

Visual Basic 6

使用详解

王道义 乔陶鹏 李颖鹏 宋德强 等编著

- Visual Basic 6基本知识
- Visual Basic 6 常用技巧
- 面向对象编程
- 数据库编程
- Internet 应用

Visual Basic 6使用详解

王道义 乔陶鹏 李颖鹏 宋德强 等编著

抖斗书屋 审



机械工业出版社
China Machine Press

作为当今流行的几大Windows编程语言之一, Visual Basic在界面设计、文件操作、多媒体应用、数据访问、Internet应用等方面都提供了功能强大的工具, 允许开发人员轻松地进行可视化应用程序设计。相对Visual Basic 5.0版本, Visual Basic 6引入了许多新的控件和技术, 功能更加完善, 开发能力更加强大。本书以通俗易懂的语言, 分别介绍了Visual Basic开发的主要专题, 并重点讨论了新版Visual Basic 6的新内容。

全书分为15章, 内容涉及集成开发环境、语言基础、文件处理与操作、错误处理与调试、多媒体应用、面向对象程序设计、创建自己的类及对象、ActiveX技术、数据库编程基础、数据访问控件、ActiveX数据对象、使用Internet Transfer控件进行网络通信、交互式主页制作、基于浏览器的应用程序设计等。

本书是Visual Basic 6应用程序开发的参考书。不同背景读者, 可以灵活选择相关的专题。

版权所有, 翻印必究

图书在版编目(CIP)数据

Visual Basic 6使用详解/王道义等编著. —北京: 机械工业出版社, 1999.7
ISBN 7-111-07264-2

I.V… II.王… III.BASIC语言—程序设计 IV.TP312

中国版本图书馆CIP数据核字(1999)第20827号

出 版 人: 马九荣(北京市百万庄大街22号 邮政编码 100037)

责任编辑: 吴 怡

北京市南方印刷厂印刷·新华书店北京发行所发行

1999年7月第1版第1次印刷

787mm×1092mm 1/16·27.75印张

印数: 0 001-8 000册

定价: 43.00元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

序

自从Microsoft公司在1991年推出Visual Basic 1.0以来, Visual Basic以其快速的程序开发环境、方便的程序界面设计而流行起来。现在, 它已经成为Windows编程的几大设计软件之一, 受到广大研究开发人员的青睐。

Visual Basic 6是Visual Studio 6.0的重要组成部分。相对旧版的Visual Basic, 它在数据访问、Internet应用、控件和语言等方面都有了长足的进步。尤其在数据访问方面, Visual Basic 6结合OLE DB技术, 提出了ActiveX Data Objects(ADO)通用数据访问技术作为新的数据访问标准。不仅如此, Microsoft公司最新版本的Internet Information Server、Internet Explorer、Visual Basic、Visual InterDev、Visual C++和Visual J++都把ADO模型作为优先的数据访问接口。并且Microsoft Office 2000也一样采用ADO作为其标准。因此, 以前版本的用户, 完全有必要对Visual Basic 6进行新的学习。

本书兼顾一些初级用户, 在简单介绍了Visual Basic的基本知识和集成开发环境之后, 开始进入专题的探讨, 主要包括常用专题、面向对象编程、数据库编程和Internet应用等内容, 涵盖Visual Basic应用程序开发的主要方面, 涉及新版Visual Basic的大部分新特点。每个专题在介绍旧版技术的基础上, 充分讨论了新的对象、控件、函数和实现技术。

本书在编写过程中, 基于易学易用的原则, 在系统介绍专题的基础上, 列举了大量的示例, 力求使读者通过开发应用来充分学习Visual Basic 6的新技术、新特点。

本书由中科辅龙计算机技术有限公司抖斗书屋策划, 王道义、乔陶鹏、李颖鹏、宋德强编写了大部分内容, 其他参加编写的人员有郭美山、史惠康、吴洁萍、郭震宇、王金玉、毛德柱、杨蓉、南莲花、杨峰、田伟、杨楠、罗晓宁、侯伟钢、张敬、曹凯、刘靖华、袁芳盛、唐立彬、袁清垒。

在本书编写过程中, 张岩、孟宪强、刘志强提供了许多宝贵建议。同时还要感谢赵伟峰、张赫权、周纯林、李浩然、王浩、吴健等。

抖斗书屋坐落于中科院计算所院内, 由中科辅龙计算机技术有限公司领导, 是一家拥有雄厚势力的计算机图书创作单位。在本书的编写过程中, 书屋的全体员工都付出了大量劳动, 借此机会对书屋全体人员的精诚团结表示由衷的感谢!

由于时间仓促、作者水平有限, 本书错漏之处在所难免, 欢迎广大读者批评指正。联系电话: 62565533 - 3301。

史惠康

1998年冬于中科院计算所

全书导读

本书的一大特点是体系性很强，一共包括五个部分，涉及了Visual Basic 6的主要专题。对于不同的读者，可以视具体情况进行灵活选择阅读。

读者情况

使用过Visual Basic 5.0版本

没有使用过Visual Basic 5.0版本，
但使用过以前的版本

没有使用过任何版本的Visual Basic

只需要了解相关专题的读者

建议阅读方法

你已经具备Visual Basic的基本知识。可以跳过第一部分，直接阅读后面的新内容

第1章和第3章的内容可略读，在阅读第2章熟练新的集成开发环境以后进入专题的应用

建议你从头读起。第1、2、3章尤为重要，从这里可以学习到创建Visual Basic应用程序的基本步骤和方法

可以直接进入相关专题。对于设计的基本控件和方法，进行查阅即可

内容介绍

本书的前三章是基本介绍，包括Visual Basic概述、Visual Basic 6新特点、集成式开发环境(IDE)和Visual Basic语言基础等开发Visual Basic应用程序必备的知识。

从第4章开始进入Visual Basic开发的专题。介绍了一些常用的专题，包括文件管理与操作、错误处理与调试、多媒体应用等内容。Visual Basic 6对这些常用功能的实现方法进行了完善，如对文件处理提出了一些新的对象和函数。

第7至9章是关于面向对象编程(OOP)的高级专题。面向对象编程为Windows编程提供了一个崭新的概念，涉及到用户接口和彼此之间的协作效率。首先介绍了OOP的基本概念和内建对象(第7章)，接着利用Visual Basic创建自己的类及对象，其中包括向已有类中添加属性以创建新类、创建完全属于自己的类以及OLE的基本概念和方法三个部分(第8章)，最后探讨了ActiveX的概念，包括ActiveX控件和ActiveX部件(第9章)。对于新版Visual Basic的新对象和新技术，在这三章都有所讨论。

第10至12章介绍Visual Basic的数据访问，主要是数据库数据的利用。第10章主要涉及了一些数据库编程的基础问题，包括数据库编程的基本概念和方法以及结构化查询语言SQL。第11章开始进入ADO新技术的介绍，首先讨论了新的数据访问接口下新的控件，包括ADO数据控件、DataGrid控件、DataList控件和DataCombo控件。接着在第12章中详细分析了ADO访问本地数据库的对象模型和方法，最后简要介绍了远程数据访问。数据访问是Visual Basic 6发展的重点主题之一。

全书的最后三章是Visual Basic在Internet方面的应用专题。第13章主要介绍了相关控件的使用，包括Internet Transfer Control控件和WinSock控件。第14章讨论交互式主页的实现方法，主要通过CGI或VBScript实现主页的动态变化。最后一章讲述一种类似于交互式网页的应用程序，使用浏览器来作为应用程序的窗体，包括DHTML、ASP和IIS应用程序三部分内容。Internet应用是新时代的要求。新版Visual Basic在Visual Basic 5.0的基础上，加强了Internet应用的能力。

目 录

序

全书导读

第1章 Visual Basic 6概述1

1.1 Visual Basic 6简介1

1.1.1 Visual Basic的发展1

1.1.2 Visual Basic的优点和不足2

1.1.3 三种版本2

1.2 对象和面向对象编程3

1.2.1 对象3

1.2.2 对象的属性和方法3

1.2.3 窗体5

1.2.4 控件7

1.2.5 面向对象编程8

1.3 事件和事件驱动编程8

1.3.1 事件8

1.3.2 焦点事件8

1.3.3 键盘事件10

1.3.4 鼠标事件12

1.3.5 关于窗体的一些事件14

1.3.6 事件驱动编程14

1.4 创建一个Visual Basic应用程序14

1.4.1 总体设计15

1.4.2 创建应用程序界面和设置属性15

1.4.3 编写程序代码18

1.4.4 产生可执行文件22

1.5 Visual Basic 6的新特点23

1.5.1 原始代码编译和速度优化的加强23

1.5.2 数据访问的发展24

1.5.3 Internet应用的发展25

1.5.4 控件的发展26

1.5.5 程序设计语言的发展26

1.6 小结27

第2章 Visual Basic 6的集成开发环境29

2.1 建立新项目29

2.2 使用菜单栏31

2.3 使用工具栏32

2.4 使用控件工具箱33

2.5 使用属性窗口35

2.6 使用项目资源管理窗口35

2.7 使用代码窗口36

2.8 定制自己的集成开发环境37

2.8.1 设置代码编辑器38

2.8.2 设置窗体客户区39

2.8.3 设置集成开发环境的桌面40

2.8.4 设置应用程序的环境41

2.8.5 设置应用程序的编译方式和

中断条件42

2.9 使用Visual Basic 6的帮助系统43

2.9.1 MSDN简介43

2.9.2 进入MSDN帮助系统44

2.9.3 使用MSDN帮助系统45

2.9.4 其它帮助48

2.10 小结48

第3章 Visual Basic程序设计语言50

3.1 Visual Basic的数据类型50

3.1.1 数字型(Numeric)51

3.1.2 字符串型52

3.1.3 日期型53

3.1.4 对象型53

3.1.5 变体型54

3.1.6 类型变换54

3.1.7 变量的类型检查55

3.2 变量和常量56

3.2.1 变量56

3.2.2 常量58

3.2.3 用户自定义数据类型59

3.3 数组59

3.3.1 定长数组59

3.3.2 动态数组	60	4.2.3 文件列表框	87
3.4 集合	61	4.2.4 文件系统管理操作举例	87
3.4.1 声明集合	61	4.3 公用对话框的文件操作功能	93
3.4.2 集合的属性和方法	61	4.3.1 文件操作公用对话框概述	93
3.5 运算符	63	4.3.2 几个重要属性	94
3.5.1 算术运算符	63	4.3.3 公用对话框文件操作示例	98
3.5.2 关系运算符	63	4.4 文件系统对象	104
3.5.3 字符串运算符	64	4.4.1 文件系统对象的使用方法	104
3.5.4 逻辑运算符	65	4.4.2 FSO对象使用举例	106
3.5.5 优先级别	65	4.5 文件的访问	113
3.6 程序流程控制语句	65	4.5.1 文件的打开与关闭	114
3.6.1 判断语句	65	4.5.2 顺序文件	115
3.6.2 循环	67	4.5.3 随机文件	116
3.6.3 控制嵌套	68	4.5.4 二进制文件	118
3.6.4 强制退出	68	4.5.5 文件的共享	120
3.7 模块、子程序和函数	69	4.6 文件与剪贴板	121
3.7.1 模块	69	4.7 小结	123
3.7.2 子程序	70	第5章 错误处理与调试	124
3.7.3 函数	71	5.1 错误类型	124
3.7.4 变元和变元匹配	71	5.1.1 编译错误	124
3.7.5 变元传递机制	73	5.1.2 运行错误	125
3.8 输入/输出函数	74	5.1.3 逻辑错误	126
3.8.1 InputBox函数	74	5.1.4 分析错误类型的意义	126
3.8.2 MsgBox函数	76	5.2 调试功能与调试工具	126
3.9 Visual Basic程序的书写	77	5.2.1 程序所处的模式	126
3.9.1 注释	77	5.2.2 调试工具	132
3.9.2 缩进和对齐	77	5.2.3 调试的一些特殊问题	137
3.9.3 续行	78	5.3 错误对象与错误处理	138
3.10 小结	78	5.3.1 错误对象	138
第4章 文件管理与操作	80	5.3.2 错误处理	140
4.1 与文件系统操作有关的语句、函数 与对象	80	5.3.3 错误处理举例	142
4.1.1 关于文件系统管理维护的语句	80	5.3.4 调试与错误的矛盾	145
4.1.2 关于文件系统管理维护的函数	83	5.4 小结	146
4.1.3 文件系统对象	84	第6章 多媒体	147
4.1.4 对文件操作的语句与函数	85	6.1 多媒体的基本概念	147
4.2 文件系统控件	85	6.1.1 声音与图形	147
4.2.1 驱动器列表框	86	6.1.2 音乐与动画	148
4.2.2 目录列表框	86	6.2 多媒体控件的使用	149
		6.2.1 Multimedia MCI控件	149

6.2.2 其他多媒体控件	156	8.2.1 属性	202
6.3 小结	161	8.2.2 方法	203
第7章 面向对象程序设计	162	8.2.3 关于属性和方法的部分问题的讨论	204
7.1 面向对象程序设计简介	162	8.2.4 事件	207
7.1.1 对象的概念	162	8.3 数据感知的类	211
7.1.2 面向对象的程序设计方法	163	8.3.1 数据源	211
7.1.3 Visual Basic的面向对象特性	163	8.3.2 数据使用者	212
7.2 操作Visual Basic中的对象	164	8.3.3 BindingCollection对象	212
7.2.1 对象变量的基本概念	164	8.4 创建集合类	213
7.2.2 Set关键字	166	8.5 Visual Basic Class Builder应用程序	215
7.2.3 New关键字	167	8.6 类的持续性	216
7.2.4 释放对象及对象变量	168	8.6.1 设置类的持续性	217
7.2.5 查找对象所属的类	168	8.6.2 用PropertyBag对象使一个对象具有持续性	218
7.2.6 通过变量名来调用对象的属性和方法	169	8.6.3 利用具有持续性的类	219
7.2.7 对一个对象执行多个操作	169	8.7 OLE技术	219
7.2.8 使用默认属性	170	8.7.1 什么是OLE	219
7.2.9 新添对象	172	8.7.2 OLE容器控件	220
7.3 控件数组	178	8.8 小结	226
7.3.1 直接引用控件数组	179	第9章 ActiveX	227
7.3.2 利用对象变量数组引用控件数组	180	9.1 ActiveX控件	227
7.3.3 添加或删除控件数组中的元素	180	9.1.1 与ActiveX控件相关的基本内容	227
7.4 集合	182	9.1.2 建造ActiveX控件的方法	229
7.4.1 内建集合	182	9.1.3 创建ActiveX控件的步骤	229
7.4.2 用Collection类创建自己的集合	182	9.1.4 创建由控件项目和测试项目组成的项目组	230
7.4.3 Visual Basic 6中新添的集合	185	9.1.5 设计控件的外观	232
7.5 对象浏览器	185	9.1.6 实现控件的接口和功能	233
7.5.1 对象浏览器的内容及使用	185	9.1.7 设计和实现控件的属性页	235
7.5.2 添加对象的描述	187	9.1.8 编译控件部件	236
7.6 对象编程的典型例子—MDI窗体	188	9.2 ActiveX部件	237
7.6.1 创建MDI窗体及其子窗体	188	9.2.1 部件的特性及部件设计的一般准则	237
7.6.2 MDI内置功能	189	9.2.2 创建ActiveX部件项目	238
7.6.3 MDI窗体及其子窗体的菜单	191	9.2.3 显示类的窗体	238
7.6.4 结束MDI应用程序	193	9.2.4 创建测试项目	239
7.6.5 生成MDI编辑器	194	9.2.5 共享类	240
7.7 小结	197	9.3 小结	241
第8章 创建自己的类及对象	199		
8.1 创建自己的Visual Basic对象	199		
8.2 建立自己的类	201		

第10章 数据库编程基础	243	11.5.2 功能实现	290
10.1 数据库的基本概念	243	11.5.3 运行程序	294
10.1.1 什么是数据库	243	11.6 小结	295
10.1.2 基本术语	244	第12章 ActiveX数据对象	296
10.1.3 数据库的规范化	245	12.1 Visual Basic的数据访问接口	296
10.1.4 记录集	246	12.1.1 数据访问接口的发展	296
10.1.5 Visual Basic数据库编程	247	12.1.2 DAO的分层结构	297
10.2 规划数据库	249	12.1.3 RDO对象模型	299
10.2.1 规范化	249	12.1.4 ADO对象模型	300
10.2.2 指定索引	250	12.1.5 Visual Basic数据访问接口小结	301
10.2.3 建立表的关联性	250	12.2 ADO操作数据的步骤	301
10.3 可视数据管理器	250	12.3 Connection对象	302
10.3.1 VisData的工作环境	251	12.3.1 明确Connection对象的属性	302
10.3.2 创建数据表	252	12.3.2 创建一个链接	303
10.3.3 建立查询	254	12.3.3 执行和取消操作	304
10.3.4 建立数据窗体	256	12.3.4 断开链接	305
10.4 结构化查询语言SQL	257	12.3.5 使用Connection对象示例	305
10.4.1 SQL简介	257	12.4 Recordset对象	305
10.4.2 SQL语句的基本元素	257	12.4.1 创建Recordset对象	306
10.4.3 DDL操作	259	12.4.2 操作记录	307
10.4.4 DML操作	262	12.4.3 复制Recordset对象	310
10.5 小结	268	12.4.4 保存Recordset对象	310
第11章 数据访问控件	269	12.5 操作字段	311
11.1 数据访问控件	269	12.5.1 Fields集合	311
11.1.1 数据访问控件介绍	269	12.5.2 操作Field对象	311
11.1.2 数据绑定和数据绑定控件	270	12.6 Command对象	312
11.2 ADO数据控件	272	12.6.1 设置Command对象的属性	312
11.2.1 ADO数据控件简介	272	12.6.2 执行命令	312
11.2.2 ADO数据控件的主要属性	273	12.6.3 创建Parameter对象	313
11.2.3 使用ADO数据控件	276	12.7 Parameter对象	313
11.3 DataGrid控件	280	12.8 错误信息	313
11.3.1 显示数据	280	12.9 应用ADO对象访问数据	314
11.3.2 操作记录数据	283	12.9.1 主窗体MDIForm1	315
11.4 DataList控件和DataCombo控件	284	12.9.2 打开数据库和数据表	316
11.4.1 简介	284	12.9.3 编辑记录数据	318
11.4.2 定位显示记录	285	12.9.4 SQL查询	322
11.4.3 操作记录数据	287	12.10 远程数据访问简介	322
11.5 应用数据访问控件	288	12.11 小结	324
11.5.1 界面设计	289	第13章 网络通信	325

13.1 TCP/IP通信的基础知识	325	14.3.1 VBScript介绍	381
13.2 使用Microsoft提供的ActiveX控件 进行网络通信	327	14.3.2 VBScript初步	381
13.3 InternetTransfer Control 控件的使用 方法	327	14.3.3 使用VBScript实现页面的 动态响应	383
13.3.1 控件特点	327	14.4 小结	392
13.3.2 使用前的基本设置	328	第15章 基于浏览器的应用程序	393
13.3.3 连接请求及数据传输	330	15.1 DHTML应用程序	393
13.4 使用WinSock控件进行网络通信	345	15.1.1 创建DHTML应用程序	393
13.4.1 控件特点	346	15.1.2 配置封装DHTML应用程序	395
13.4.2 使用前的基本设置	347	15.2 Active Server Pages(ASP)	396
13.4.3 使用TCP协议进行通信	348	15.2.1 概述	396
13.4.4 使用UDP协议进行通信	355	15.2.2 ASP的工作过程	396
13.5 小结	359	15.2.3 在ASP中结合Scripts脚本	397
第14章 交互式主页制作	360	15.2.4 在ASP中结合对象和组件	399
14.1 交互式主页概述	360	15.2.5 利用ASP的内嵌对象编程	400
14.2 CGI与WinCGI	361	15.3 IIS应用程序	403
14.2.1 表单的组成元素	362	15.3.1 关于IIS应用程序的基本概念	403
14.2.2 制作表单	367	15.3.2 简单的IIS应用程序制作实例	404
14.2.3 WinCGI编程	369	15.4 小结	408
14.3 在主页中直接使用VBScript脚 本语言	381	附录A Internet Transfer控件参考	409
		附录B Winsock控件参考	420
		附录C VBScript脚本语言基础	430

第1章 Visual Basic 6概述

本章首先介绍Visual Basic的发展、Visual Basic的优点和不足,以及它的三种版本。接着将讨论Visual Basic作为Windows编程工具的基本特征:面向对象编程和事件驱动编程。其中解释了对象和事件的概念,分析了几种常用对象的属性、方法和事件。然后利用已有的基本知识,创建了一个简单的Visual Basic程序,从而说明了Visual Basic应用程序创建的基本步骤。最后介绍Visual Basic 6相对以前版本的新特征。本章的内容要点如下:

- Visual Basic 6简介;
- 对象和面向对象编程;
- 事件和事件驱动编程;
- 创建一个Visual Basic应用程序;
- Visual Basic 6的新特征。

1.1 Visual Basic 6简介

在利用Visual Basic进行编程开发工作之前,对Visual Basic的发展过程、优缺点和版本结构有一个大致的了解是必要的。本节将提供这方面的内容。

1.1.1 Visual Basic的发展

Basic语言是20世纪60年代由Dartmouth学院的两位教授John G. Kemeny和Thomas Kuntz发明的。发明Basic最初的目的是为了更方便计算机教学,相对于速度和效率,他们更强调简洁性。Basic语言对程序的结构及变量的使用要求不严格,是一种直观的非结构化语言,易于理解,易于学习使用。

但是,早期Basic这种解释性语言的易于使用是以它的速度为代价的。在程序运行的时候,必须先将Basic语言翻译成计算机可以理解的代码解释执行,而不是预先编译为机器码生成可执行文件来执行。因此,它的早期版本运行起来都很慢。Basic语言甚至被认为只是一种“玩具”语言,不适合重要工作。

直到20世纪80年代末,被誉为“Visual Basic之父”的Alan Cooper在Basic语言基础上成功地建立了很多Windows开发工具思想和概念,将Basic的易学易用与可视化编程方法及事件驱动结合起来,Visual Basic才日趋流行起来。现在,Visual Basic已经成为Windows编程的几大设计软件之一,受到广大研究开发人员的青睐。

归纳起来,Visual Basic的发展历史如下:

- 60年代初,Dartmouth学院的两位教授John G. Kemeny和Thomas Kuntz发明Basic语言;
- 1976年,Bill Gates和Paul Allen开发出了Basic语言的早期版本;
- 80年代初,微软公司推出了适合IBM PC的GW-Basic语言;
- 80年代中期,微软公司推出了快速、易学易用的QuickBasic版本;
- 1991年,微软公司开发出了面向图形用户界面、具有事件驱动编程机制的Visual Basic 1.0;

- 1992年, 微软公司推出Visual Basic 2.0版;
- 1993年, 微软公司推出Visual Basic 3.0版, 增加了数据库开发和对象链接与嵌入技术(OLE)等功能;
- 1995年, 微软公司推出Visual Basic 4.0版, 着重改善了程序设计环境, 与Windows 95紧密结合;
- 1996年, 微软公司推出Visual Basic 5.0版, 增加或增强了Internet开发、数据访问、程序调试、ActiveX控件等方面的功能; 支持Internet开发和原始码编译, 是Visual Basic 5.0的两大特点;
- 1998年, 微软公司推出Visual Basic 6版。

1.1.2 Visual Basic的优点和不足

Visual Basic是“可视化的Basic”, “Visual”指可视化, 即开发图形用户界面(GUI)的方法。使用这种方法, 无需编写大量代码去描述界面元素的外观和位置, 只要把预先建立的对象(如命令按钮、文本框)拖放到屏幕上需要的位置即可。也就是说, Visual Basic是一种面向图形用户界面的, 交互性很强的可视程序设计工具。但是, 对于Visual Basic只提可视化可视程序设计还远远不够。相对原有的Basic语言, Visual Basic在面向对象编程和事件驱动机制方面实现了革新性的进步。面向对象编程支持创建可重用组件, 使得Windows编程变得十分容易。事件驱动编程模拟了真实世界处理任务的方式, 为用户提供事件触发的权利, 使程序代码能在需要时执行操作。

作为Windows编程语言, Visual Basic可以实现其它任何Windows编程语言的功能, 所设计的程序具有Windows环境的五大优点, 即标准的图形用户界面、动态链接(DDL)、多任务、设备独立性及直接操作特性。

相对其它Windows编程语言, Visual Basic又有自己的优势。它最显著的优点就是具有快速的程序开发环境, 能迅速有效地编制程序界面。Visual Basic的图形用户界面程序量很小, 只需要按照用户的设计风格 and 实际要求修改窗体、控件等对象的属性。无疑, VC的程序设计工具更专业一些, 但其界面编程量大, 参数修改不直观。对于实用软件开发和工程一般应用, 只要不涉及开发程序设计工具或环境, 选择Visual Basic程序设计工具, 开发周期短, 见效快。

但是, Visual Basic的内核仍然是Basic语言, 运行速度较低。使用Visual Basic的一个有效方法是将Visual Basic和其它开发工具(如C语言)结合起来。利用Visual Basic做程序界面, 而一些经常变化的部分, 则用其它语言编制动态链接库, 并由Visual Basic调用, 以优化速度。另外, 新版Visual Basic提供了原始码编译和速度优化功能, 可以大幅度提高程序运行速度。

1.1.3 三种版本

不论是专业开发人员还是初学者, Visual Basic 6都为他们提供了整套编程工具。他们都可以在这套工具中各取所需。Visual Basic 6有三种版本, 学习版(Learning)、专业版(Professional)和企业版(Enterprise), 各自满足不同的开发需要。

• 学习版是入门版本, 包含Visual Basic所有主要的功能和所有的内部控件, 使编程人员能够轻松开发Windows 95和Windows NT的应用程序。对于那些想尝试Visual Basic或者刚刚接触程序设计的人员而言, 选购Visual Basic的学习版不失为一种很好的选择。

- 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控件等专业开发所需的控件集合。对很多编程人员而言, 专业版几乎可以满足其所有要求。

- 企业版是最高级的版本, 它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自动化管理器、部件管理器、数据库管理工具, 等等。

1.2 对象和面向对象编程

为了便于理解Visual Basic应用程序开发过程, 首先简要介绍一下Visual Basic的一些重要概念。因为Visual Basic是为Windows这样的图形用户界面的操作系统开发的, 所以它具有Windows编程的主要特征。Windows编程的最基本特征, 包括两个方面: 面向对象编程和事件驱动编程。

1.2.1 对象

窗体和控件是任何Windows应用程序接口的基本元素。在Visual Basic中, 这些元素被统称为对象。对象是所有面向对象程序设计的核心。Visual Basic作为面向对象编程的语言, 当然也是以对象为中心的。可以认为, 在Visual Basic的程序设计中, 对象和对象的概念无所不在。因此, 在介绍窗体和控件这两种Visual Basic基本对象之前, 首先应该建立对象的基本印象。

其实, 对象并不是一个很特殊、很神秘的概念。在日常生活中, 人们无时无刻不在接触不同的对象。如马路上跑动的汽车、树上的水果、课桌里的书本。在面向对象的程序设计中, 对象的概念就是对现实世界对象的抽象。如果稍加归纳, 就会发现现实世界的对象具有一些共有的特征: 它们具有自己的特定状态, 如汽车的型号、颜色、是否有油; 它们还有自己的行为, 如汽车可以启动、在马路上跑、停止。

同样, 在Visual Basic编程语言中, 对象也有类似的特征。Visual Basic对象是数据和数据操作方式的综合体。数据就是对象的状态; 数据操作方式就是对象的行为。一般地说, 对象指将数据和处理该数据的过程进行打包, 形成的新的数据类型。对象中的数据称为对象的属性。对象中处理数据的过程称为对象的方法。

下面介绍对象的一些基本特征, 即属性和方法。

1.2.2 对象的属性和方法

对象具有属性和方法, 并响应外部事件。属性、方法和事件, 是对象的三大要素。要了解和使用的对象, 必须涉及它们的含义和使用方法。

1. 对象的属性

对象中的数据称为对象的属性。对象的属性决定了所创建的应用程序中对象的表象和行为。改变对象的属性就可改变对象的特性。用面向对象编程的观点来看, 属性是对象的外部可见数据。设置对象的属性, 实际是在向对象添加某些可见的数据。编程人员对于对象的了解和控制被限制在对象提供的属性之中, 对象内部的结构则无从知道。

表1-1列出了Visual Basic编程语言中大多数对象, 如文本框、按钮和窗体, 都具有的一些属性。

注意 Name属性和Caption属性是两种不同的属性。对于控件来说, Name属性表示控件

的名称，用于访问控件的属性和方法。Caption属性只是控件显示的标题。

如果改变了对象的Caption属性，只会影响对象在窗体上显示的文字。而一旦改变了对象的Name属性，将会导致所有涉及该对象的引用都发生改变。这一点在代码的应用程序的设计中必须注意。否则，会出现意想不到的错误。

表1-1 对象的属性

属 性	说 明
Name	设置控件的名称，用于访问控件的属性和方法
Appearance	取0表示平面外观；取1表示三维外观
BackColor	设置显示文字或绘图时的背景颜色
ForeColor	设置前景色，即线条颜色和文本颜色
Font	设置控件文本所用字的字体、字号等
Caption	设置许多不接受输入控件显示的文本，或命令按钮的标题，或复选框与选项控件旁边显示的字符串
Text	设置接受输入的控件上显示的文本
Width, Height	设置对象的尺寸。默认单位是twips(1in=1440 twips)
Left, Top	设置对象左上角的坐标。默认单位是twips
Enable	设置为True，表示控件可以变为焦点；设置为False，关掉控件，不允许输入
Visible	设置对象的可见性

Visual Basic的编程需要花费大量时间处理对象的属性。对象的属性，可以在建立对象时设置，也可以在程序运行时通过程序代码设置。通常，属性是在建立对象时设置，如利用属性窗口定制，此时只要单击该对象，接着在属性窗口选中需要修改的属性(Visual Basic的对象都有默认属性，如第一个窗体的Name属性默认为Form1)，然后在属性值一栏中选择或输入新的属性即可。此外，还可以在程序运行时使用属性访问语句对属性进行访问。具体做法是将属性名附在对象名之后，中间用小数点隔开。例如，对于文本框对象Text1，可以用下列表达式访问其Text属性：

```
Text1.Text
```

赋给文本框对象Text属性或将它的Text属性赋给其它对象或变量。

2. 对象的方法

对象的方法指对象处理数据的过程，要执行的动作，即对象不需要编程人员帮助就可以完成的操作。方法隐藏了对象特性实现的具体细节，编程人员只要调用这些，它们就可以实现这些特性。例如调用窗体的Cls方法可以清除窗体；调用Hide方法可以隐藏窗体。

有些方法是简单的动词，直接告诉对象要进行的操作，如Clear方法可以告诉控件放弃它的内容。使用时，只要指定所用的控件名即可：

```
控件名.Clear
```

而其它一些方法，如AddItem和RemoveItem，则要复杂些。AddItem和RemoveItem用于操作列表框和组合框中的项目。应用程序发出AddItem方法，控件就会负责向表中添加或插入新的项目。使用它时，除了指定控件名外，还要指定所加的项目。例如，要向列表框控件1stStudents中加入项目“Zhang Yi”，使用下面的语句：

```
1stStudents.AddItem"Zhang Yi"
```

表1-2列出了一些重要的方法及其使用该方法的一般语法。

表1-2 一些重要的方法

方 法	语 法	说 明
Cls	对象名.Cls	清除窗体和图片框中运行阶段产生的文本或图形
Hide	窗体名.Hide	隐藏窗体, 但不影响窗体的使用
Show	窗体名.Show	在屏幕上前台显示窗体
Move	对象名.Move left, top,width,height	移动对象。四个参数只有left是必需的, 但要设置后面的参量, 前面的必须先设置
Print	对象名.Print “要打印的表达式”	在窗体或图片框中打印文本

1.2.3 窗体

从本小节开始介绍Visual Basic编程的两个最基本的对象：窗体和控件。

Visual Basic中的主要构成单位是窗体(Form)。它是Visual Basic应用程序中的顶层对象，每个应用程序都是从窗体开始的。简单地说，窗体指可以用作定制应用程序界面的窗口，或者用作向用户收集信息的对话框，用户可以通过向窗体增加控件、图形或图片来创建自己的应用程序界面。如果需要，窗体可为不可见。实际上，窗体是一种用于组织相关项的方法，或者说是用于组织程序的一种机制。

窗体根据其交互性可以分为两类：模态窗体和常态窗体。在程序运行过程中，模态窗体禁止与用户进行交互，常态窗体则允许用户交互。

1. 模态窗体

模态窗体完全占有当前应用程序控制，禁止与用户进行交互，只有在关闭模态窗体之后才能切换到同一个应用程序的其它窗体。模态窗体必须具备关闭窗体的方法，以便返回装入模态窗体的窗口。一般应用程序的出错提示窗口都设计成模态窗体，只有单击OK、Cancel或关闭按钮之后，才能切换到主窗口，继续应用程序。

注意 模态窗体只是关闭了当前应用程序的其它部分，而没有关闭其它应用程序。显示模态窗体的时候，不能切换到同一个应用程序的其它窗体，但可以切换到其它应用程序的窗体。

2. 常态窗体

常态窗体，也即非模态窗体，是允许用户交互的窗体，允许切换到任何应用程序的其它任何窗体。常态窗体的地位是对等的，用户可以任意选择常态窗体进行交互，如输入文本、单击命令按钮。

图1-1表示了一个空白窗体，这是运行时的状态。

窗体有一些固有的功能，不需要任何编程工作就可以得到，如图1-1所示，它有控制菜单图标、标题栏、最小化、最大化及关闭按钮，有边框和客户区。在设计定制窗体的时候，客户区内还可以有一些网格，以便于对齐按钮和边框，设计美观的程序用户接口。

窗体可以移动位置、缩放大小和覆盖其它窗体，所有这些操作都可以通过键盘、鼠标和控制菜单来实现。

关于窗体的一般属性，前面已经涉及。这里介绍一些窗体特有的属性，主要有MinButton和MaxButton、Control Box、Icon及BorderStyle几个。

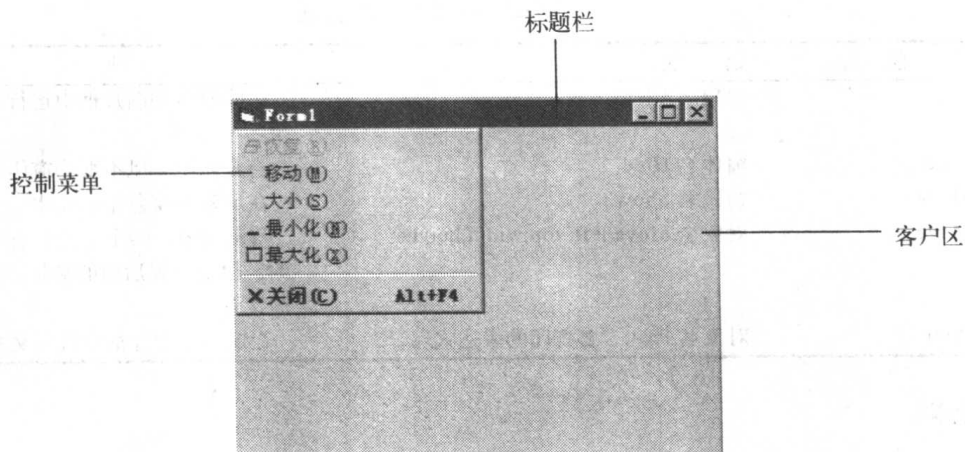


图1-1 窗体

3. MinButton和MaxButton属性

这两个属性只能在窗体设计阶段设置，在运行阶段是只读的。MinButton和MaxButton属性默认为True；若设为False，则隐藏标题栏上的对应最小化最大化按钮。

4. Control Box属性

Control Box(控制框)属性用于设置运行时期的表单是否有控制框。该属性设为True，则在程序运行时，标题栏右边会出现最小化、最大化及关闭按钮(如果其它设置为默认值的话)，左边有控制菜单图标。单击该图标，会出现窗口常有操作的菜单，如移动、放大、缩小、关闭等。Control Box设为False，则隐藏标题条上的所有按钮和控制菜单，无法利用它们进行操作，不能关闭程序。

5. Icon属性

Icon(图标)属性用于设置窗体最小化后的图标。

在设计阶段，打开属性窗口的Icon选项，就会有弹出式菜单让你选择图表(扩展名为.ICO)。在运行阶段，可用LoadPicture函数给窗体指定一个图标：

```
Form1.Icon=LoadPicture(图标所在的路径和图标名)
```

6. BorderStyle属性

BorderStyle属性不仅确定窗体边框的样式，也确定窗体的外观。BorderStyle属性一共有六个选项：

- 1) 没有边界，也没有按钮；
- 2) 固定大小的单线框，有标题栏、最小化、最大化及关闭按钮；
- 3) 可调整大小和位置。这是默认值；
- 4) 固定大小的对话框；
- 5) 样子像工具条，有关闭按钮，无法调整大小；
- 6) 同上，但可以调整大小。

注意 只有BorderStyle属性设置为1、2或3时，才能设置窗体的Icon属性及MinButton和MaxButton属性。

1.2.4 控件

控件是用户操作对象的图形表示,用于向应用程序提供控制等信息。例如,按钮、标注、编辑框、计时器等。注意,有的控件没有图形表示,如计时器。控件是窗体中用于和用户交互的非窗口形式的对象。控件是一个工程的基础。像命令按钮和复选框这样的对象,使得Windows应用程序相对非Windows应用程序用起来简单一些。

控件的数量很多,用户甚至还可以创建自己的控件。Visual Basic的控件大致可以分为三类:

- 内部控件;
- ActiveX控件;
- 可插入的对象(也叫插件)。

1. 内部控件

内部控件都在Visual Basic的.exe文件中,例如CommandButton和Frame控件。内部控件总是出现在工具箱中,不像ActiveX控件和可插入对象那样可以添加到工具箱中,或从工具箱中删除。

2. ActiveX控件

ActiveX控件是扩展名为.ocx的独立文件,其中包括各种版本Visual Basic提供的控件(如DataGrid控件和DataCombo控件)和仅在专业版和企业版中提供的控件(例如ListView控件、Toolbar控件和Animation控件),另外还有许多第三方提供的ActiveX控件。

旧版的ActiveX控件具有文件扩展名.vbx,在Visual Basic的早期版本编写的应用程序中可以找到这些控件。当Visual Basic 6打开包含.vbx控件的项目时,在默认情况下用.ocx控件取代.vbx控件。

3. 可插入对象

可插入对象指能添加到工具箱中的对象,可以把这些对象当作控件使用。Visual Basic的可插入对象很多,包括BMP图像、Word文档、声音文件等。

单击项目菜单的部件命令(Project/Components)可以弹出一个菜单,里面包括ActiveX控件和可插入对象的选项卡。通过选项卡,可以将它们添加到工具箱中,或者从工具箱中删除。工具箱是用来管理控件的工具。可以使用工具箱画出任何所需的控件,具体内容以后再详细介绍。

Visual Basic提供大量的控件,并且随着Visual Basic版本的升级,其控件的数量不断扩展。这意味着Visual Basic功能的增强。由于适用范围、场合的不同,控件的种类也不相同,不同控件的属性和方法都有不一样的地方。至于一些常见的属性,如名字、标题、尺寸等,前面在介绍对象的属性和方法时已经提到过。

前面介绍了Visual Basic中最重要的两种对象:窗体和控件。此外,Visual Basic 6还有一些应用程序环境的系统资源对象,比如剪贴板对象、屏幕对象和打印机对象等等。一般对象都包含三大要素:属性、方法和事件。对象是一个抽象化的概念,只有在设置了属性之后,才能生成真正的对象。所有的Visual Basic对象都具备各自的属性集合,以决定它的外观和行为。同时,对象还必须通过内部的一段代码(方法)完成某一特定的任务。方法的内容是不可见的,我们只能使用它,当我们控制某个对象时,实际上是执行了对象内部的某段代码。