



Designed for

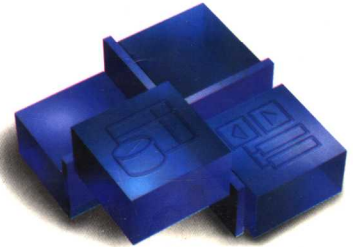
Microsoft®
Windows NT®
Windows 98



Learn Microsoft
Visual Basic 6.0 Now

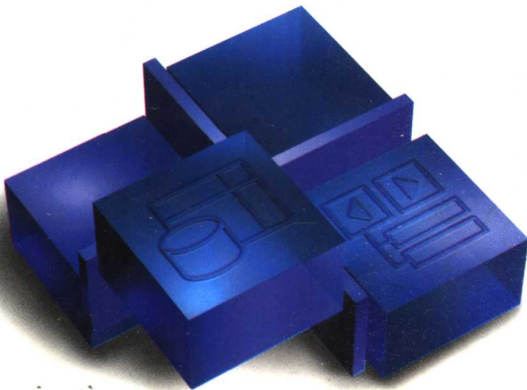
微软公司
核心技术书库

(美) Michael Halvorson 著
张钟军 沈雪梅 强锐 译



Visual Basic 6

学习指南



机械工业出版社
China Machine Press

Microsoft Press

Visual Basic 6学习指南

(美) Michael Halvorson 著

张钟军 沈雪梅 强 锐 译



机械工业出版社
China Machine Press

本书提供了完整的Visual Basic 6工作模型,您可从中学习Visual Basic 6的特性。本书从熟悉Visual Basic 6编程环境入手,循序渐进,详细介绍了如何编写Visual Basic语句,如何向用户界面中添加打印机支持、错误句柄、艺术作品和动画,如何管理不同工作组和团体中的数据以及包括ActiveX、Internet编程基础和高级数据库管理等内容的高级编程技术和工具。

本书内容精练,通俗易懂,无论对初学者还是有一定经验的编程人员,都有很高的参考价值。

Michael Halvorson: Learn Microsoft Visual Basic 6.0 Now.

Copyright © 2000 by Microsoft Corporation.

Original English language edition Copyright © 1999 by Michael Harvorson and Microsoft Corporation.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A.

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

本书中文简体字版由微软出版社授权机械工业出版社出版, 未经出版者书面许可, 本书的任何部分不得以任何方式复制或抄袭。

版权所有, 翻印必究。

本书版权登记号: 图字: 01-1999-2606

图书在版编目(CIP)数据

Visual Basic 6学习指南/(美)哈文森(Halvorson, M.)著;张钟军等译.-北京:机械工业出版社, 1999.12

(微软公司核心技术书库)

书名原文: Learn Microsoft Visual Basic 6.0 Now

ISBN 7-111-07650-8

I. V… II. ①哈… ②张… III. BASIC语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字(1999)第65281号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 郭东青

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

1999年12月第1版·2000年4月第2次印刷

787mm×1092mm 1/16·17.25印张

印数: 6 001-9 000册

定价: 42.00元(附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前 言

Microsoft Visual Basic是当今最流行的编程语言之一。到1999年初，Visual Basic软件开发团体已经超过了350万人，而且这个数字还在不断增长。学生、管理人员和技术人员使用Visual Basic学习编写实用的Windows程序；商业人士使用Visual Basic在他们的Microsoft Office应用程序中编写宏；有经验的软件开发人员使用Visual Basic编写功能强大的商用程序以及协同生产工具。

不管属于上述情况中的哪一种，还只是对Visual Basic编程感到好奇，这本供自学的教程和配套软件包组成一个学习系统。包含了Microsoft Visual Basic 6的工作模型（Working Model）版本，可以用来编写Microsoft Windows操作系统上实用、有趣的应用程序。

本教程每章介绍一个编程概念，从最基本的开始，通过阶梯练习、实践操作以及深入说明来加强对这些概念的理解。专门准备的光盘中包含了Visual Basic 6开发系统的工作模型，书中所有的示例程序和练习文件，使得编程更加容易入手。在前言的“安装软件”部分，你将发现把软件和练习文件拷贝到计算机硬盘中的指令。

注意 本书共分19章，取材于畅销书Microsoft Visual Basic Professional 6 Step by Step(Microsoft出版社，1998出版)，以及一张CD-ROM，其中包含了具备Visual Basic 6专业版特性的Visual Basic 6工作模型，你不必购买Visual Basic 6就可使用本书。

你以前编写过程序吗？

本书是为两种人设计的：一种是开始学习Visual Basic的新手，一种是打算从其他编程环境中，比如早期的BASIC、VBA（Visual Basic for Application）、Microsoft Visual C++、或者Microsoft Visual Java++转移到Visual Basic上的有经验的程序员。本书最大的受益者是那些熟悉Microsoft Windows和基于Windows的应用程序，如Microsoft Word、Microsoft Excel和Microsoft Access应用程序和数据库的用户。在本书中将要创建的图形应用程序和这些产品所提供的程序看起来很相似。如果希望创建应用程序和功能，使Windows 98和Windows程序更加容易使用，那么你就找对门了。

随着对本书的学习，你将通过Visual Basic 6工作模型学到很多Visual Basic的特性。Visual Basic 6工作模型类似于Visual Basic 6专业版，只是它不能创建可执行文件（.exe），而且某些特性无效（如不包含某些ActiveX控件，不能使用DHTML Page Designer、Data Environment Designer或Online Help）。Visual Basic开发组希望你能够喜欢Visual Basic，并升级到Visual Basic的一种完整零售版（参阅下一段）。如果读过这本书后，希望把创建的程序分发给你的朋友、同事或客户，那么请购买Visual Basic。但是在你看到给人印象深刻的工作模型到底能完成什么功能之前，先不要花光手中的钞票。你会发现，工作模型目前已经足以处理很多有用的任务，正如示例程序所演示的那样。

Visual Basic的不同版本

Visual Basic 6是以三种版本销售的：学习版、专业版和企业版。这三种版本的基本特性都一样，因此本书中的练习适用于这三种产品。Visual Basic学习版是最基础的产品。它充分支持在Windows 98和Windows NT下的32位编程。Visual Basic专业版包含了一些附加特性，如创建IIS和DHTML应用程序、数据库编程、创建OLE自动化服务器和增强的数据绑定。专业版还包括了更多的Windows图元文件和位图文件，可以用来给程序添加色彩丰富的艺术效果。Visual Basic企业版是最顶级的客户机/服务器版本。它包含了专业版的所有特性，另外还包括支持团队开发、远程自动化和可视化数据库的元素集成。

Visual Basic还出现在几种Microsoft应用程序中，如Word、Excel、Access、Microsoft PowerPoint和Microsoft Project。因此如果你打算在Microsoft Office应用程序中编写宏，那么在这里受到的训练将帮助你在现在和将来完成卓有成效的工作。

本书的组织结构

本书分成五个主要部分，每一部分又包含相互关联的几章。第一部分介绍Visual Basic的编程环境，教你如何通过创建对象、设置属性、编写程序代码，创建具有一定功能的用户界面。第二部分讨论了如何编写Visual Basic编程语句，着重介绍变量和操作符、判断结构和调试。第三部分讲述了如何向用户界面中添加打印机支持、错误句柄、艺术作品和动画。第四部分讨论了管理各种工作组或团体中数据的几种方法，包括：使用模块和过程共享代码；使用集合和数组管理数据；使用字符串处理技术和加密手段处理文本文件；使用Access数据库；使用自动化调用Microsoft Office程序中的命令。最后，第五部分讲述了高级编程技术和工具，包括“专业的”ActiveX控件、Internet编程基础，以及高级数据库管理。

从阅读前言的下一个主要部分“安装软件”开始，按照指示安装Visual Basic和练习文件。然后依照表1，看看你适合从什么地方开始学起。

表 1

你的情况	学习步骤
对Visual Basic陌生，对编程陌生	一定要阅读前言中的“如何以程序员的思维方式思考”，然后从第1章开始顺序学习
熟悉Visual Basic for Application	学习除第二部分以外的所有章节
熟悉QuickBasic或MS-DOS Qbasic	学习除第二部分外的所有章节
熟悉Visual C++或Visual J++	跳过第一和第二部分中的章节，再学习第三至第五部分中的章节。要获得如何使用Visual Basic创建用户界面的详细信息，请阅读第2~4章、8~9章和第16章

在开始学习之前，让我们来看看本书中使用的术语和约定。

过程约定

- 用编号列表 [1), 2), 3)等] 表示应遵循的步骤。
- 用箭头 (->) 表示应完成的单项任务。

本书的其他特点

- 从屏幕上捕捉的画面显示了范例程序的用户界面和操作完成后的结果。
- 注意标有“提示”、“注意”的备注段落，它们提醒你在继续往下学习之前，应该考虑的其他信息或变通方法。
- 加框的文字介绍了特殊的编程技术、背景知识或者与所讨论内容相关的信息。
- 按照章尾的“如果你想进一步提高”部分中的指令，可以继续探索已经学过的技术。这些程序的功能更为复杂，提供给你一个机会练习已学过的新技术（硬盘C:\LearnVb6\Extras文件夹中列出了这些工程）。

使用练习文件盘安装软件

本书后面所附的CD包含了Visual Basic Working Model软件和学习本书的过程中要用到的练习文件。其中的一些练习文件是部分完成的程序工程，你将通过完善它们来掌握编程的技巧。其他文件是书中工程的完成结果，用于参考。

在下面的练习中，你将在硬盘中安装Working Model和练习文件。如果可能的话，请把练习文件安装在默认的目录（C:\LearnVb6）中，以保持安装结果与每章中使用的路径一致。Visual Basic软件将安装在硬盘C:\Program Files\Microsoft Visual Studio\Vb98文件夹中。

注意 本书中的指令和练习都假设练习文件安装在硬盘中默认的C:\LearnVb6文件夹。如果练习文件不在这个文件夹，在学习书中的练习时，请把练习文件的路径名进行必要的调整。

安装Visual Basic软件和练习文件

按照以下步骤把Working Model和练习文件安装到计算机的硬盘中，以便完成本书的学习。

- 1) 启动Microsoft Windows。
- 2) 把CD-ROM从书中取出来。
- 3) 把CD-ROM插入CD-ROM驱动器中。
- 4) 在屏幕底部的任务条上，单击开始按钮，然后单击运行，出现运行对话框。
- 5) 在运行对话框中，输入d:\vb\setup.exe，然后单击OK。输入时不要有空格（如果CD-ROM的盘符不同，比如是E，请进行替换）。

- 6) 按屏幕的指令进行操作(见图1)。

Visual Basic安装程序出现时，系统建议的选项都被预先选中。如果你想把所有的Visual Basic程序特性拷贝到硬盘上，请选择Custom installation(定制安装)方式，再选中所有的选项。Working Model至少需要50MB的硬盘空间。

提示 如果你想在C:\Program Files\Microsoft Visual Studio\Common\Graphics目录中安装大约3MB的图标、光标、图元文件、位图和视频片断，那么请选中Graphics选项。这些

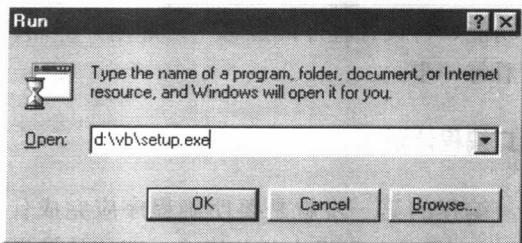


图 1

图形文件使用起来很有趣，本书中将多次用到它们。

安装好Visual Basic后，应该安装本书中的练习文件，请遵循以下步骤：

7) 单击任务栏上的开始按钮，然后单击运行。

8) 在运行对话框中，输入d:\setup，然后单击OK。

9) 按照屏幕的指示操作（请务必按默认方式进行安装，以便把文件放到合适的位置中，见图2）。

10) 两种安装都完成后，把CD-ROM从计算机中取出，放回到书的后面。

如果将来再次运行这两个安装程序，安装程序将进入维护模式，允许你从系统添加或删除单独的元素（或整个安装程序）。

注意 除了安装练习文件外，本书的安装程序还在桌面上创建了一个指向Microsoft出版社万维网站点的快捷方式。如果计算机已建立了到Internet的连接，那么可以双击快捷方式连接到Microsoft出版社的站点。也可以直接连接到<http://mspress.microsoft.com/>。

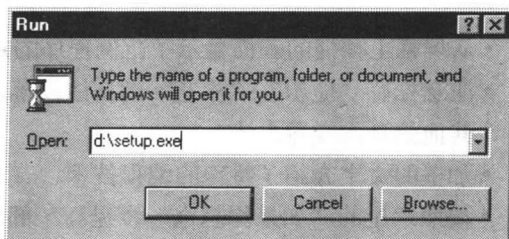


图 2

按程序员的思维方式思考

如果你以前没有编写过程序，也许会想到底什么是程序，以及如何在Visual Basic中创建程序。本节就让我们了解一下程序的概貌。

程序是使计算机完成某项任务的指令集合，比如处理电子艺术作品或管理网络上的文件。程序可以是很小的工具，比如计算家庭债券的程序，也可以是很大的应用程序，如Microsoft Excel或Internet浏览器。所谓Visual Basic程序，就是使用Visual Basic开发系统，专门为Windows或Windows NT编写的应用程序。要编写一个Visual Basic程序，需要确定程序应完成的功能，再设计程序的屏幕——即用户界面，然后使用Visual Basic创建程序。让我们仔细地看看每一步。

制订编程计划

编程的第一步就是要明确程序应完成什么工作。这听起来似乎相当简单（这还不明显吗？），但是如果没有任务描述，即使是最优秀的程序员编程，也会遇到麻烦。制订编程计划就像制订烧烤计划一样。为使烧烤能顺利进行，需要事先准备好。你需要制定菜单、邀请朋友、购买食品、最好还要清扫房屋。如果有朋友正巧登门拜访，并带来烤肉的原料，烧烤也会相当有趣，但是通常情况下，如果程序也像“精神会餐”那样创建，将不会产生太好的结果。

你在学习编程时，应该在坐到计算机面前之前，花些时间思考一下在程序中要解决什么问题。良好的计划可以节约开发时间，也会达到更令人满意的结果。计划的一部分工作可能是创建一份编程步骤的顺序表，或叫做算法。烧烤计划的算法大概如图3所示：

任务:	下星期六进行烧烤
计划:	- 制定菜单 (汉堡包) - 发出邀请 - 清扫房屋 - 修葺草坪
烧烤当日:	- 购买食物 - 点火烤肉 - 准备食物 - 打排球
最后	- 清扫

图 3

设计用户界面

建立了明确的程序目标后，考虑一下程序的外观以及程序如何处理信息是很重要的。程序中所使用的屏幕、图像的集合叫做程序的用户界面。用户界面包括所有的菜单、对话框、按钮、对象和用户在操作程序时看到的图片。Visual Basic开发系统可以让你迅速有效地创建出专业Windows程序中的所有元素。其中的一些元素在下面的Windows程序中可以看到，这个程序是用Visual Basic创建的（见图4）。

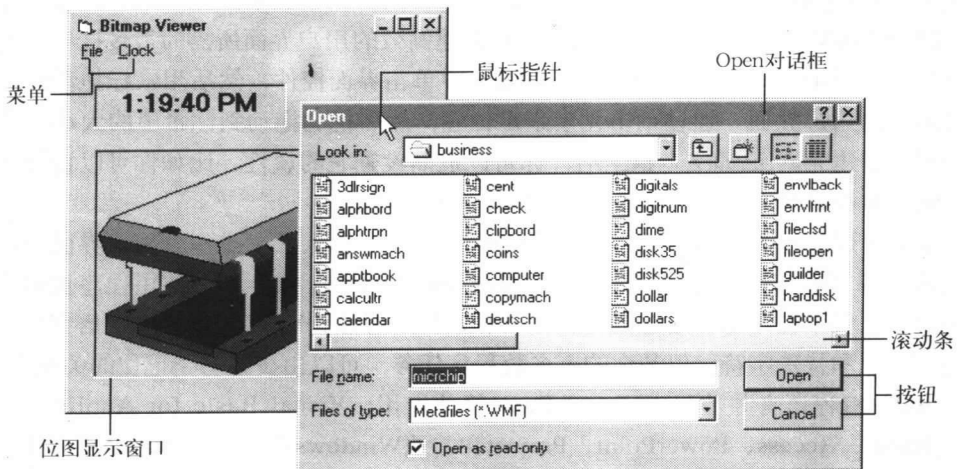


图 4

如果你熟悉Windows程序，你会认识大多数的用户界面元素。菜单中包含了很多Windows程序中都包含的命令；对话框、按钮和鼠标指针帮助用户向程序提供信息；窗口和滚动条允许用户查看显示在屏幕上的信息。为了最大限度地提高你的Visual Basic程序的可用性，按照标准的、可预知的方式使用Windows用户界面元素是很重要的。Microsoft公布了设计Windows程序的一些准则，当阅读本书时，你会学到最重要的界面设计准则。因为大多数Windows程序是由用户而不是程序员使用，因此应尽量把它们设计得具有相似的特征。

向自己提问

当你计划Visual Basic的编程工程时，会发现绘制一个你希望提供给用户的用户界面草图是很有用的。这样做不仅会帮助你设计用户界面，还会帮助你思考程序应如何工作。询问自己以下有关程序的几个问题：

- 编写程序的目标或者说任务是什么？
- 谁将使用这个程序？
- 程序启动时是什么样？
- 用户将向程序输入什么信息？
- 程序如何处理输入？
- 程序产生什么信息（输出）？

甚至在弄清如何在程序中实现你所希望的任务之前，就应该知道程序能干什么，以及它的外观怎样。在完成了这些预备工作后，就可以开始用Visual Basic编程了。

提示 第2章和第3章讨论如何创建一个高效的用户界面

使用Visual Basic建立程序

建立Visual Basic程序包含三个编程步骤：

- 1) 用Visual Basic控件创建用户界面。
- 2) 根据需要设置用户界面元素的特性，即属性。
- 3) 根据需要给一个或多个用户界面元素编写程序代码。

创建一个Visual Basic的Windows程序很容易，你在第2章会看到这一点。如果你曾使用过一个绘图程序如Microsoft Paint，你就会具有创建高效的用户界面所需的很多技巧。要创建界面元素，请在Visual Basic Toolbox（工具箱）中单击界面控件，然后用鼠标拖动并单击，就可以“画”出用户界面。通过单击元素来定位它，然后把它拖动到你希望的大小，这是件很简单的事。创建元素，比如文本框之后，你可以进而设置它的属性。比如你可以设置文本框的属性，使文本框中的文本显示为粗体、斜体或带下划线。

程序是通过在一个特殊的窗口（叫做Code窗口）中输入一个或多个用户界面元素的代码来完成的。编写程序代码比单独使用控件更能控制程序的工作。通过使用程序代码，你完全能够表达自己关于程序如何处理数据、输入和输出的想法。Visual Basic编程语言包括了几百个语句、函数和特殊字符，但是你的大多数程序任务，可以由少量易于记住的关键字来处理。Visual Basic编程语言并非只由Visual Basic软件专用；Visual Basic for Application包含在Word、Excel、Access、PowerPoint、Project和其他Windows应用程序中。如果你使用过这些应用程序的最新版，或者如果你过去具有使用Basic的经验，那么你一定认出了Visual Basic 6中大多数的关键字、语句和函数。

注意 约束Visual Basic编程语言的结构和内容的规则叫做语法 (syntax)。单词syntax取自希腊语syntattein, 意思是有秩序地调动军队。当你编写Visual Basic程序代码时, 你就是在为编译器指挥关键字。

Basic的起源

最初的Basic语言是由John G. Kemeny和Thomas E. Kurtz于1963年在Dartmouth大学创建的。它作为一门教学语言在大学和中学中迅速地普及起来, Bill Gates在70年代中期把它用在个人计算机上。从那时起, 出现了很多适用于个人计算机的Basic升级版, 包括Microsoft QuickBasic和MS-DOS Qbasic。尽管Visual Basic完美的图形编程环境和编程语言与早期的Basic解释程序有天壤之别, 但是早期Basic语言的优美和简洁性却在很大的程度上在语言中保留下来。把Visual Basic作为开发Windows应用程序, 如Excel的编程语言的主要原因就在于它的强大功能和易用性。

测试、编译和发布程序

提示 完整的编程过程通常叫做软件开发生命周期。

创建完程序之后, 你需要仔细地测试它, 确保运行得正确。如果你希望发布程序, 那么还需要编译器把它编译成可执行文件 (即单独的Windows程序) 发送给用户。这是Visual Basic Working Model所不能完成的 (你只能用学习版、专业版和企业版来编译)。现在介绍最后一个步骤, 让你对整个编程过程有一个完整的了解。如果你打算以后修正程序, 可以重复这些步骤, 这些步骤从制定计划和分析目标开始, 组成了软件开发生命周期, 如图5所示。

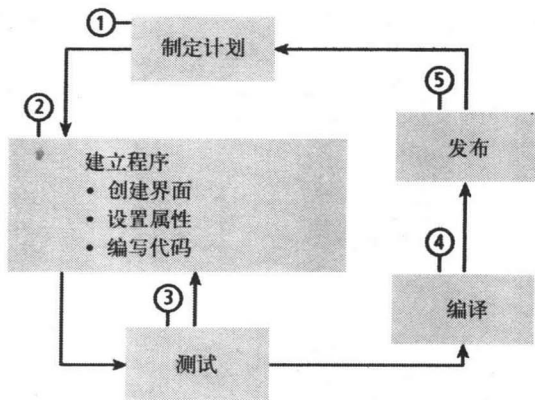


图 5

测试程序

测试程序就是根据实际操作中的各种条件, 确定程序工作得是否正常。导致程序运行停止, 或者没有产生预期的结果的错误叫做软件缺陷, 又称bug。如果你的软件使用了一段时间, 你或许曾遇到由于软件错误而导致的计算机故障。现在你作为一个程序员, 应该在把软件交给最终用户之前, 去掉这些错误。Visual Basic提供了一些优秀的调试工具来查找错误。你将在第6章中查找和纠正错误。

编译程序

当创建并测试完程序后, 你可以把它编译成Windows 98或Windows NT环境下的可执行 (.exe) 文件。用Visual Basic的零售版创建可执行文件很容易, 只需在File菜单上单击Make Project1.exe命令就可以了。本书包含的Visual Basic Working Model允许你运行Visual Basic程序, 但必须在Visual Basic编程环境内运行, 而不能作为一个单独应用程序运行。如果对你来说, 创建可执行 (.exe) 文件很重要, 那么你必须购买一种Visual Basic零售版。

发布程序

如果你想和朋友或同事共享Visual Basic的可执行文件，或者想销售你的程序，就需要把程序和一些必要的支持文件存放到盘中（或网络中），使得你的用户或客户能够获得它们。所有Visual Basic程序的运行都需要一个或多个动态链接库（DLL）文件，因此你需要把这些必要的文件从硬盘中拷贝到发售盘中，使用户能够运行。所需要的确切DLL文件取决于用户的操作系统，以及程序所用到的Visual Basic的特性（Microsoft Visual Basic 6 Tools文件夹中的Package & Deployment Wizard管理这个过程）。

如果你要销售软件，就需要升级到Visual Basic的零售版。零售版附带的文档和帮助文件会告诉你如何建立发布盘，以及如何创建一个定制的安装程序来安装你的软件。从现在起可以肯定，在你创建了Visual Basic程序后，你和你的用户将会立即享受到该新软件带来的好处。

让我们开始吧！

目 录

前言

第一部分 开始学习Visual Basic

第1章 打开并运行一个Visual Basic

程序1

1.1 Visual Basic的编程环境1

1.1.1 启动Visual Basic1

1.1.2 使用菜单条打开一个现有的工程2

1.1.3 移动、连接和调整工具4

1.2 用户界面窗体4

1.3 ToolBox4

1.4 Properties窗口5

1.5 Project窗口6

1.6 退出Visual Basic7

第2章 编写第一个程序8

2.1 Lucky Seven: 你的第一个Visual Basic

程序8

2.2 编程步骤8

2.3 正在创建用户界面9

2.3.1 创建用户界面9

2.3.2 移动命令按钮并改变其大小11

2.3.3 添加第二个按钮11

2.3.4 添加数字标签11

2.3.5 添加图像12

2.4 设置属性12

2.4.1 设置标签属性13

2.4.2 设置第四个标签的属性14

2.4.3 图像框属性14

2.4.4 设置图像框属性14

2.5 编写代码15

2.5.1 使用Code窗口15

2.5.2 编写Spin按钮的代码17

2.5.3 分析Command1_Click过程18

2.6 保存程序19

2.6.1 保存Lucky Seven程序19

2.6.2 运行程序20

2.7 向程序中添加内容21

第3章 使用控件22

3.1 控件的基本用法: “Hello World”

程序22

3.1.1 创建Hello World程序22

3.1.2 运行Hello World程序24

3.2 使用文件系统对象24

3.2.1 Browser程序25

3.2.2 创建Browser程序25

3.2.3 运行Browser程序27

3.3 接收输入的对象28

3.3.1 Online Shopper程序28

3.3.2 运行Online Shopper程序28

3.3.3 查看Online Shopper程序代码30

3.3.4 查看复选框和列表框的代码30

3.4 使用OLE对象启动应用程序33

3.4.1 创建Bid Estimator程序33

3.4.2 运行Bid Estimator程序35

3.5 使用数据对象查看Microsoft Access

数据库36

3.5.1 创建一个数据对象37

3.5.2 创建文本框和标签对象37

3.5.3 设置对象的属性37

3.5.4 运行MyData程序38

3.6 修改数据库38

3.7 安装ActiveX控件39

第4章 使用菜单和对话框41

4.1 使用菜单编辑器添加菜单41

4.1.1 创建菜单41

4.1.2 给菜单命令添加访问键43

4.1.3 改变菜单项的次序44

4.2 处理菜单选择44

4.2.1	编辑菜单的事件过程	45
4.2.2	保存MyMenu工程	46
4.2.3	运行MyMenu程序	46
4.3	使用Common Dialog对象	47
4.3.1	把MyMenu保存成MyDialog	47
4.3.2	添加公共对话框对象	48
4.3.3	使菜单命令无效	49
4.3.4	向Clock菜单添加TextColor命令	50
4.4	管理公共对话框的事件过程	50
4.4.1	编辑Open命令的事件过程	50
4.4.2	编辑TextColor命令的事件过程	51
4.4.3	使用Flags控制颜色的选择	51
4.4.4	编辑Close命令事件过程	52
4.4.5	编辑Exit命令的事件过程	52
4.4.6	运行MyDialog程序	53
4.5	指定菜单的快捷键	54

第二部分 编程基础

第5章 Visual Basic的变量和操作符	57
5.1 剖析Visual Basic的程序语句	57
5.2 使用变量存储信息	57
5.2.1 变量预留: Dim语句	58
5.2.2 不使用Dim声明变量	58
5.2.3 在程序中使用变量	58
5.3 使用变量接收输入	60
5.4 使用变量输出	61
5.5 使用特殊的数据类型	62
5.5.1 在代码中使用基本的数据类型	63
5.5.2 常量: 不变的变量	65
5.6 使用Visual Basic操作符	66
5.6.1 基本运算: +、-、*和/操作符	66
5.6.2 查看BasicOp程序的代码	67
5.6.3 使用高级操作符: \、Mod、^和&	69
5.6.4 操作符的优先级	70
5.6.5 在公式中使用括号	71
第6章 使用判断结构	72
6.1 事件驱动的程序设计	72
6.2 使用条件表达式	73
6.3 If...Then判断结构	74

6.3.1	用If...Then判断结构测试几种条件	74
6.3.2	使用If...Then验证用户	75
6.3.3	在条件表达式中使用逻辑运算符	77
6.3.4	使用And运算符添加密码保护	78
6.4	Select Case判断结构	78
6.4.1	在Select Case结构中使用比较运算符	79
6.4.2	使用Select Case结构处理列表框	80
6.5	查找和改正错误	81
6.6	使用Stop语句进入中断模式	86
第7章	使用循环和计时器	87
7.1	编写For...Next循环	87
7.1.1	使用Print方法显示计数器变量	87
7.1.2	在MyCtlArray程序中使用Step关键字	92
7.1.3	修改MyCtlArray程序	92
7.2	编写Do循环	94
7.2.1	避免死循环	95
7.2.2	使用Do循环转换温度	95
7.3	使用Timer对象	97
7.4	使用Timer对象设置时间限制	98

第三部分 创建完美的用户界面

第8章 使用窗体、打印机和错误句柄	101
8.1 向程序中添加新窗体	101
8.1.1 空白或预先设计的窗体	101
8.1.2 如何使用窗体	101
8.1.3 程序代码中的窗体语句	102
8.1.4 隐藏和卸载窗体	102
8.1.5 最小化窗体	103
8.2 使用多个窗体: Italian程序	103
8.2.1 运行Italian程序	103
8.2.2 在程序中创建第二个窗体	104
8.2.3 在事件过程中访问第二个窗体	105
8.3 把程序的输出送到打印机上	106
8.3.1 向MyItalian2程序添加打印支持	108
8.3.2 运行MyPrintFrm程序	109
8.4 使用PrintForm方法打印整个窗体	109
8.5 使用错误句柄处理错误	111

8.5.1 何时使用错误句柄	111	10.4.2 调用函数过程	144
8.5.2 设置陷阱: On Error语句	111	10.4.3 使用函数完成计算	144
8.5.3 关于Resume语句	112	10.5 编写子过程	147
8.5.4 创建一个软驱错误句柄	112	10.5.1 子过程的语法	147
8.5.5 错误句柄的其他技巧	114	10.5.2 调用子过程	148
8.5.6 指定重试次数	114	10.5.3 使用子过程管理输入	148
第9章 添加图片和特殊效果	115	10.5.4 创建文本框子过程	148
9.1 使用Line和Shape控件添加图形	115	10.5.5 引用传递	151
9.1.1 Line控件	115	10.5.6 ByVal关键字	152
9.1.2 Shape控件	115	10.5.7 值传递变量	152
9.1.3 使用Line和Shape控件	116	第11章 使用集合和数组	153
9.1.4 运行StartFrm程序	118	11.1 使用对象集合	153
9.2 创建图形命令按钮	119	11.1.1 集合表示法	153
9.2.1 检测MouseDown事件	119	11.1.2 在集合中引用对象	153
9.2.2 交换按钮	119	11.1.3 编写For Each...Next循环	154
9.2.3 创建工具条按钮	120	11.1.4 移动集合对象	154
9.3 向程序中添加拖放支持	124	11.1.5 在For Each...Next循环中使用Tag 属性	155
9.3.1 拖放的基本步骤	124	11.1.6 VBA集合	157
9.3.2 DragDrop程序	125	11.2 使用变量数组	158
9.4 向程序添加动画	128	11.2.1 创建数组	158
9.4.1 使用窗体的坐标系统	128	11.2.2 声明定长数组	159
9.4.2 在坐标系统中移动对象	129	11.2.3 使用数组元素	160
9.4.3 使用Move方法和计时器对象创建 动画	129	11.2.4 创建定长数组存放温度	160
9.4.4 向DragDrop程序中添加烟雾	129	11.2.5 创建动态数组	162
9.4.5 程序运行时放大和缩小对象	131	11.3 使用多维数组	165
9.5 在程序中命名对象	132	第12章 浏览文本文件 处理字符串	168
9.5.1 对象命名约定	133	12.1 使用文本框对象显示文本文件	168
9.5.2 使用Name属性改变对象的名称	134	12.2 在磁盘中创建一个新文本文件	171
		12.2.1 运行QNote程序	172
		12.2.2 查看QNote程序代码	174
		12.3 用程序代码处理文本串	175
		12.3.1 文本排序	175
		12.3.2 什么是ASCII	175
		12.3.3 在文本框内排序字符串	177
		12.3.4 运行SortDemo程序	178
		12.3.5 查看SortDemo程序代码	179
		12.4 对文本进行加密处理	181
		12.4.1 把文本加密成ASCII码	181
第四部分 管理全体数据			
第10章 使用模块和过程	137		
10.1 使用标准模块	137		
10.2 使用公共变量	139		
10.2.1 回顾Lucky Seven工程	139		
10.2.2 添加标准模块	140		
10.3 创建通用过程	142		
10.4 编写函数过程	143		
10.4.1 函数语法	143		

12.4.2 查看Encrypt程序代码	182	15.3 处理RTF格式化的事件过程	213
12.4.3 使用Xor操作符	183	15.3.1 All Caps命令	213
第13章 管理Access数据库	186	15.3.2 Font命令	213
13.1 在Visual Basic中操纵数据库	186	15.3.3 Bold、Italic和Underline命令	214
13.1.1 创建定制的数据库应用程序	186	15.4 使用Windows剪贴板编辑文本	214
13.1.2 使用绑定控件显示数据库信息	186	15.4.1 Cut、Copy和Paste命令	214
13.1.3 使用文本框对象显示数据	187	15.4.2 Find命令	215
13.1.4 运行Course程序	187	15.5 使用Rich Textbox控件管理文件 操作	215
13.2 使用Recordset对象	189	15.5.1 Open命令	215
13.2.1 在Students.mdb中查找数据	189	15.5.2 Close命令	216
13.2.2 运行MyFindRec程序	190	15.5.3 Save As命令	217
13.3 向Students.mdb数据库中添加记录	191	15.5.4 Print命令	217
13.3.1 让用户向数据库添加记录	191	15.5.5 Exit命令	218
13.3.2 运行MyAddRec程序	192	15.6 在文档中显示RTF代码	218
13.4 从Students.mdb数据库中删除记录	193	第16章 显示过程和状态信息	220
13.4.1 允许用户从数据库中删除记录	193	16.1 使用Progress Bar控件显示任务过程	220
13.4.2 运行MyDelRec程序	194	16.1.1 安装Progress Bar控件	220
13.5 制作文件备份	195	16.1.2 向工具箱添加Windows Common Control	221
13.5.1 使用FileCopy制作Students.mdb的 备份	195	16.2 Progress程序	221
13.5.2 运行MyBackup程序	196	16.2.1 运行Progress程序	221
第14章 连接Microsoft Office	197	16.2.2 查看Progress程序的代码	222
14.1 使用自动化编写应用程序对象	197	16.3 用Slider控件图形化管理输入	223
14.1.1 在Visual Basic中使用自动化	197	16.3.1 运行RTFEdit2程序	224
14.1.2 Visual Basic的Object Browser	198	16.3.2 查看Slider控件的源代码	224
14.2 从Visual Basic中自动化Word	200	16.4 用Status Bar控件显示应用程序的 信息	226
14.2.1 创建个人拼写检查器	200	16.4.1 运行RTFEdit2程序	226
14.2.2 运行MyUseWord程序	202	16.4.2 显示Caps Lock和Num Lock的 状态	229
14.3 从Visual Basic中自动化Excel	203	第17章 用Internet Explorer显示HTML 文档	230
14.4 从Visual Basic中自动化Microsoft- Outlook	204	17.1 开始使用Internet Explorer对象	230
		17.1.1 向程序添加对Microsoft Internet Controls的引用	230
		17.1.2 研究Internet Explorer对象模型	231
		17.2 显示HTML文档	232
		17.2.1 运行ShowHTML程序	233
第五部分 高级技术			
第15章 用Rich Textbox控件进行字处理	209		
15.1 浏览专业的ActiveX控件	209		
15.2 Rich Textbox控件简介	210		
15.2.1 向工具箱中添加Rich Textbox 控件	210		
15.2.2 运行RTFEdit程序	211		

17.2.2 查看ShowHTML里的Internet Explorer代码	235	18.2.2 高级的查找和排序操作	245
17.3 响应Internet Explorer事件	236	18.3 查找Biblio.mdb数据库	247
第18章 用FlexGrid控件管理数据	238	第19章 探索ActiveX Data Objects	250
18.1 把FlexGrid用作通用的工作表	238	19.1 深入ADO内部	250
18.1.1 理解行和列	238	19.2 使用ADO ActiveX控件	250
18.1.2 向单元格中插入图形	239	19.2.1 安装ADO控件	251
18.1.3 选择单元格	239	19.2.2 创建一个ADO对象和绑定控件	251
18.1.4 格式化单元格	240	19.2.3 创建数据源名称	252
18.1.5 添加新行	240	19.2.4 把ADO数据绑定到文本框对象	254
18.1.6 使用FlexGrid控件记录销售数据	241	19.2.5 运行ADO控件演示程序	255
18.1.7 编写FlexGrid程序代码	241	19.3 编写ADO程序代码	255
18.2 使用FlexGrid显示数据库记录	243	19.3.1 创建MyAdoData程序	256
18.2.1 运行DataGrid程序	243	19.3.2 运行AdoData程序	258
		19.4 Visual Basic编程的其他资源	259

第一部分 开始学习Visual Basic

第1章 打开并运行一个Visual Basic程序

本书第一部分将要讲述有关Microsoft Visual Basic的重要知识, 这些知识都是实现令人难忘的复杂任务的基础。经过几章的学习之后, 你会发现使用Visual Basic编写功能强大的Windows程序相当容易。即使在此之前从未编写过程序, 你也会发现, 编程中用到的许多逻辑功能和技巧, 与日常生活中的推理和计算机技巧是相似的。本章, 你将学习如何启动Visual Basic, 如何使用Visual Basic开发环境来打开和运行一个简单的程序; 将了解Visual Basic主要的菜单命令和编程步骤; 将学习打开并运行一个叫做StepUp的简单Visual Basic程序, 将尝试用属性来改变程序的设置, 还将学习如何退出Visual Basic。

1.1 Visual Basic的编程环境

Visual Basic的开发环境中包含了所有用于快速、高效地创建功能强大的Windows程序需要的工具。按下面步骤启动Visual Basic。

注意 如果还没安装Visual Basic Working Model和本书的练习文件, 请按“前言”中所要求的去操作, 然后再返回到本章学习。

1.1.1 启动Visual Basic

1) 在Microsoft Windows中, 单击Start按钮, 把鼠标移到Programs, 然后再移到Microsoft Visual Basic 6文件夹上。这时, 将列出Microsoft Visual Basic 6文件夹(包括文件夹和一个应用程序图标)。

提示 本书中的练习是为Visual Basic Working Model设计的, 你也可以使用Visual Basic 6开发系统的零售拷贝(学习版、专业版和企业版)。如果你的Visual Basic来自Microsoft Visual Studio开发工具套件, 请把鼠标指针移至Microsoft Visual Studio文件夹中的Visual Basic程序图标上。

2) 单击Microsoft Visual Basic 程序图标, 出现New Project对话框。该对话框提示你选择要创建的工程类型(对话框的具体内容依赖于所使用的Visual Basic版本)。

3) 单击Open打开默认的新工程。在Visual Basic开发环境中打开新工程后, 会出现如图1-1所示的窗口和工具。

Visual Basic编程环境包含的编程工具能帮助你创建Visual Basic程序。菜单条可以访问大多数命令, 这些命令控制着编程环境。菜单和命令的作用, 与所有Windows程序中的一样。当然也可以使用键盘和鼠标来访问它们。位于菜单条下面的是工具条, 它集中了用于执行命令、控制Visual Basic编程环境的快捷按钮。如果你曾使用过Microsoft Excel和Microsoft Word,