 WindowState 属性	195
4.3	建立 MDI 应用程序	196
4.3.1	建立 MDI 应用程序的一般过程	196
4.3.2	MDI 应用程序举例	198
4.4	MDI 应用程序中的菜单	205
4.4.1	MDI 中菜单的建立	205
4.4.2	MDI 菜单应用程序举例	206
4.5	用 MDI 建立工具条	211
4.5.1	用 MDI 建立工具条的一般步骤	211
4.5.2	程序举例	212
第五章	Windows 应用程序接口(API)	215
5.1	静态链接与动态链接库	215
5.1.1	静态链接	215
5.1.2	动态链接库	216
5.2	在 Visual Basic 中使用动态链接库	217
5.2.1	声明	217
5.2.2	Visual Basic 4.0(32 位版本)中的 DLL 过程声明	220
5.3	API Text Viewer	222
5.3.1	API Viewer 的使用	222
5.3.2	把声明、常量或类型拷贝到 Visual Basic 代码中	226
5.4	API 过程调用	228
5.4.1	调用 API 绘图过程	228
5.4.2	文本输出	230
5.4.3	环境设置	231
5.5	句柄	234
5.5.1	窗口句柄	235
5.5.2	设备描述表句柄	238
5.6	特殊类型数据的传送	239
5.6.1	字符串的传送	240
5.6.2	数组的传送	241
5.6.3	其它数据的传送	242
5.6.4	把 C 文档中的声明转换为 Visual Basic 中的声明	244
5.7	错误处理	245
5.7.1	常见错误	245
5.7.2	需要注意的几个问题	246
5.8	API 调用举例	248
5.8.1	图形拷贝	248
5.8.2	屏幕拷贝	252

第六章 建立和访问数据库	255
6.1 概述	255
6.1.1 什么是数据库系统	255
6.1.2 Visual Basic 的数据库访问	257
6.2 数据库结构	258
6.2.1 基本概念	258
6.2.2 数据访问对象	259
6.3 组织数据	263
6.3.1 把数据分配到表中	263
6.3.2 关系	265
6.4 用 Data Manager 建立数据库	265
6.4.1 建立新的数据库	265
6.4.2 表的修改和索引	269
6.4.3 输入数据	274
6.5 用数据控件访问数据库	277
6.5.1 一个简单例子	278
6.5.2 约束控件(Bound Control)	282
6.6 数据控件的属性、事件和方法	286
6.6.1 数据控件属性	286
6.6.2 数据控件方法	288
6.6.3 数据控件事件	289
6.7 记录的查找、增加和删除	291
6.7.1 记录的查找	291
6.7.2 记录的增加	292
6.7.3 删除记录	294
6.8 用数据访问对象(DAO)建立数据库	295
6.8.1 数据访问对象库	296
6.8.2 打开和关闭数据库	297
6.8.3 获取数据库结构信息	298
6.8.4 建立数据库	304
6.8.5 修改数据库结构	306
6.9 用数据访问对象(DAO)处理数据	308
6.9.1 记录集(RecordSet)	309
6.9.2 记录的增加、修改和删除	313
6.9.3 在记录集中移动	317
6.9.4 查找与定位	319
6.10 编写建立数据库的通用程序	324
6.10.1 建立表和字段	325

6.10.2 编写通用的建立数据库的程序	326
6.11 结构化查询语言 SQL	329
6.11.1 SQL 语句	329
6.11.2 SQL 语句的使用	332
6.12 报表的设计与输出	335
6.12.1 用报表书写器建立报表	335
6.12.2 用报表控件输出报表	341
第七章 动态数据交换(DDE)	347
7.1 基本概念	347
7.1.1 源、目标和会话	347
7.1.2 应用程序、主题和项目	348
7.1.3 DDE 数据的链接方式	349
7.2 在设计阶段建立 DDE 链接	350
7.2.1 目标链接	350
7.2.2 源链接	352
7.3 链接属性	355
7.3.1 目标链接属性	355
7.3.2 源链接属性	358
7.4 链接事件	360
7.4.1 目标链接事件	361
7.4.2 源链接事件	364
7.5 用方法执行 DDE 操作	366
7.5.1 启动其它应用程序(Shell 函数)	366
7.5.2 发送数据(LinkPoke 方法)	367
7.5.3 请求发送数据(LinkRequest 方法)	368
7.5.4 向其它应用程序发送命令(LinkExecute 方法)	370
7.5.5 发送图形信息(LinkSend 方法)	370
7.6 其它应用程序中的 DDE	371
7.6.1 Microsoft 其它应用程序中的 DDE	371
7.6.2 非 DDE 链接	372
第八章 对象的链接与嵌入(OLE)	377
8.1 基本概念	377
8.1.1 链接与嵌入	378
8.1.2 OLE 的版本	379
8.2 OLE 控件的属性、事件和方法	379
8.2.1 OLE 控件的属性	380

8.2.2	OLE 控件的事件	393
8.2.3	OLE 控件的方法	393
8.3	在设计阶段建立 OLE 对象	396
8.3.1	建立链接对象	397
8.3.2	建立嵌入对象	398
8.3.3	OLE 控件弹出式菜单	399
8.3.4	用“选择性粘贴”(Paste Special)对话框建立对象	400
8.4	在运行期间建立 OLE 对象	401
8.4.1	在运行期间建立链接对象	401
8.4.2	在运行期间建立嵌入对象	403
8.4.3	在运行期间插入对象	405
8.4.4	对象的操作	406
8.4.5	数据的保存和装入	407
8.5	OLE 自动化	408
8.5.1	对象与 OLE 自动化	409
8.5.2	CreateObject 和 GetObject 函数	410
8.5.3	访问外部可建立对象	412
8.5.4	访问其它对象	415
8.5.5	OLE 自动化程序举例	417
8.5.6	错误处理	420
第九章	应用程序的发行	423
9.1	概述	423
9.1.1	Visual Basic 应用程序的运行	423
9.1.2	与应用程序一起发行的文件	424
9.2	用 Application Setup Wizard 建立安装程序	425
9.2.1	Setup Wizard 的主要功能	425
9.2.2	用 Setup Wizard 建立发行媒体	426
9.3	建立定制安装程序	434
9.3.1	Setup Toolkit 的功能和相关文件	434
9.3.2	用 Setup Toolkit 建立安装程序的一般过程	435
9.3.3	建立 setup.lst 文件	439
9.3.4	安装盘的建立和测试	442

第 一 章

对象、类和集合

面向对象程序设计(OOP)是目前较为“时髦”的技术,它取代了70年代出现的结构化程序设计技术,代表着程序设计的发展方向。3.0版以前的 Visual Basic 支持面向对象的程序设计,但本身不是面向对象的程序设计语言,因为它只提供了系统预定义类和对象,不能由用户自己定义类和对象。Visual Basic 4.0 是 Microsoft 公司推出的第一个面向对象编程的 Visual Basic 版本。

在这一章中,将简单介绍面向对象的一些基本特征,同时介绍 Visual Basic 是如何支持面向对象程序设计的。

1.1 程序设计发展简况

面向对象方法及编程是在计算机软件技术发展的基础上产生的。在具体介绍 Visual Basic 面向对象程序设计之前,我们先来回顾一下程序设计的发展情况。

1.1.1 程序设计发展的三个阶段

从发展历史来看,程序设计大致经历了三个阶段。

第一个阶段叫做无序态程序设计。在这种模式中,程序员的注意力直接集中在问题求解本身,很少考虑求解的方法。其代表性语言是 BASIC 的低级版本,这种语言没有明显的子程序结构,程序员无从考虑结构性或可扩展性等问题,不得不把精力放在特殊问题上。因此,在无序态程序设计阶段,只能编写小型程序,而且调试和测试比较困难。

第二个阶段称为过程化程序设计。它是对无序态程序设计的改进,主要以 FORTRAN 和 COBOL 语言为代表。在过程环境中,典型的程序由一个主程序和被这个主程序调用的子程序构成。在这种自顶向下程序设计方法中,主程序一般比较短小,它把操作分解到其他程序或子程序中。程序执行一般从主程序顶部开始,并在主程序底部结束。在这种模式中,程序员把一个问题分解成许多个子程序,理论上可以把每个子程序看作是一条语句。子程序本身被抽象化了,于是在源代码和数据之间形成了一道屏障,使得程序较难维护,当对程序进行增加或删除操作时,必须重新开发整个程序,以便包括新的过程。

第三个阶段是结构化程序设计阶段。在这个阶段,代码抽象已超出函数的范围。结构化程序设计类似于过程化程序设计,它把各种控制结构(如 for、while、if 等)作为子功能,以便进行深入的数据抽象。例如,可以把一个 for 循环看作是独立的程序块,因为全部通路都以同一种方式进出,不允许从循环体外转入循环体内,执行一些指令后再转出来。因此,支持结构化程序设计的语言必须像 PASCAL 语言所规定的那样,有一套完整的控制结构,同时不能忽视数据部分。子程序之间的参数按一定规则进行传递,且只允许存取以输入参数形式接收的数据。此外,不提倡用全局变量作为通信参数来传递。这些限制反映出过程化程序设计的弱点,因为如果函数可以存取或修改全局变量,就难以抽象成一个实体。

为了更加准确地描述客观对象,大多数结构化语言都具有定义新的数据类型的功能。例如在 C 语言中,用 enum 或 typedef 等语句定义简单的数据类型,而用 struct 定义结构体数据类型,这能保证数据的一致性,更加接近模型化之后的客观对象。

1.1.2 从结构化到面向对象

模块化程序设计是结构化程序设计风格的延伸。函数作为模块,被分解为独立性更强的可编译源文件。每个模块含有一组唯一由本身访问的数据和函数,一组独立的数据和函数可以被所有模块访问。所有类似的函数归在一个模块之中,所有定义的函数对模块外部并不一定都透明。限制全局定义函数的个数可以减少模块与模块之间通信参数的个数,从而降低整个系统的复杂性。为了能更好地实现模块化,单个函数使用的数据对其他模块应当是不可见的,以实现数据隐藏或封装。同时,用户自己定义的数据类型与原有的类型不同,为了支持数据抽象,必须能为新类型定义内部操作。但是,定义新数据类型及其相关的全部操作是一个复杂的过程,而且根据内部类型定义新类型时,掩盖了用户定义类型之间的关系,不如从现有用户定义类型中派生出新类型简单、清晰。派生类的出现,产生了继承和多态等概念。这样,模块化程序设计语言已不能满足需要,于是出现了面向对象的程序设计语言。

80 年代以后,面向对象语言大量涌现,逐渐形成了两大类,一类是纯面向对象语言,如 Smalltalk 和 Eiffel;另一类是混合型面向对象语言,即在过程语言中增加面向对象的结构,如 C++、Objective-C。一般来说,纯面向对象语言着重于方法研究和快速原型的思想,而混合型面向对象语言着重运行速度和使过程语言的程序员容易接受面向对象的思想。比较成熟的面向对象语言,如 Smalltalk,还提供了健全的类库和丰富的开发工具集,这些功能正逐步融入混合型语言中。

Visual Basic 4.0 是第一个支持面向对象程序设计的 Visual Basic 版本,以前的版本不支持面向对象的程序设计。和 C++ 一样,Visual Basic 4.0 不是纯面向对象的语言,而是一种混合型面向对象的语言。