

前言

P. 1 关于作者

Jim Boyce 是 WINDOWS 杂志的特约编辑,同时也是 CADENCE 杂志的报刊专栏作者,而且还是其他计算机刊物的定期投稿者。他自从七十年代以来就涉入到计算机领域,在某种程度上是作为结构设计者、产品规划人员、系统管理人员、编程人员以及大学讲师而使用计算机。作者对 DOS、Windows 和 UNIX 操作系统有相当丰富的经验。JIM 所编写的、由新骑士出版社(New Riders Publishing)出版的几本书《Maximizing Windows 3.1》、《Windows for Non-Nerds》和《Keeping your PC Alive》相当畅销。Jim 还编写及合作编写了许多其它书籍:

- AutoCAD 手册
- Windows 3.1 基础
- Windows for Workgroups 内核
- Windows 3.1 内核
- Windows 3.1 网络
- 优化 Windows 3.0
- 优化 MS-DOS 5.0
- 初学 AutoCAD
- AutoCAD 内核
- AutoCAD for Windows 内核
- 优化 AutoCAD 卷 1

Jim 是一位德克萨斯州人,现居住在明尼苏达州,他爱好广泛,但不适应气温变化。读者可以通过 CompuServe ID 76516.3403 与他联系,或者通过在 Internet 网中的地址 76516.3403@CompuServe.com 建立联系。

P. 2 奉献

本书专门奉献给 Tina Haugse。亲爱的读者,您是一位幸运的人,好好学,不要辜负她的期望。

P. 3 致谢

许多人在本书的编写出版工作中给予了许多方式的帮助。在此表示衷心感谢:

Tim Huddleston, Cheri Robjnsn 和 Rob Tidraw 为帮助编写本书,为处理一些编写新书经常遇到的头疼的问题付出很多精力。

Forrest Houlette 在编写本书时提出不少建议,作了不少指导。Scott Flanders 和 Lloyd Short 毫无怨言地辛勤开发未知领域。

Stacey Beheler 对本书作出了辛劳而杰出的工作。

Steve Weiss 为本书的润色及可读性做了非凡的及时的工作。也感谢 Sarah Kearns 为编辑本书做出的贡献。

Bill Beem 对本书的技术编辑做出了杰出的工作。由于他的检验及校对,使本书更加完善。

还要感谢 Macmillan Computer Publishing 公司的生产部门,正式出版本书。

John Schmitt 给予本书许多建议。

Snerdly T. Whipplefoot, 感谢他在最后的阶段将整个工作从濒临解散中挽救回来。

P. 4 商标说明

New Riders Publishing 公司尽最大努力提供本书中提到的有关公司名称、产品及服务的商标信息。这些商标信息有不同的来源。New Riders Publishing 公司无法逐个检验这些信息的准确性。

Microsoft 和 MS-DOS 是注册商标,Windows 是 Microsoft 公司的商标。

本书中所涉及到的其他产品的商标,归生产它的公司所有。

P. 5 警告及声明

本书是为使用 Visual Basic 编程提供信息而编写的。我们尽力使本书尽可能的完整并准确,但难免有疏漏之处。

本书提供的信息是基于实践的基础之上。作者与 New Riders Publishing 公司对于任何个人或团体有关本书中包含信息所导致的疏漏或对其中程序修改所引起的问题不负任何责任。

简介

I.1 VB 及本书的意义

现在让我们谈谈 Visual Basic。

恕我冒昧地猜测买本书的顾客不是程序员,那么很容易猜测出,他想要学习使用 Visual Basic 来编程。因为这个用户想买关于 Visual Basic 的书的原因只有两个:或者是想学习使用 VB 编程,或者是需要一个新的“隔门”。虽然用于建造好隔门的砖石都很便宜并且好买到,但如果个人真想买书用来建造隔门我也不会阻拦。我愿意用我所有的信用卡来买他房子中组成每个门的书。

我判断本书读者不是程序员的原因是:如果是程序员,他会寻找一本大约十二英寸厚的 Visual Basic 书,其中充满各种深奥的但确实对理解 VB 很重要的资料。如果是已经对程序结构、变量、数组、控件、事件以及其他所有计算机编程的专用术语很了解的顾客,那种类型的书很合适。但如果并非程序员,或者仅仅使用过 BASIC,那种类型的书只能使人着急而没有什么用处。因此,这里提一个小小的建议:加书名号是为这些用户服务的书!

为什么说对于初学用户来说,这是一本学习 Visual Basic 的好书呢?以前所见到的关于 Visual Basic 方面的书都包括了 VB 函数和语句,并介绍了在程序代码的一小段中是如何使用它们的,这当然很好,但这些书中绝大多数没有很好地解释一个一般的 VB 程序是如何构成的。对于一个不明白做什么的用户来说,除非已经有许多使用其他语言编程的丰富经验,介绍如何使用 VB 编程可以说并不是最重要的。这本加书名号正是基于这种想法而编写的。

I.2 如何入门

本书是从用户不很了解有关编程知识入手的。如果已有一些使用 BASIC 或其他语言编程的经验,那么很好——这表示已经超前了几步。但本书仍是学习 Visual Basic 的好的起点。

本书尽量避免使用大量技术术语以及大段的程序代码,代之以对基本概念为重点的介绍。本书用一通俗的、非术语性语言解释了一个 VB 程序的各个组成部分。当介绍一个新的概念时,例如变量,将说明什么是变量,以及为什么在程序中要用到变量。我们不仅告诉读者做什么,还要告诉读者为什么这样做。

除了着重于基本概念之外,加书名号一书还将使用面向任务的途径。也许用户知道想要做什么,但不知道如何用技术语言来表示。例如,假如用户问我:“Gee,我怎样编写一个简单的程序来显示一个目录列表,并显示不同磁盘上的文件呢?”我们将尽量用这种方式展开这个话题,而不是着重于文件操纵的语句与函数。

1.2.1 本书的主要内容是什么

本书并不介绍有关 Visual Basic 的所有内容(尤其以这样的代价...)。但无论如何,本书将帮助读者涉足并学习基本的 VB 概念,学会如何编写一些简洁的小程序并能自由运用 VB。懂得使用 VB,仅仅几百条语句代码就可以做出一个真正有价值的程序,给人留下深刻印象。读完这本书后,就可以阅读那种十二英寸厚的 VB 书,而不会不知所云了。

1.2.2 第一部分:学习程序

第一部分“简单的想法与任务”,引导读者接近 Visual Basic。第零章,“做好准备工作”,开始生成一个程序。不用担心,这部分十分浅显易懂。在第一章结束时,读者将建立一个工作程序。第一部分的其他章,“窗体、控件及其他小技巧”、“进一步的了解”、“代码运行到哪儿去了”、“关于文本和数字”和“作出决定”更加详细地介绍了一个 VB 程序的某些基本部分,并教读者如何开始编写程序代码,还将学习使用诸如数字及文本字符串的数据。

1.2.3 第二部分:学习做一些有用的事

在第二部分“做一些实用的工作”中,用户开始为特定的任务编写 VB 程序。例如在第六章“操作简单的文件填充”中,将学习如何创建一个程序,使其作用类似于简单的文件管理器,学习与磁盘、目录列表与文件列表有关的内容。第二部分的其他各章,教给读者如何在自己的程序中包含菜单、图形以及对话框。第二部分还介绍如何将资料从用户程序中传送到打印机上。读者将知道向程序中加入菜单是十分方便的。

1.2.4 第三部分:开始深入并且或许需要一个小保护程序

第三部分,“认真对待”中包含了一些较复杂的编程任务及主题。在这部分可以学习错误处理和使用大量数组数据与特殊的数据结构。还将学习如何读写文件,使用户程序将其数据保存到文件中成为可能。第三部分的最后简要介绍了 Visual Basic 提供的一些调试工作,可以用来跟踪程序中出现的错误并解决它们。

1.3 获得帮助

在使用本书的过程中,总是可以很方便地通过击键获得帮助。Visual Basic 具有功能非常强大的帮助系统,不仅解释如何使用 Visual Basic,而且包含关于 VB 语句和函数的解释,甚至还包含代码示例。

获得一般帮助的方法是:选择 Visual Basic 的 Help 菜单下的 Contents 选项,这就打开了主要的 VB 的 Help 文件并显示它的内容及页数。如果要想寻找关于某个指定主题的帮助,可以从 Help 菜单中选择 Search for Help On 选项。如果想获得关于某个特定函数,语句或关键字的帮助,可以先将这个字高亮度显示,然后按下 F1 键。Visual Basic 就会打开 Help 文件,并显示包含有被高亮度显示的函数所在的页数。可以浏览此主题或显示一个代码示例,来帮助理解此函数或语句是如何工作的。也可以将此代码示例的全部或部分复制到剪贴板(Clipboard)中,然后再将它粘贴到用户自己的程序中去,这样就可以节省用户自己再键入这

些代码的时间。

I.4 准备好

学习本书的准备工作不多。如果还没有 Visual Basic 软件,要去买一套。因为没有它无法编写 VB 程序。但在你出发之前,先决定你是否要买 Professional Edition。Professional Edition 中包含了许多附加的 gizmos 和 gadgets,可以用于添加诸如 3D 控件和动画按钮等新东西。它还有添加邮件能力、通信功能的控件,以及向程序中添加其他复杂特性的控件。以下是 Professional Edition 中包含的额外 gadgets 的列表:

- 动画按钮(Animated Button):此控件使用户可以显示一系列位图来生成动画按钮、多状态按钮和复选框。
- 通信(Communication):此控件为用户程序提供一系列完整的通信功能,包括文件传送和其他数据通信性能。
- 标尺(Gauge):此控件可以生成线性或针状标尺指示器,来显示程序中任务的进度。
- 图形(Graph):此控件可以使用很多种不同的风格创建并显示图形。
- 键盘状态(Key status):此控件可以显示并修改 CAPSLOCK, NOMLOCK, INS 和 SCROLL LOCK 键的状态。
- MAPI 控件:这些控件可用于向用户程序中加入 Mail 能力。可以编写自己的 Mail 系统,也可以加入从程序中发送和接收邮件信息的能力。
- 屏蔽编辑(Masked Edit):此控件与标准文本框类似,但它提供的特征能使用户容易地控件和限制输入到框中的数据格式。
- Multimedia MCI:此控件使用户程序可以控件多媒体设备,例如 CD 驱动以及其他支持 MCI 界面的设备。
- Outline:此控件可以生成水平列表框,其分支可以扩展或收缩,就象在 File Manager 中的目录列表一样。
- 画笔控件(Pen Control):这些控件可以发展 Windows 的 Pen Computing 环境的程序。
- Picture Clip:此控件可以在一个图形框或窗体中显示一个位图的某个部分。使用这个控件可以在用户自己的程序中生成位图图像。
- Spin Button:此控件可以增加或减少在一个文本框、标签或其他控件中显示的数目。
- 3D Control:这些控件代替了标准的二维控件,并包括三维复选框、命令按钮、框架及选项按钮等。
- 数据访问(Data Access):Professional Edition 中具有此功能,可以使用户以许多不同的格式,用各种不同的方法生成并访问数据库文件。
- Crystal Reports:这个附加控件可以让用户很容易在程序中设计并完成报告,这时开发有关数据库的应用程序十分有利。

虽然本书不包含对 Professional Edition 的介绍,但本书是帮助用户学习概念及编程技术的好的起点。有了这些知识,才能在将来使用这些新加的控件编写程序。Professional Edition 中的许多新加的控件与 Standard Edition 中的控件很相似。当对 Standard Edition 较

熟悉之后,再自学其他控件应该不会有什问题。

1.4.1 安装示例程序

本书使用了一些程序实例来进行 VB 教学。本书带有一片包含这些程序实例和代码的磁盘,在研究各章中的例子时需要这些程序实例,因此应在硬盘上安装这些文件。具体方法如下:

1. 启动 Windows,将本书另购磁盘插入到系统软盘驱动器中。
2. 在 Program Manager 中,选择 File,Run。
3. 在 Run 对象框的 Command Line 编辑框中,键入 A:SETUP(如果软盘插入在 B 驱中,应键入 B:SETUP),然后选择 OK。
4. 按照 Setup 窗口中的指示,将文件安装到所选择的目录下。

以上这些指示已足够了。如果想成为一名程序员,毕竟要学会独立思考。因此,本书中所有指示保证足够详细。

1.4.2 使用本书

阅读本书时,当然可以坐在比较舒适的座位上。但要想学习,还应坐在计算机前面。因为只有这样才能使用那些例子学习,并且监视器的射线将使人有良好的“程序员的感觉”。

1.4.2.1 协议和特殊字体

首先要了解本书中所采用的几种不同的印刷字体的意义:

- 象这种小形字体,用于表示程序代码。这种字体用得不多,但有些地方确实需要用到。
- 使用斜体印刷的程序代码(文本中)主要表示该斜体字是一个变量——实际情况视场合而定。在练习的程序代码中,使用斜体字表示解释代码的注释。
- 斜体字也用于指出将要被定义的新名词和技术。
- 热键(或简捷键),其某个字母使用黑体,并加有下划线。
- 用户键入的文本输入使用黑体字表示。

1.4.2.2 学习 VB 应制定计划吗

一般来说,应按顺序阅读本书,至少第一部分是这。第二部分中,可以跳过某些章节,而阅读与所做的任务有关的章节。例如,如果暂时不想学习有关菜单的内容,可以跳过第七章。

第三部分也同样。只要具备了基本知识,就可以相当自如地选择当前所需内容进行学习。到那时,就可正式着手 VB 的使用了。

当然,这并不意味着接部就班地学习,从前到后逐渐入门的途径将具有高度的逻辑性,完整的实践性,以及对 Visual Basic 全面的了解。这不正是广大读者所希望的吗?

无论怎样做,不必恐慌! 每个人都可以学会使用 Visual Basic 编写程序。

I.5 练习简介

本节列出了本书中所有的练习,将它们按章排列,并提供了每个练习的简要介绍。这将

有助于快速查找到所需内容的位置。

第零章

绘制名为 Useless 的界面：

通过一个简单的程序即在一个窗体上绘制控件，来创建界面(窗口)。

设置设计时属性：

当用户正在设计程序界面来控制控件的外形和功能时，可设置控件和窗体的属性。

编写程序代码：

增加许多简单的 Visual Basic 语句至一个小程序。

编制 Useless 程序：

将一个 Visual Basic 项目文件编译为 Windows 可执行(EXE)文件。

测试 Useless；对用户第一个示范程序进行测试。

第一章

设置窗体的边框类型：

设置窗体的边框类型，以便在程序运行时控件窗口外观以及用户改变窗口大小的能力。

激活和禁止窗体：

在设计阶段激活和禁止窗体，以使得窗体对用户来说是可见或不可见的。

为窗体分配图标：

在设计阶段为窗体制定一个图标。当程序在缩小化方式运行时。该图标便代表此程序，在程序组中也是用此图标来代表程序。

改变鼠标外形：

当鼠标经过一确定的控件或窗体时，改变鼠标的形状。

利用文本框属性来工作：

利用文本框的 PasswordChar 属性来建立一个简单的密码输入框。

用选项按钮工作：

用选项按钮实验，来看它们是如何协同工作的。

使用滚动条控件：

在程序中增加滚动条控件，使用户可以通过它来增加或减少数值。

第二章

指明开始窗体：

指明程序开始运行时先执行哪个窗体或过程，这也就定义了程序的“入口点”。

利用 load 事件工作：

使用窗体的 Load 事件来定义当窗体被载入内存时执行哪些操作。

卸掉窗体：

使用窗体的 Unload 事件，使得当程序结束或窗体从内存中卸掉时执行某些任务。

单击鼠标：

为按钮的 Click 事件添加代码。当用户单击按钮时，执行 Click 事件过程。

双击鼠标:

为列表框的 DblClick 事件过程添加代码,用户可以在表中双击鼠标来选中一表项。

利用改变来工作:

使用文本框的 Change 事件可以在文本框中的文本改变时执行一个过程。

使显示按钮工作:

为按钮添写代码,使得程序可在图像框中显示图像。

使关闭按钮工作:

为按钮添写代码,当用户单击它时结束程序运行。

使双击鼠标工作:

为文件列表框添加代码,使用户在文件上双击鼠标时能将其显示在图像框中。

建立可执行文件:

把在图像框中显示位图的程序编译成可执行文件。

第三章

通用过程:

测试一个计算耗牛奶价格程序的通用过程,通用过程可简化程序,因为这可被其他过程调用。

建立一个函数过程:

测试一个程序,该程序使用通用函数计算耗牛奶价格,并将其值返回至调用函数。

第四章

利用不同数据工作:

测试不同类型的数据,用户可在程序中使用字符串、整型数、日期以及更多其他的东西。

对各种数据类型进行转换:

当存储在一个变量中的数据的类型要改变时,如从整型转变为实型,可以使用各种数据类型。

用 Dim 显式说明一个变量:

使用 Dim 语句将变量显式地说明为整型、字符串以及其他数据类型。

理解范围:

演示一个程序,它将帮助用户理解变量范围(Variable scope),Variable scope 图例说明一个变量在整个程序中如何创建、定义及使用。

关于文本:

在一个程序中试验,如何控件文本串,教会用户如何做类似将一个串分为单独的词的事以情。

第五章

使用 if... then:

检测一个程序,解释 if... then 语句的使用。只有当条件为真时才执行语句。

使用 if... then... else:

增加一些代码至 if... then 结构中,以便在某条件不成立时执行第二个指令集。

退出一个子过程:

使用 Exit Sub 语句使从一个未完成的过程中退出。

使用 Select Case:

检测一个程序,该程序使用 Select Case 结构来测试具有多种参考值的条件,然后依据各自条件表达式来执行各语句集。

使用一个 For...Next 结构:

试验一个程序该程序,使用 For...Next 结构,用指定次数执行一些语句集(在语句循环体内)。

使用一个 Do...Loop 结构:

检验一程序,程序内容使用 Do...Loop 结构,在语句循环体内执行语句集,直至条件为真。

修复 Do...Loop:

修复一个 Do...Loop 结构以便能正常终止。

第六章

绘制文件控件:

在一窗体内绘制一个目录列表框、驱动器列表框和文件列表框,最终完成一个简单的文件浏览实用程序的创建。

同步文件控件:

增加代码至驱动器、目录和文件列表框,使他们能同时工作。在目录列表内选择目录,例如:在选定的目录中改变文件列表以显示文件。

处理"Device Not Available"错误:

当用户试图选择一个过程时,增加一些错误检测代码至该过程中,来解决磁盘未能利用的问题。

增加一个删除的特征:

在一个简单的文件实用程序中使用 Kill 语句删除一个文件。

拷贝和移动文件:

在一个简单的文件实用程序中使用 FileCopy 和 Kill 语句增加拷贝和移动文件的能力。

文件更名:

在一个简单的实用程序中使用 Name 语句对文件进行更名。

修改 FileMan 来启动程序:

增加一些功能至一个文件实用程序中以启动程序。

增加双击至文件列表:

在一个文件列表框中的一个程序文件名上,允许用户双击以启动程序。

第七章

创建一个菜单:

使用 Menu Design Window(菜单设计窗口)来增加一个标准下拉式菜单至一程序中。

增加更多菜单项:

增加程序菜单和菜单项至一个程序窗口中。

编辑 Delete 菜单项的代码:

增加一些代码至一个允许用户删除文件的菜单项。当用户选择了一个菜单项,该示例为此按钮分配一单击键。

分配热键:

在用户菜单中给菜单项分配热键,诸如 Ctrl+C 为 Copy 菜单项,Ctrl+V 为 Paste 菜单项。

增加分隔符至文件菜单:

修改一个菜单使之包含分隔符。分隔符是分隔一个菜单不同部分的水平线。

完成 FileMan 菜单:

增加一些有选择性的菜单来完成一个文件实用程序的菜单。

增加 Checks 至一个菜单:

增加一些代码至一个菜单单击事件过程中,以便当用户单击菜单项时,能从菜单项中增加或删除一个检测标记。

改变 DOC/Text 文件项:

增加一些语句至一个文件实用程序,以便从用户能选择的类型中提供一些预定义文件类型来显示一些列表文件。

完成改变:

增加一些最后的附件至一菜单中,使用户能通过选择各种菜单项来选择一个文件框的文件显示模型。

增加 Cascading 菜单项:

创建一个级联菜单。

完成 FileMan:

增加一些最后的附件至 FileMan 程序中——一个可以删除、复制、移动和执行文件的文件实用程序。

第八章

设置对话框显示模式:

使用 Show 或 Hide 方式使一个窗体可显示或隐藏。

填充对话框:

在显示对话框之前初始化一个对话框(设置控件的参考值)的控件。

在对话框基础上修改设置:

当用户在对话框作了些改变后,创建一个可以读取对话框控件状态的过程。然后使用对话框设置来改变其它程序设置和变量。

建立控件数组:

开始创建一些选项按钮,其功能为一个控件数组(Control Array)。一个控件数组允许用户创建许多可分享同一事件过程的控件。

完成控件数组:

完成一个由选项按钮组成的控件数组。

为控件数组增加一个过程：

创建一个与控件数组有关的过程。该过程执行的任务基于数组中哪一个控件产生事件。

第九章

增加项目至一列表：

检测一个使用 AddItem 方式来增加一个项目至一列表框的程序。当包含该列表框的窗体被装载后该项目才增加。

为一个 Spin 增加 COMBO. MAK：

对一个这样的程序进行试验，当程序启动时，该程序增加项目至一个列表框。

初始化组合框：

增加一些语句至一个过程，该过程在显示之前，初始化一个对话框。

增加项目至 List1：

增加代码至一过程，当用户单击一按钮时该过程将一文本框中的内容放置于一个列表框内。

从 List1 中删除项目：

创建一个过程，当用户选择项目并单击按钮后将使用 RemoveItem 方式从一列表框中删除项目。

增加一个命令对话控件：

在一个窗体中绘制一个命令对话控件。一个命令对话控件允许用户程序使用标准对话框来对文件进行存取、打印、字体选择和颜色选择。

增加开放式对话至 COMMDLG. MAK：

创建一个过程，显示一命令文件开放式对话框，来读取关于对话框的参考值状态，包含只读复选框、文件名等等。

增加一个打印设置特征：

创建一个过程，允许用户程序显示打印设置对话框以及标识缺省打印机设置。

开启打印命令：

建立一个显示 Print 对话框并确定用户在对话框中选择的过程。

运行字体对话框：

建立一个过程，使用通用的 Font 对话框支持用户对字体、字体大小及其它字体特征的选择。

运行颜色对话框：

建立一个过程，使用通用的 Color 对话框支持用户对颜色的选择。

第十章

装载图形到窗体：

建立一个过程，将位图，图标或元文件放到窗体背景中。

设计期装入图形：

设计程序时，将图形装入到窗体或其它控件中。

加入图像控件：

在窗体中画出一个图像控件并添加相应程序代码,使得装载窗体时显示一幅图像。

设定窗体中的图像框尺寸：

建立一个过程,在用户重设窗体大小时,自动重设图像框尺寸匹配窗体。

运行期移走图像：

使用 LoadPicture 在运行时从图像框控件中移走一幅图形。

使用 Stretch 属性：

设置图形控件的 Stretch 属性,使图形控件内的图形随控件拉伸。

保存图形至文件中：

显示一个 Save As 对话框,并使用 SavePicture 语句支持用户保存位图、图标或元文件至一个文件中。

绘制工具条：

通过绘制包含工具条按钮的图像框,建立程序的工具条。

绘制按钮：

绘制图像框内的图形控件,用作程序工具条内的按钮。

分配按钮的图形：

设置图形至用作工具条按钮的图形控件中。

使工具条按钮运转：

给图形控件的 Click 事件过程加入代码,使它们执行按钮代表的功能。

重定位 Image1 控件：

将程序稍加改动,使图形控件总显现在工具条下方。此技术对于放置控件至工具条底线尤其重要。

从按钮产生“按动”的感觉：

做一些简单的图形切换工作,使得用户单击工具条按钮时,按钮产生“被按下”的视觉感觉。

第十一章

设置左端和顶端空白：

运用 Printer 对象的 ScaleLeft 和 ScaleTop 属性设置打印时的左端和顶端空白。

第十二章

给 FileMan 加上错误处理：

建立一个简单的错误处理程序,用作在驱动器或目录不可用时显示出错信息。

编写完整的错误处理程序：

建立一个更复杂和更巧妙的容错处理程序,用作测定特定类型的错误并作出相应处理。

建立 DiskErrorHandler 过程：

建立一个通用容错过程,且可以被程序中的其它部分调用。

产生自己的错误：

使用 Error 语句产生错误,以测试程序和错误处理子程序。

第十三章

定义一个数组:

用 Dim 语句建立一个数组,数组可用来存贮数据序列。

使用下标(索引)变量:

用变量操作数组中的特定项目。

重设数组大小:

重设数组以容纳更多的项目。当数组已填满时,用 Redim 语句使之变大,就可以给数组添加项目。

修改滚动条:

对程序稍加改动,使滚动条控件的 Max 属性总与数组数目相匹配。这是同步滚动条和数组的最好手段。

添加删除功能:

建立一个简单过程,可以让用户删除数组的项目。数组包含一系列人名。

添加全选能力:

修改一个简单的文本编辑器,加入全选文档中所有文本的功能。

开启拷贝和粘贴(Paste)项目:

建立一个过程,允许用户拷贝文本到剪贴板或从剪贴板粘贴文本。

剪切功能的编程:

建立一个过程,允许用户从文本框中剪切选择的文本并放入剪贴板。

第十四章

以顺序存取方式打开文件:

建立一个简单过程,以顺序存取方式打开一个文本文件。

读取文本文件至文本框中:

建立一个简单过程,实现将文本文件内容拷贝到文本框中。这是创建文本编辑器的第一步。

保存顺序文件:

编写一个过程,用顺序存取方式拷贝多行文本框的内容到文件中。这就实现了文本编辑器的保存文件变化或建立新文本文件的功能。

定义数据类型和数组:

建立一个数据类型,用作保存地址簿程序中的数据记录。

创建 FileOpen 过程:

编写一个过程,以随机存取方式打开文件。

测试 FileOpen 过程:

测试以随机存取方式打开文件的 FileOpen 过程。

创建 SaveFile 过程:

编写一个过程,以随机存取方式将定长记录保存到一个文件中。

察看 OpenFile 过程:

检测一个过程,此过程以二进制存取方式打开一个文件,并从文件中拷贝数据至一个数据结构。

第十五章

修改运行时的错误:

学习在测试程序时,如何修改运行时发生错误的有关代码。

跳到不同的语句执行:

控件程序在发生错误时跳到指定语句处执行。

单步执行:

学习如何使程序单步运行,以检测和调试程序代码。

单一过程执行:

单步执行程序直至遇到对其它过程和函数的调用。以普通的运行模式过程单步执行调用过程和函数。

在 Debug 窗口中使用即时区域:

使用即时区域修改变量和执行其它调试任务。

第一部分

简单的想法和任务

本书第一部分向用户介绍 Visual Basic,从一些很基本的概念开始,逐步深入。下面列出第一部分包含的章节,并对各章内容作一简要解释:

- 零章:作好准备工作。本章未向用户介绍很多深奥的概念。相反,只是提供一些“按下这些按钮”的练习,让用户熟悉 Visual Basic。
- 一章:窗体控件和其他小技巧。第一章,开始集中讲解概念,用户会学习窗体、控件及这些对象对于一个 Visual Basic 程序的重要性,用户还会学到属性。
- 二章:进一步的了解。本章介绍事件及事件驱动编程,教用户在程序中通过建立事件过程来使事件发生,事件过程和各种不同类型的事件相关,比如单击鼠标按钮。
- 三章:代码运行到哪儿去了? 第三章开始接触一些技巧(但不是很多),本章介绍一些东西比如事件过程运行到何处去了,一般的函数和过程放在何处,一般的 Visual Basic 程序是怎样构成的。
- 四章:关于数字和文本。本章介绍变量,用户可能操作的不同数据类型,及操作文本和数字的技巧。
- 五章:判断。Visual Basics 为用户提供了大量不同的结构用于控制程序功能,本章将会使用户成为这方面的专家。

第零章 做好准备工作

用户开始准备编程了吗？好，无论如何应走好第一步。本章最后会给用户一个 VB 工作程序。如果用户想编写有用的，好看的程序，至少必须读完第一章。下面是本章将要介绍的内容：

- 学习启动 Visual Basic 界面上的小器械。
- 认识 Visual Basic 界面上的小器械是作什么用的。
- 为一个小程序建立一个窗口。
- 往程序窗口里加入一些命令按钮及文本框。
- 写几条简单的程序代码。
- 建立一个可执行(EXE)文件并进行测试。
- 让用户美梦成真。

用户在阅读文章时，可能会遇上一些讲解得不太详细的术语。本章只是“(test the waters)测试水准”，用户在后续章中会学习更多关于这些术语的知识。

0.1 浏览 Visual Basic

象用户这样的 Windows 专家启动 Visual Basic 是不费脑筋的。只需在 Visual Basic 程序组(在程序管理器里)的 Visual Basic 图标下用鼠标点两下就行了。Visual Basic 启动后，自动开始一个新的项目文件，项目文件是用户用于最终建立一个程序的工作文件的集合，VB 假定用户想为用户程序建立一个窗口，所以为新的项目文件显示了一个窗体，窗体 在用户程序中成为窗口和对话框。图 0.1 显示了一个名叫 Form1 的新的空白窗体。

注意： 用户微机上的显示可能和图 0.1 略有不同，比如说，可能还显示了属性对话框。这更好。

用户程序可能只使用一个窗体，也可能使用很多个，每一窗体的名称是唯一的，缺省情况下，VB 为用户窗体 取一个唯一并有趣的名字，如 Form1, Form2, Form3...。用户应知道这点，用户在建立窗体 时应该重新命名，因为这将为用户编写程序代码时跟踪窗体和控件有帮助，窗体前以 frm 开始，后接描述性名字，比如：frmMain, frmEditor, frmAboutBox, 在后续章中用户会学到更多的命名协议。

0.1.1 工具框(Toolbox)

Visual Basic 启动时，除了 form1 外，用户还会看到 Toolbox, Toolbox 包含所有的控件，如命令按钮，复选框，列表框，诸如此类，用户可用于添加到窗体里。在 Visual Basic 程序里，控件在很多方面都是最重要的，所有用户程序界面里的器件，比如文本框，菜单，命令钮，滚