

流行软件丛书



浩强创作室

HAO QIANG STUDIO

谭浩强 主编

Visual Basic 6.0 中文版提高与应用

高峰霞 廖彬山 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

URL: <http://www.phei.com.cn>



流行软件丛书

谭浩强 主编

Visual Basic 6.0 中文版提高与应用

高峰霞 廖彬山 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Visual Basic 是美国微软公司推出的使用极为广泛的可视化开发环境。本书从实用角度出发,对 Visual Basic 的编程技术和应用技巧作了深入介绍。全书在介绍 Visual Basic 的各种功能时,结合国内用户的实际情况和可能遇到的重点、难点问题,有针对性地列举了大量的编程实例,为读者营造了一个轻松学习和掌握 Visual Basic 的环境。

本书是一本实用性很强的普及性读物。全书内容新颖,语言浅显、通俗易懂,适合于在计算机编程方面希望进一步深入和提高自己的读者使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,翻版必究。

图书在版编目(CIP)数据

Visual Basic 6.0 中文版提高与应用/高峰霞,廖彬山编著. - 北京:电子工业出版社,1999.11

(流行软件系列)

ISBN 7-5053-5685-2

I. V… II. ①高… ②廖…

中国版本图书馆 CIP 数据核字(1999)第 33585 号

丛 书 名: 流行软件丛书

主 编: 谭浩强

书 名: Visual Basic 6.0 中文版提高与应用

编 著 者: 高峰霞 廖彬山

责任编辑: 詹善琼

排版制作: 电子工业出版社计算机排版室

印 刷 者: 北京牛山世兴印刷厂

装 订 者: 三河市路通装订厂

出版发行: 电子工业出版社 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 20.75 字数: 531 千字

版 次: 1999 年 11 月第 1 版 1999 年 11 月第 1 次印刷

书 号: ISBN 7-5053-5685-2
TP·2923

印 数: 6000 册 定 价: 26.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换。

若书店售缺,请与本社发行部联系调换。电话 68279077

序

人类正在准备进入 21 世纪,在新世纪中,计算机将成为人们广泛使用的现代工具。每一个有文化的人都应当学习计算机知识,掌握计算机使用技能,有效地利用计算机去处理面临的任务。人们已经认识到:计算机应用知识已成为当代知识分子知识结构和能力结构中不可缺少的重要组成部分。我们的口号是:把计算机从少数计算机专家手中解放出来,使之成为广大群众手中的工具。

广大群众迫切要求学习计算机知识与应用,那么怎样学习才能取得较好的效果呢?根据 20 年来的经验可以得出以下的共识:

1. 计算机的应用是分层次的,同样,计算机教育也是分层次的,不应该用一个统一的模式去套不同的对象。计算机教育可以分为三个层次:(1)计算机入门教育;(2)计算机技术教育;(3)计算机专业教育。对不同的人应当有不同的要求。

2. 应当区分计算机研制人员和计算机应用人员,区分计算机专业和非计算机专业,区分学校教育和面向社会的教育。对社会上大多数人来说,计算机不是一个专业、不是一种理论课程,而是一种工具。因此应当强调以应用为目的,以应用为出发点。

3. 学习内容的选择不应从理论入手,而应当从实际入手,需要什么就学什么。不同的人有不同的学习内容。提倡学以致用、急用先学、立竿见影。不必过分强调学科的系统性和完整性。

4. 当前已有大量优秀的应用软件可供选用,这为广大计算机的应用人员和初学者提供很大的方便。现在使用计算机的人不必事事都要自己去编写程序,而只要选择合用的应用软件就能顺利解决一般任务。因此学习有关软件的使用方法就成为许多人学习计算机的第一步。当然,对大学生和中级以上计算机应用人员,学习程序设计知识是有必要的。

5. 在计算机的应用中,存在一种“二八现象”,即计算机软件的功能中 20% 足可供 80% 的人(初学者和初级应用人员)使用,而其他 80% 的功能只是为 20% 的人(高级应用人员和专业人员)设计的。因此开始学习时决不能面面俱到,贪多求全,先把最基本的 20% 学好用好。在有一定使用基础之后,再逐步掌握其余 80% 的功能。

6. 学习应用软件,应当提倡熟练掌握其使用技能,并用于实践。学习好坏的标准不是“知道不知道”,而是“会不会用”。在学习方法上,对初学者应提倡:由实际到理论,由具体到抽象,由个别到一般,由零碎到系统。

根据广大计算机初学者和计算机应用人员的要求,按照以上认识,我们组织了这套“流行软件丛书”,介绍最新、最常用的软件的使用方法。本丛书分两个层次:第一层次为“使用导引”,介绍有关软件最基本、最常用的应用知识;第二层次为“提高与应用”,介绍比较深入的应用知识和技巧,供不同基础的读者阅读。应当说明:本丛书着眼于应用,而不是理论教材。最好一边阅读,一边上机,以加深理解。

本丛书由浩强创作室组织编写。参加本丛书策划、组织和编写的有:谭浩强、朱桂兰、薛淑斌、王电、杨一平、周山芙、王耆、王启智、于双河、梁洵、廖彬山、高峰霞、赵野军、钱国梁、赵英

良、鲁鹏、宋旭明、谭亦峰等。电子工业出版社对本丛书的编写工作给予大力支持,在此表示感谢。

由于计算机技术发展很快,许多新软件问世不久,我们自己也处在不断学习和实践过程中,书的内容可能不能完全满足大家的要求,甚至会有某些错误,敬请批评指正。

谭浩强
1999年5月

前 言

Visual Basic 是美国微软公司推出的对市场冲击最大、最优秀的可视化开发环境,它采用一种非常巧妙的方法将 Windows 的编程复杂性封装起来,使编程人员轻松容易地步入编写 Windows 程序的殿堂。

自从 1991 年 Visual Basic 1.0 推出以来,Visual Basic 版本不断得到更新,功能不断得到增强,现在的最新版本是 Visual Basic 6.0。为了满足不同的开发需要,Visual Basic 6.0 本身又分为学习版、专业版和企业版。Visual Basic 6.0 对已有功能作了完善,同时增加了许多新的功能。

新的数据访问特性

Visual Basic 6.0 在数据访问技术方面比 Visual Basic 5.0 有了极大的增强:

- ◆ **新的数据访问接口 ADO:** Visual Basic 6.0 支持的数据访问接口有 ADO(ActiveX 数据对象)、RDO(远程数据对象)和 DAO(数据访问对象),每个接口都代表了数据访问技术的不同发展阶段,最新的接口是 ADO。ADO 是比 RDO 和 DAO 更加简单、更加灵活的对象模型,它是为 OLE DB 而设计的。OLE DB 是新的低层 COM 接口,可以为任何数据源提供高性能的访问,这些数据源包括关系和非关系数据库、电子邮件和文件系统、文本和图形、自定义业务对象等。ADO 封装并实现了 OLE DB 的所有功能,它将全面取代 DAO 和 RDO,成为数据访问接口的主流。
- ◆ **新的数据源:**除了已有的 Data 和 Remote Data 控件外,Visual Basic 6.0 引入了几种新的数据源,包括数据识别的类模块和用户控件、数据环境、ADO 记录集和 ADO Data 控件等。ADO Data 控件的功能与 Data 和 Remote Data 控件十分相似,都允许编程人员用最少的代码来创建数据库应用程序。
- ◆ **新的数据绑定控件:**Visual Basic 6.0 新增加了为数不少的数据绑定控件,包括 DataGrid、DataList、DataCombo 和 MSHFlexGrid 等。
- ◆ **集成的可视化数据库工具:**包括查询设计器和数据库设计器,提供了可视地查看并操作表、视图、存储过程和数据库结构的功能。
- ◆ **数据环境设计器:**允许编程人员可视化地创建和操作 ADO 对象,为编程人员操纵数据源提供了极大的方便。
- ◆ **数据报表设计器:**允许编程人员利用拖放技术来快速地从任何数据源创建报表。

新的 Internet 特性

Visual Basic 6.0 提供了多种不同的途径来帮助编程人员开发出动态的、功能强大的 Internet 应用程序:

- ◆ **IIS 应用程序:**可以直接使用 Visual Basic 6.0 来创建 IIS 应用程序,响应浏览器用户的请求。
- ◆ **DHTML 应用程序:**可以直接通过 Visual Basic 6.0 来创建 DHTML 应用程序。DHTML 应用程序直接在客户端处理 HTML 页面上的操作,而不用把处理传给服务器,大大减轻了服务器和网络的负担。

◆ **对下载 ActiveX 文档的支持:** 在 Internet Explorer 中下载 ActiveX 文档与下载 ActiveX 控件的方法相同。

除了新的数据访问特性和新的 Internet 特性外, Visual Basic 6.0 在其余方面也都作了重大改进, 包括新的或增强的控件, 新的或增强的语言特性, 以及新的或增强的向导, 所有这些都使得 Visual Basic 6.0 功能更为强大。总之, 无论是开发小型应用系统, 还是大型企业应用系统, 甚至开发功能强大的 Internet 应用程序, Visual Basic 6.0 都能胜任。

由于 Visual Basic 功能异常丰富, 概念十分庞杂, 如果在一本书中对全部内容逐一进行介绍, 不但会使篇幅过于庞大, 而且会使读者感到如入云雾之中, 抓不着重点, 也入不了门。因此, 我们设想将 Visual Basic 的学习分为两个阶段, 第一阶段是入门, 第二阶段是提高与应用。针对不同阶段的学习需要, 我们相应地编写了以下两本书:

◆ **《Visual Basic 6.0 中文版快速入门》**

◆ **《Visual Basic 6.0 中文版提高与应用》**

本书是第二本。第一本书已经介绍了 Visual Basic 最基本、最常用的功能, 起到了引导读者快速入门的作用。然而, 以我们的经验来看, 对于 Visual Basic 这样的软件, 不学到一定深度和广度是很难在实际编程中得以应用自如的。因此, 我们在第一本书的基础上编写了本书, 目的在于帮助读者进一步深入和提高。全书以大量典型的应用实例, 教会读者更全面、更系统地掌握 Visual Basic 的编程技术和应用技巧。

本书由浩强工作室策划, 高峰霞和廖彬山负责编写, 高越明仔细审阅了全书。参与本书部分编写工作的还有高月华、谢华英、邵光伟、高星泉、林利和章少峰。本书的录排工作由郭静负责。

由于时间仓促, 加之水平有限, 不当之处在所难免, 欢迎广大读者批评指正。

编 者
1999 年 5 月

目 录

第 1 章 对象编程	(1)
1.1 对象和类	(1)
1.2 对象变量	(1)
1.2.1 声明对象变量	(2)
1.2.2 赋值对象变量	(3)
1.2.3 特定对象变量和通用对象变量	(3)
1.2.4 释放对象引用	(5)
1.2.5 检查两个对象变量的引用是否相同	(5)
1.2.6 找出对象所属的类	(5)
1.2.7 CallByName 函数	(6)
1.2.8 Me 关键字	(6)
1.2.9 使用 With 语句	(6)
1.2.10 给过程传递对象	(7)
1.2.11 缺省属性	(7)
1.2.12 容器对象	(8)
1.2.13 系统对象	(9)
1.3 集合	(9)
1.3.1 创建自己的集合	(10)
1.3.2 基于 0 和基于 1 的集合 L	(11)
1.3.3 For Each 循环	(12)
1.4 类的创建	(12)
1.4.1 创建一个类模块 L	(12)
1.4.2 向类中添加属性	(13)
1.4.3 向类中添加方法	(14)
1.4.4 类模块和标准模块的比较	(14)
1.4.5 编程实例	(15)
1.4.6 往类中添加事件	(20)
1.5 对象浏览器	(23)
1.5.1 查找和浏览对象	(24)
1.5.2 添加对象的描述	(25)
1.5.3 查看代码	(26)
1.5.4 浏览其他应用程序的对象	(26)
1.6 FSO(文件系统对象)	(26)
1.6.1 FSO 包含的对象	(26)
1.6.2 Scripting 类型库	(27)
1.6.3 用 FSO 编程	(28)
1.6.4 获得驱动器的信息	(28)

1.6.5 处理文件夹	(30)
1.6.6 处理文件	(30)
第2章 界面设计	(34)
2.1 界面样式	(34)
2.1.1 SDI 界面	(34)
2.1.2 MDI 界面	(35)
2.1.3 资源管理器样式的界面	(37)
2.2 管理注册表	(37)
2.2.1 保存或创建注册表设置值	(38)
2.2.2 删除注册表设置值	(38)
2.2.3 检索注册表设置值	(39)
2.3 通用对话框	(40)
2.3.1 “打开”对话框	(42)
2.3.2 “另存为”对话框	(43)
2.3.3 “字体”对话框	(43)
2.3.4 “颜色”对话框	(44)
2.3.5 “打印”对话框	(45)
2.4 启动窗体和 Sub Main 过程	(46)
2.4.1 启动窗体	(46)
2.4.2 Sub Main 过程	(47)
2.4.3 展示屏幕	(48)
2.5 MDI 编程实例	(48)
2.5.1 创建 MDI 窗体和子窗体	(49)
2.5.2 设计菜单	(50)
2.5.3 使“文件”菜单动态增长	(52)
2.5.4 为 MDI 窗体创建工具栏	(54)
2.5.5 程序清单	(55)
2.5.6 几个补充说明	(70)
2.5.7 “窗口”菜单	(71)
2.6 窗体的生存期	(74)
2.6.1 创建但不加载	(74)
2.6.2 加载但不显示	(74)
2.6.3 显示状态	(75)
2.6.4 卸载并收回窗体所占的内存和资源	(75)
2.6.5 窗体处于卸载但未被引用	(76)
2.7 应用程序向导	(76)
2.8 快捷菜单	(77)
第3章 图形和文本处理	(80)
3.1 图形基础	(80)
3.1.1 AutoRedraw 属性与持久性	(80)

3.1.2 ClipControls 属性和裁剪区域	(81)
3.1.3 图形层次	(81)
3.2 坐标系和刻度	(82)
3.2.1 标准刻度	(82)
3.2.2 自定义刻度	(82)
3.2.3 设置当前坐标	(83)
3.3 图形方法和 Print 方法	(83)
3.4 颜色设置	(84)
3.4.1 RGB 函数	(84)
3.4.2 QBColor 函数	(84)
3.4.3 内部颜色常数	(85)
3.4.4 直接使用颜色的十六进制值	(86)
3.5 Picture 对象	(86)
3.6 字体	(87)
3.6.1 检查可用的字体	(87)
3.6.2 设置字体属性	(87)
3.7 TextHeight 和 TextWidth 方法	(88)
3.8 格式化数字、日期和时间	(88)
3.8.1 格式化数字	(88)
3.8.2 格式化日期和时间	(89)
3.8.3 格式化文本	(89)
3.8.4 命名格式	(89)
3.9 Clipboard 对象	(90)
3.10 打印	(91)
3.10.1 使用 PrintForm 方法	(91)
3.10.2 Printer 对象	(91)
第 4 章 ActiveX 控制的使用	(93)
4.1 Visual Basic 中的 ActiveX 控件	(93)
4.2 在工程中加载 ActiveX 控件	(95)
4.3 RichTextBox 控件	(97)
4.4 ImageList 控件	(99)
4.5 Toolbar 控件	(101)
4.5.1 编程实例	(101)
4.5.2 ToolBar 控件的新增功能	(121)
4.6 StatusBar 控件	(122)
4.7 ProgressBar 控件	(126)
4.8 Slider 控件	(127)
4.9 Multimedai 控件	(128)
4.9.1 多媒体设备类型	(128)
4.9.2 MCI 命令	(128)

4.9.3 常用属性	(129)
4.9.4 编制一个“媒体播放器”	(130)
4.10 Animation 控件	(135)
4.11 ImageCombo 控件	(136)
4.12 MonthView 控件	(138)
4.13 Internet Transfer 控件	(141)
4.14 WebBrowser 控件	(144)
4.15 Winsock 控件	(148)
4.15.1 Winsock 控件的属性、方法和事件	(149)
4.15.2 编写 TCP 应用程序	(150)
4.15.3 编写 UDP 应用程序	(154)
第5章 拖放和 OLE 拖放	(157)
5.1 控件拖放	(157)
5.2 OLE 拖放	(160)
5.2.1 OLEDragMode 属性	(161)
5.2.2 OLEDropMode 属性	(161)
5.2.3 DataObject 对象	(162)
5.2.4 OLEDrag 方法	(162)
5.2.5 拖放事件	(163)
5.2.6 编程实例	(165)
第6章 调试与错误处理	(167)
6.1 程序中的错误	(167)
6.2 程序调试	(169)
6.2.1 工作模式	(169)
6.2.2 调试实例	(169)
6.2.3 断点	(171)
6.2.4 Stop 语句	(173)
6.2.5 逐语句和逐过程执行	(173)
6.2.6 本地窗口	(174)
6.2.7 监视表达式与监视窗口	(175)
6.2.8 调用堆栈	(177)
6.2.9 立即窗口	(177)
6.2.10 用断言检查代码	(178)
6.3 错误捕获及处理	(178)
6.3.1 如何处理错误	(179)
6.3.2 On Error Resume Next 语句	(180)
6.3.3 关闭错误处理例程	(181)
6.4 条件编译	(181)
6.4.1 条件编译语句	(181)
6.4.2 定义条件编译常数	(182)

第 7 章 数据访问编程	(184)
7.1 数据库基础	(184)
7.1.1 表、记录和字段	(184)
7.1.2 数据库	(184)
7.1.3 索引	(185)
7.1.4 关系	(185)
7.1.5 结构化查询语言	(185)
7.1.6 可视化数据管理器	(188)
7.2 数据访问对象和 Data 控件	(191)
7.2.1 数据访问对象	(191)
7.2.2 使用 Data 控件	(192)
7.2.3 数据绑定控件	(194)
7.2.4 记录集	(195)
7.2.5 Data 控件与 DAO 结合使用	(195)
7.2.6 自动添加、更新和删除记录	(196)
7.2.7 记录集的排序与筛选	(197)
7.2.8 定位记录集中的记录	(197)
7.2.9 从记录中获取数据	(200)
7.2.10 用代码更新记录集	(201)
7.2.11 Error 事件	(204)
7.2.12 事务处理	(204)
7.3 远程数据对象和 Remote Data 控件	(205)
7.3.1 远程数据对象	(205)
7.3.2 RDO 与 DAO 的比较	(207)
7.3.3 Remote Data 控件	(208)
7.4 ActiveX 数据对象和 ADO Data 控件	(209)
7.4.1 OLE DB	(209)
7.4.2 ActiveX 数据对象	(210)
7.4.3 ADO Data 控件	(212)
7.4.4 ADO Data 控件支持的事件	(217)
7.4.5 数据绑定控件	(218)
7.4.6 DataList 和 DataCombo 控件	(218)
7.4.7 DataGrid 控件	(221)
7.5 数据环境设计器	(225)
7.5.1 添加一个 DataEnvironment 对象	(226)
7.5.2 Connection 对象	(226)
7.5.3 “数据视图”窗口	(227)
7.5.4 Command 对象	(228)
7.5.5 数据环境如何在应用程序中使用	(229)
7.5.6 字段映射	(232)

7.5.7 查询设计器	(234)
7.5.8 参数查询	(236)
7.5.9 Command 层次结构	(237)
7.5.10 MSHFlexGrid 控件	(243)
7.6 数据报表设计器	(244)
7.6.1 数据报表设计器的结构	(244)
7.6.2 创建一个简单的数据报表	(246)
7.6.3 添加日期、时间、页号和标题	(247)
7.6.4 打印数据报表	(247)
7.7 数据访问总结	(248)
第 8 章 ActiveX 部件的使用	(250)
8.1 ActiveX 部件类型	(250)
8.2 使用 ActiveX 部件	(250)
8.2.1 添加对部件的引用	(251)
8.2.2 声明对象变量	(252)
8.2.3 将对象引用赋予变量	(252)
8.2.4 使用对象的属性和方法	(253)
8.2.5 响应对象的事件	(254)
8.2.6 释放对象	(254)
8.2.7 部件对象的层次结构	(255)
8.2.8 编程范例	(256)
8.3 插入 OLE 对象	(262)
8.3.1 链接对象与嵌入对象	(262)
8.3.2 设计时创建 OLE 对象	(262)
8.3.3 “选择性粘贴”对话框	(264)
8.3.4 运行时创建 OLE 对象	(264)
8.3.5 InsertObjDlg 与 PasteSpecialDlg 方法	(265)
8.3.6 激活 OLE 对象	(265)
8.3.7 在位激活	(266)
8.3.8 将数据库与 OLE 容器控件绑定	(266)
8.3.9 ObjectMove 事件	(266)
8.3.10 确定对象如何显示	(267)
8.3.11 保存和检索嵌入对象中的数据	(267)
8.3.12 Updated 事件	(267)
8.3.13 其他有关属性	(268)
8.3.14 协调菜单与工具栏	(268)
8.3.15 使用工具箱嵌入 OLE 对象	(269)
第 9 章 ActiveX 部件的创建	(270)
9.1 创建 ActiveX 控件	(270)
9.1.1 基本概念	(270)

9.1.2	制作一个简单的 ActiveX 控件	(271)
9.1.3	UserControl 对象的关键事件	(280)
9.1.4	扩展属性	(280)
9.1.5	环境属性	(281)
9.2	ActiveX 文档	(281)
9.2.1	创建简单的 ActiveX 文档	(282)
9.2.2	ActiveX 文档的组成部分	(285)
9.2.3	为 UserDocument 对象添加属性	(285)
9.2.4	UserDocument 对象的关键事件	(287)
9.2.5	ActiveX 文档的用户界面	(288)
第 10 章	Internet 编程	(291)
10.1	Internet 基础	(291)
10.2	Internet 开发历史	(294)
10.3	Visual Basic 的 Internet 技术	(295)
10.4	DHTML 应用程序	(295)
10.4.1	系统要求	(295)
10.4.2	创建一个 DHTML 工程	(296)
10.4.3	页面设计	(297)
10.4.4	动态 HTML 的对象模型	(301)
10.4.5	动态 HTML 中的事件	(301)
10.4.6	使页面元素可编程	(303)
10.4.7	元素的缺省操作	(303)
10.4.8	设置和获取属性	(304)
10.4.9	文本的替换处理	(305)
10.4.10	漫游	(305)
10.5	IIS 应用程序	(305)
10.5.1	系统要求	(306)
10.5.2	WebClass	(306)
10.5.3	开始 IIS 工程	(307)
10.5.4	添加模板 WebItem 到 WebClass	(308)
10.5.5	WebClass 事件	(309)
10.5.6	WebClass 使用的 ASP 对象	(312)
10.5.7	Start 事件	(313)
10.5.8	发送 HTML 到浏览器	(313)

第1章 对象编程

Visual Basic中,几乎所有的编程活动都是围绕着对象展开的。对象无处不在,它是Visual Basic编程的核心。编程时可以使用Visual Basic提供的对象(如窗体和控件等),也可以从应用程序内部去访问其他应用程序提供的对象(如Excel工作表对象和Word文档对象等),甚至可以创建自己的对象。

《Visual Basic 6.0中文版快速入门》一书已经初步介绍了对象的有关知识,本章将进一步讨论对象。

1.1 对象和类

1. 对象

对象这一概念并不神秘,它来源于现实生活中。现实生活中,我们时时刻刻都在和对象打交道,我们骑的自行车、用的计算机和电话等都是对象。Visual Basic中,对象的概念是现实生活中对象的模型化,它是代码和数据的组合,可以作为一个整体来处理,如窗体、控件和数据库等都是对象。对象可以包含其他对象,如数据库作为一个对象,还可以包含字段和索引等对象。

Visual Basic中,每个对象都有自己的特征和行为。对象的特征用数据来表示,称为对象的属性;对象的行为用对象中的代码来实现,称为对象的方法。此外,每个对象还可以识别和响应某些预定义的动作,这些动作称为事件,如单击鼠标、移动鼠标、按下键盘上的某个键时都会产生事件。如果说属性决定了对象的特征,方法决定了对象的行为,那么事件就决定了对象之间相互联系的方式。

2. 类

类是对象的归纳和抽象。在Visual Basic中,所有对象都是由类定义的。例如,工具箱中的控件代表类(即控件类),窗体上画出的具体控件则是控件类的实例,这个类实例就是应用程序中引用的对象。

对象具有的属性、方法和事件都是在类中定义的。类就像是一个蓝图或模板,确定由它生成的对象所具有的公共特征和功能。例如,在某个窗体上画了三个文本框,那么每个文本框都是文本框类的一个实例,都具有文本框类定义的公共特征和功能。

1.2 对象变量

创建对象最简单的办法就是利用工具箱在窗体上画出各种各样的控件。但是,如果想充分体现对象的优势,那么必须使用对象变量。例如,开始一个新的工程,再在窗体中装入一幅图片并画一个命令按钮,然后将窗体的Name和Caption属性分别设为“frmSample”和“克

克隆窗体”，将命令按钮的Name和Caption属性分别设为“cmdNew”和“克隆”(图1.1的左边)，最后为命令按钮编写以下的Click事件过程：

```
Private Sub cmdNew_Click()  
    Dim objForm As New frmSample  
    objForm.Show  
End Sub
```

这样，运行时用户每单击一次命令按钮，就会“克隆”出一个新的完全一样的窗体(图1.1的右边)。

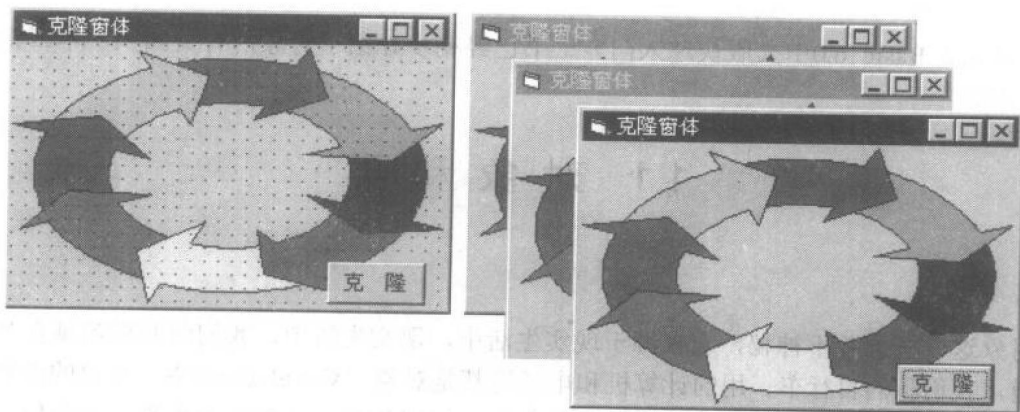


图 1.1 “克隆”出新的窗体

1.2.1 声明对象变量

对象变量用于在程序代码中引用对象。与其他普通变量一样，对象变量使用之前要进行声明，声明方法是：

```
{Dim|ReDim|Static|Private|Public} variable As [New] class
```

Dim、ReDim、Static、Private和Public是关键字，含义类似于普通变量。class表示对象所属的类型。例如，可以声明一个对象变量anyForm来引用应用程序中的任何窗体：

```
Dim anyForm As Form
```

又如，可以声明一个对象变量anyText来引用应用程序中的任何文本框：

```
Dim anyText As TextBox
```

New是可选关键字，用于在运行时创建新对象，该对象被视为它的类所定义的对象。当使用一个已经存在的窗体作为一个类时，就可以在运行时用New关键字来创建一个新的窗体实例。例如，前面“克隆”出新窗体的例子中，就声明了一个对象变量objForm来引用应用程序中名为“frmSample”的窗体。

***注意：**还可以用New关键字来创建集合的实例以及类模块所定义类的实例。但是，对于基本数据类型、通用的控件类型、特定的控件类型以及特定的控件对象，则不能使用New关键字。例如，以下语句都是错误的：

```
Dim X As New Integer '基本数据类型Integer不能用
```



```
Dim X As New Control '通用的控件类型Control不能用
Dim X As New ListBox '特定的控件类型ListBox不能用
Dim X As New txtName '特定的控件对象txtName不能用
```

1.2.2 赋值对象变量

当要求一个对象变量指向应用程序中的某一具体对象时，必须用Set语句将对象赋给对象变量，语法是：

```
Set variable = object
```

其中，variable是对象变量，而object是具体的对象。例如：

```
Dim anyText As TextBox '声明一个引用文本框的对象变量anyText
Set anyText = txtName '将具体的文本框txtName赋给anyText
```

***注意：** New关键字可以和Set语句并用来创建新的实例，这样可以加快运行速度。例如：

```
Dim objForm As frmSample
Set objForm = New frmSample
```

1.2.3 特定对象变量和通用对象变量

1. 特定对象变量

特定对象变量只能引用应用程序中特定类型的对象，如特定的窗体变量只能引用应用程序中的某一已存在的窗体，而特定的控件变量只能引用应用程序中特定类型的控件(如文本框、列表框或命令按钮等)。例如，开始一个新的工程，再在窗体中画一个命令按钮(图1.2的左边)，然后编写窗体的Click事件过程：

```
Private Sub Form_Click()
    Dim objCmd As CommandButton '声明一个引用命令按钮的对象变量objCmd
    Set objCmd = Command1 '将具体的命令按钮Command1赋给objCmd
    objCmd.Caption = "克隆" '改变命令按钮的标题
End Sub
```

这样，运行时单击窗体，将使命令按钮的标题改为“克隆”(图1.2的右边)。

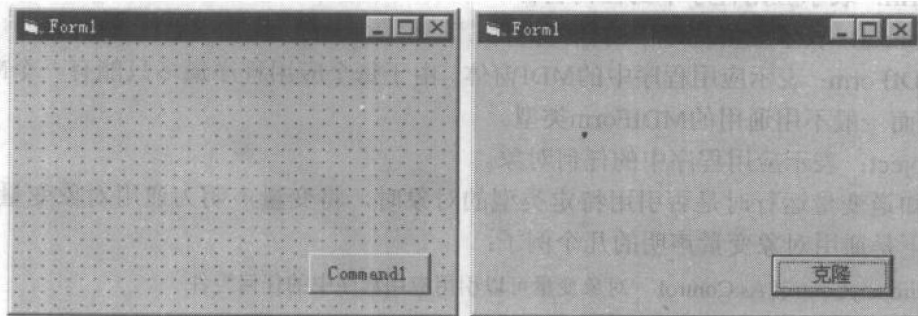


图 1.2 改变命令按钮的标题