

第零章 内容简介

Visual Basic 3.0 是微软(Microsoft)公司推出的具有革命性的 Windows 程序设计环境。它提供在 Windows 图形界面内使用的一组可访问的、动态的开发工具集。使用 Visual Basic 可以开发像目前正在使用的 Windows 应用程序一样功能强大的应用程序。

在 2.0 和 3.0 版中,微软(Microsoft)对 Visual Basic 环境和语言作了许多改进。程序设计环境中提供了一些新的、重要的工具,包括属性(Properties)窗口,工具条(Toolbar)以及增强的调试工具集,用以加快应用程序的开发速度。另外,Visual Basic 还提供了新的控件,用户可以添加在应用程序中。现在,Visual Basic 比以前任何版本功能都强大,因为新添了很多控件、函数和结构,其中值得一提的是适应性强的 Variant 数据类型,以及对创建多文档接口(MDI)应用程序的完全支持。更重要的是,在 3.0 版本中,用户可以使用新的数据控件建立 Visual Basic 应用程序与数据库之间的联系,这些数据库可以在 Microsoft Access, dBASE, FoxPro 或 Paradox 中建立。使用这些新特性,可以在应用程序中访问已经存在的数据库中的数据(Visual Basic 还有一种专业版,其中提供了其它的新特性。读者可以在 2.0 或 3.0 的标准版及专业版中运行本书的程序范例,但本书中有关屏幕的内容都是基于标准版的)。

本书可引导读者逐步学习 Visual Basic 应用程序开发,同时还提供了 Visual Basic 程序实例供实际操作。如果读者已经熟悉 Basic 语言,本书将拓宽你的程序设计知识,也教会你熟悉 Visual Basic 的独特功能。如果读者不是程序设计人员,而是一个 Windows 的热心用户,那么也可以用本书中提供的 Visual Basic 应用程序,充实你的 Windows 程序库。

本书的第一部分以导引的方式介绍了 Visual Basic 环境中的工具。读者可以学会如何给你设计的新的应用程序开发 Windows 风格的界面,其中包括如何选用 Visual Basic 已有的各种控件,诸如命令按钮、文本框、图标、列表框、标签、复选框以及网格这些图形对象。然后,读者就会明白在这种面向对象的 Basic 语言中,如何通过设置属性及书写过程来构成应用程序。读者也会发现 Visual Basic 的事件驱动程序设计的独特性,然后可以使用一个庞大的程序设计工具库开始练习。

本书第二部分提供了一个完整的可执行的项目集合,读者可以查阅、运行以及按自己的需要修改。为了阅读方便,所有的应用程序也包含在本书的配套(另卖)软盘中。每章开始部分介绍应用程序以及运行和使用的方法。读者可以把每一个项目装入 Visual Basic 中,检查各个部分,然后运行。最后,读者可以使用 Visual Basic 的 Make EXE File 命令创建可执行文件,这样就可以在 Windows 环境下直接运行。

本书第二部分以实例导引的方式分别选择了各种程序设计并加以介绍,读者可根据自己的兴趣按任意次序学习。图 I-1 显示了各章节中介绍的应用程序的图标,这些程序覆盖了家庭及工作中广泛范围内的各种任务,这些实例将会帮助读者构思有想象力的 Visual Basic 应用程序。

附录中介绍了把 Access 数据库与 Visual Basic 3.0 应用程序联系起来的设计方法。使用这些新的数据控件及其相关的属性,读者将学会创建这样一种应用程序。在这种程序中,用

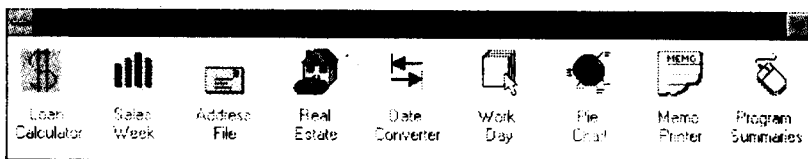


图 I-1 本书中提供的 Visual Basic 应用程序

户可以对现有的数据库进行检索、修改及添加。

安装 Visual Basic

Visual Basic 自带 Setup 程序,用户只需简单地按照屏幕上的提示即可进行安装。下面是在硬盘上安装 Visual Basic 的粗略步骤:

1. 启动 Windows 并把 Visual Basic 的第 1 张软盘插入驱动器中。在 Windows 的程序管理器(Program Manager)中,从 File 菜单选用 Run 命令。然后像图 I-2 所示,在 Command Line 方框中键入 A:SETUP 或 B:SETUP,按下 OK 按钮。

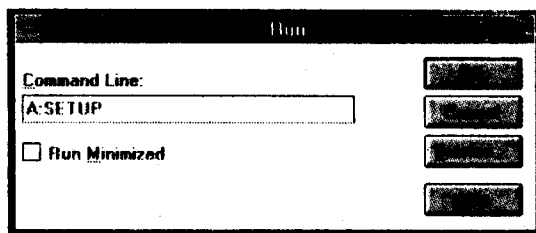


图 I-2 运行 Setup 程序

2. Setup 程序将进行一些初始化工作,然后显示一个对话框询问用户名。键入用户名后,出现的另一个对话框将让用户选择安装 Visual Basic 的盘号及目录名。如图 I-3,缺省的目录名是 C:\VB。无论使用缺省值或键入了新的路径名,Setup 程序将在硬盘中创建所需目录及子目录。按下 Continue 按钮继续进行安装操作。
3. 在图 I-4 所示的对话框中,用户可以选择全部或部分安装整个 Visual Basic 系统。此方框中将告诉用户,全部安装需要多少硬盘空间。如果硬盘空间足够,可以按下 Complete Installation 按钮;否则,按下 Custom Installation 按钮,在引出的对话框中又可以选择所需安装的部分。Setup 程序把文件顺序拷贝到硬盘中,在全部过程中,Setup 以图表显示安装的操作,在需换盘时提醒用户。整个安装程序需要一段时间。
4. 安装完毕,用户将在 Windows 的程序管理器(Program Manager)中发现一系列名为 Visual Basic 3.0 的程序组,如图 I-5 所示。启动 Visual Basic,可双击它的图标。
5. 要准备本书所要求的练习,读者可以在硬盘上另外创建一个目录,存放练习盘中的应用程序。目录名可定为 \VB3APPS,然后把练习盘中所有 .MAK, .FRM, .BAS, .ICO 文件拷贝到这个目录下。这些扩展名区别如下:

- ☐ .MAK 表示一个项目文件,其中是构成一个 Visual Basic 应用程序的文件列表。
- ☐ .FRM 表示一个窗体文件,决定了应用程序的窗口及对话框的外观和运行方式

(.FRM文件中也可包含代码,即 Basic 语言中的声明、过程及函数)。

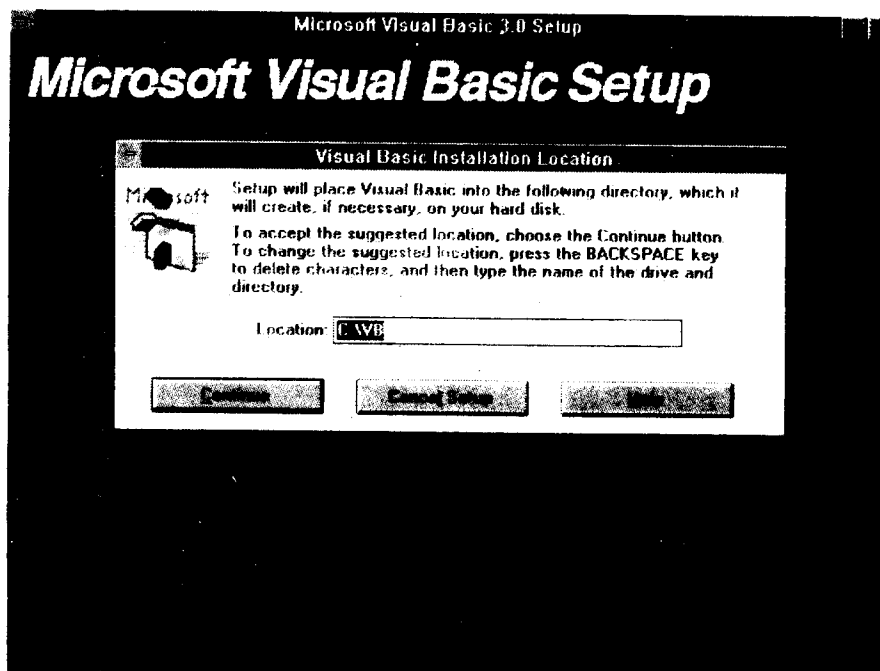


图 I-3 安装 Visual Basic 时,选择盘号及目录

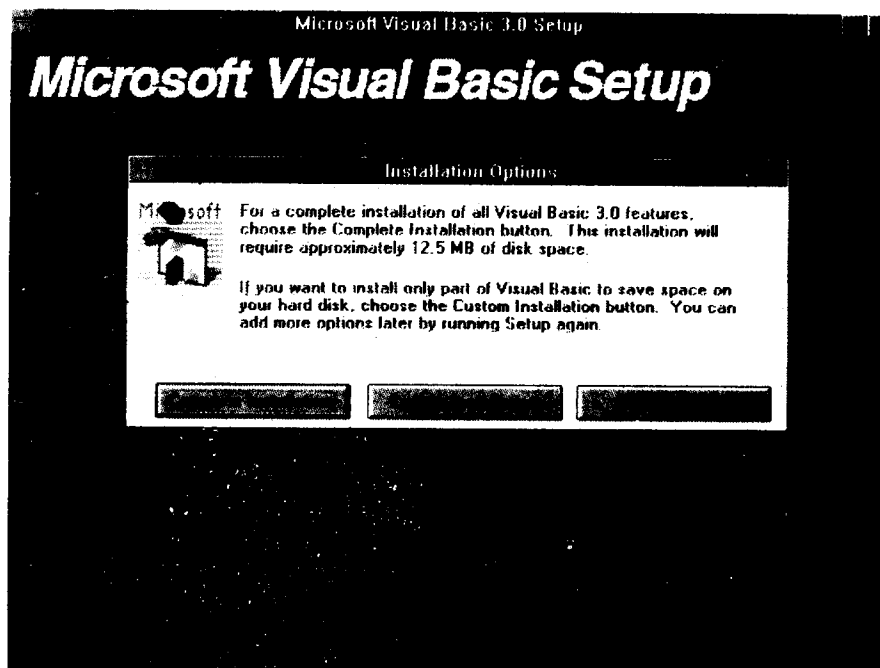


图 I-4 选择全部还是部分安装 Visual Basic 3.0

- ☐ .BAS 表示一个模块文件,其中包括应用程序的附加代码。
- ☐ .ICO 表示图标,可以作为应用程序的一个部分出现。

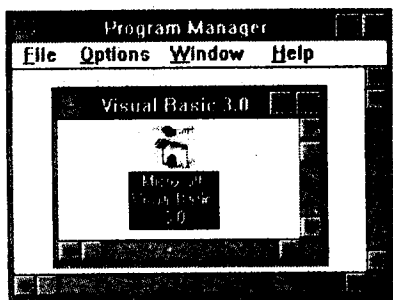


图 1-5 Visual Basic 的图标

关于练习盘的说明

练习盘中的项目以 Visual Basic 2.0 格式存放,这样读者可以在 Visual Basic 2.0 或 3.0 中打开或运行它们。把一个项目装入 Visual Basic 3.0 时,遇到每一个窗体,屏幕上都会显示下列信息:

'FORMNAME.FRM' has old file format. When saved it will be saved in newer format. (意思是:“FORMNAME.FRM”是旧的文件格式。再存储它时,将是新格式。)

按下 OK 按钮,就可以打开这个窗体。整个项目装入完毕,读者应马上从 File 菜单中选择 Save Project 命令,把项目以 Visual Basic 3.0 格式存储起来。

第一部分

Visual Basic 3.0 程序设计环境

第一章 Visual Basic 入门

Microsoft Visual Basic 是一个令人激动的、可以产生 Microsoft Windows 应用程序的新方法。相对传统程序设计语言来说,这是一个能帮助用户构造更复杂的 Windows 风格程序的、精心制作的开发环境。使用 Visual Basic 提供的工具,用户可以设计专业水平的 Windows 应用程序,具有直观性与操作一致性。这些程序可以包含 Windows 用户界面中任何一个常见的元素,例如:下拉式菜单、对话框、图标、工具条和鼠标控制功能。当使用传统程序设计语言时,构造这些元素需花费大量的时间和精力。

Visual Basic 是一个功能齐全的程序设计环境,提供了丰富的工具和强大的功能。这个环境的意义在于把 Windows 用户界面中的可视化元素作为标准控件,随时加入应用程序中,从而简化了开发过程。开始建立应用程序时,用户可以使用这些控件,在屏幕上直接设计、构造程序的界面。在大多数情况下,当应用程序的界面已经完成时,用户将开始书写 Basic 命令和过程,这和大家所知的传统程序设计工作是一样的。在开发过程中,用户可以随时修改界面:增加或移动控件或重新组织用户程序的外观。

有代表性的是用户使用 Visual Basic 的控件来描述准备在应用程序中实现的操作与选择。方便之处就在于这些控件的外观和运行是预定义的。请看一些例子:

- Visual Basic 的命令按钮控件,其外观与操作都与 Windows 其他命令按钮相似;用鼠标单击按钮,可看到按钮有被按下的效果。用户可以使用命令按钮描述程序能够执行的运算和其它操作。
- 文本框控件接收键盘的输入值。Visual Basic 文本框具有与其它 Windows 文本框相同的键盘编辑功能。

除了命令按钮和文本框外,Visual Basic 提供了标准控件的完整集合,包括列表、标签、复选框、单选钮、滚动条、框架、图片框和 Windows 环境中的其它功能。这些控件大大地简化了程序设计人员的工作,使他们可以把精力集中在重要的计算和数据处理过程上。Visual Basic 自动地提供用户界面元素,包括屏幕图形、键盘功能和鼠标操作。

在这章里,我们将开始使用 Visual Basic 的工具和资源。此章分为三个主要部分:

1. 关于 Visual Basic 开发环境的简单介绍。
2. Visual Basic 的项目范例,一个名为 Real Estate 的数据库程序。用此书附带的练习盘装上程序,运行 Real Estate 并理解它是如何工作之后,读者就可以开始研究应用程序部分了。
3. 用 Visual Basic 开发应用程序所需步骤的简介。在第一部分的其它章节中,读者还将学会有关这些步骤的更详细的内容。

启动 Visual Basic(如果用户还没有安装 Visual Basic,请按书中简介部分介绍的去做)。在 Windows 环境中,双击 Microsoft Visual Basic 3.0 的图标,几秒钟之后,屏幕上即出现 Visual Basic 开发环境,如图 1-1 所示。

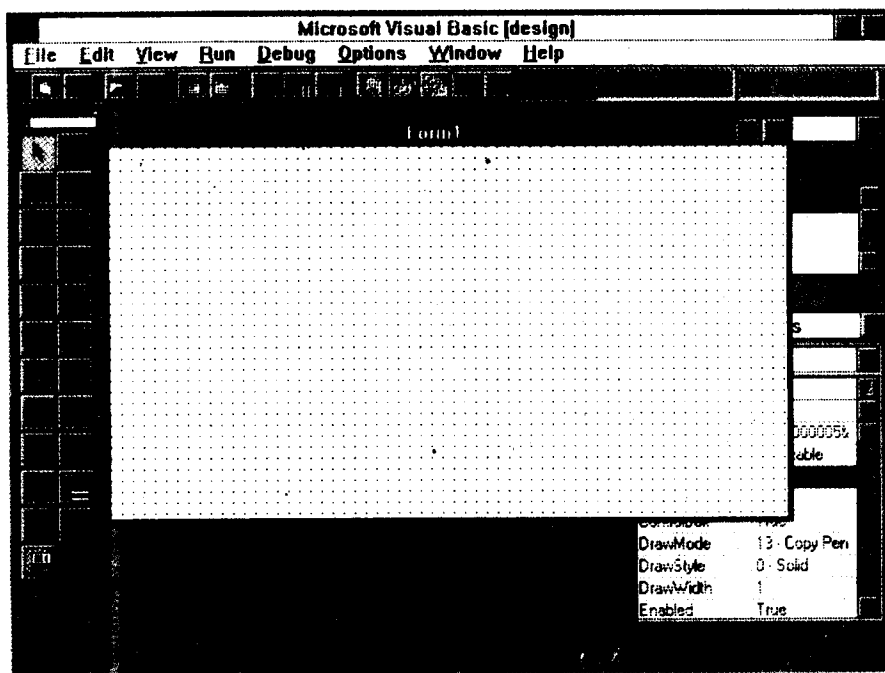


图 1-1 打开 Visual Basic 的屏幕

1.1 Visual Basic 开发环境

正如用户在屏幕第一行所看到的程序标题, Visual Basic 的初始模式为设计模式。这意味着环境已经具备了供用户定义新的应用程序界面的条件。开始时, 用户可能对各种交错的窗口和菜单感到迷惑, 但是很快就能轻松地工作, 就象熟练地使用字处理或报表程序一样。

1.1.1 窗体和项目

在屏幕中央, 留有大片空白, 这是一个名为 Form1 的窗口。按照 Visual Basic 特定的操作步骤, 这个新窗口将是用户设计新的应用程序的出发点。

用户构造 Visual Basic 应用程序的窗口称作窗体。只要用户应用程序需要, 窗体可以用来做不同的工作。举例来说, 窗体可以当成对话框。用户可以通过选择控件, 并把它们放置在窗体中央来设计一个对话框。另一方面, 窗体可以用于显示输出的文本、数字数据表或图形。开始时, 每个新的应用程序都只有一个窗体可用, 但是用户可以在设计程序的时候, 根据需要加入更多的窗体。

在 Visual Basic 2.0 和 3.0 中, 用户都可以添加一种特殊风格的窗体, 即 MDI 窗体。它允许用户为应用程序开发多文档接口。MDI 窗体起着控制 Windows 应用程序的作用, 允许用户同时创立和打开多文档窗口。第九章详细介绍了此种功能。

用 Visual Basic, 用户可以一次完成对应用程序的设计。用户添加在应用程序中的每个窗体都各自保存在磁盘文件中。在进行开发时, 一个给定的应用程序可能因此包含许多不

同的磁盘的文件(举例来说,读者将在本章后面看到的一个程序就是由六个磁盘文件组成的)。组成应用程序的所有文件合称作项目。Visual Basic 可以记住给定项目的文件,并可在一次操作中打开所有这些文件。

在本章里,我们将主要介绍组成 Visual Basic 环境的主要工具的一般特性:

- 工具框最初定位在屏幕左边,包括各种用户可以选择利用的可视化控件。
- 属性窗口提供的工具可供用户定义和修改添加到应用程序中的窗体和控件的初始特性。
- 项目窗口,最初出现在屏幕右边(部分隐藏在 Form1 后),列出当前创建的应用程序的组成部分。
- Visual Basic 菜单系统和工具条为程序开发提供了各种各样的命令和快捷键。

用户可以通过执行一些 Visual Basic 开发环境中介绍的练习来研究这些工具的功能。



图 1-2 工具框

1.1.2 工具框与属性框

如图 1-2 图示,工具框包括两列图标。这些图标代表了用户放置到应用程序中的控件。用户可以把其中任意多个控件加入到窗体中,并按照应用程序设计需要对它们加以安排。

呈现在工具框中的每一个控件都有其特殊的特性。学会如何成功地使用这些控件,是用户开发应用程序之前的主要任务之一。现在,简单地看一下工具框中的每一个图标并猜想它的用途。如果用户曾经使用过 Windows,这些控件就是很熟悉的了;大多数图标都与 Windows 典型阶段中出现的图标相似。

以下是对工具框中控件的简单介绍,依照从上到下的逐行顺序:

- 在工具框左上角的指针图标,实际上不是一个控件,它代表在窗体中操纵控件的鼠标。
- 右上角的图片框控件可以作为应用程序的图标或其图形显示元素。
- 标签控件(工具框中表示为 A)是具有代表性的短文标签,用于定义对话框的元素。用户还可用标签控件来显示长的文本块。
- 文本框 控件(表示为一个内有黑体 ab 字符和一个垂直箭头的小方框),是一个从键盘接收文本的输入框。
- 框架控件(表示为一个小方框,标题的位置上为 XYZ 字符)用于把窗体中其它的控件组合起来。
- 命令按钮(工具框中的一个带有阴影的小长方形按钮)是一个控件。用户可以敲击命令钮,执行一种特殊的运算或选择一个应用程序中的选项。
- 复选框(表示为带有 X 的正方形)表示一个可以打开或关闭的选项。对话框中可以包括复选框列表,其中的每一个选项都能打开或关闭,不受其它选项的影响。
- 单选钮(带有黑点的圆圈)是另一个显示开/关的选项。一组单选钮通常代表互相排斥的选项。在同一时间内,只能有一个为打开状态,其它选项都为关闭状态。
- 组合框是一个带有下拉式列表的文本框。用户可以通过键盘在框中直接显示文本,也可从下拉式列表中选择文本输入。
- 列表框是文本选项列表,其中任何一个都可被选为输入项。
- 水平滚动条是物理位置或数值量的控件,它与最小和最大值有关。在滚动条的最左和最右边有箭头标志,方框可以通过滚动条向前或向后移动。一个具有代表性的应用就是框中文本的滚动显示。
- 垂直滚动条包括顶端和底端的箭头标志,滚动框可以上下移动。
- 计时器控件(钟表图标)表示应用程序中的时间事件。
- 驱动器列表框控件(驱动器图标)允许用户显示有效驱动器和启动一个新的驱动器。
- 目录列表框控件用来显示磁盘上的目录体系。
- 文件列表框(列出已存在的文件)显示目录中的文件名。
- 形状控件(在工具框中表示为一组互相重叠的几何形状)在窗体中表现为实心或空心的形状。用户可以用这个控件选择各种不同的形状,包括正方形、长方形、椭圆和圆。
- 直线控件(表示为一条对角线)在窗体可以显示垂直线段、水平线段、斜线段。这些控件可用来把一个窗体分为几部分或把一组控件和另一个分开。
- 图像控件(表示为一幅框起来的图画)与图片框功能相似,但是有更好的性能和存储效率。
- 数据控件允许用户使用 Visual Basic 应用程序中已经存在的基本数据(如何使用数据控件,可参看附录)。
- 通用对话框控件为一些通用运算,如打开、保存、打印文件和访问颜色、字体等提供了标准对话框。
- 网格控件(工具条第二列的最后一个图标)显示为二维排列的一组方格,与 Microsoft Excel 程序中的工作表相似,用户可以用这个控件在窗体中显示数据表或者访问输入

数据。

- OLE 控件(对象链接与嵌入控件,第一列中的最后一个图标)允许用户把对象链接和嵌入到 Visual Basic 的应用程序中。OLE 是一种在不同应用程序间共享信息和图形的 Windows 技术。

稍后,读者将看到应用程序的举例,从而了解一些控件是如何工作的。以下简单的练习将告诉读者把控件放入窗体是多么简便易行。

练习:把控件放置在窗体上

正如用户所看到的,在 Visual Basic 屏幕上的一个活动窗体最初被命名为 Form1。现在可以练习一下通过在窗体中放置控件来设计应用程序的对话框。开始执行下列步骤:

1. 把鼠标移动到工具框的命令按钮上(右列上端向下数第三图标)。
2. 双击鼠标左键。一个名为 Command1 的按钮立即出现在 Form1 的中央,如图 1-3 所示。命令按钮是三维的,非常直观。注意命令按钮周围有黑色的小方框,被称作尺寸把柄。它可以根据随后的动作改变当前选择的大小。
3. 把 Command 按钮拖到 Form1 的左上角:使鼠标对准控件,按下左键。当用户拖动图标时,将会看到控件移向新的位置。如果仔细观察,会发现这种移动是在点状网格背景上沿水平和垂直方向的小步前进。当用户拖动控件时,定位于窗体上工具条右边的一对数字快速地变化。这些数字给出了与当前窗体左上角相对的控件位置。当用户放开鼠标,重定位就完成了。

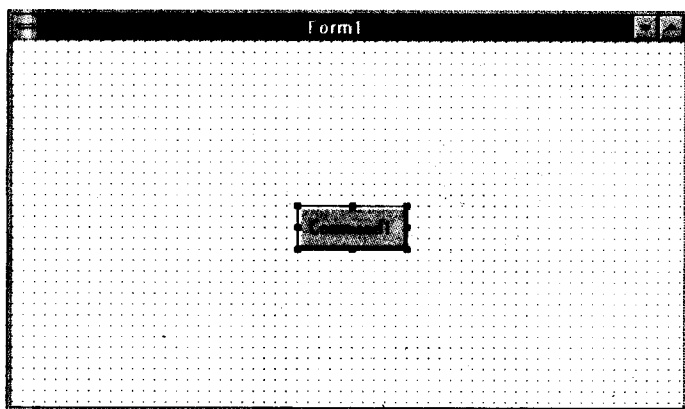


图 1-3 在窗中放置一个命令按钮

4. 减小命令按钮的尺寸。把鼠标定位在尺寸把柄的右下角上,鼠标变成一个双头对角的箭头。按下鼠标的左键,小心地把箭头的一端向左移,另一端向上移(请注意:所引起的变化发生在屏幕右上角附近位置坐标右的一对数值中,它们显示出控件的宽度和高度)。放开鼠标按钮,重定尺寸的操作就完成了。
5. 在 Form1 窗体的任何一个地方单击鼠标,使命令按钮控件无效。

现在,用户已经在窗体中放置了第一个控件。如果用户实际上在设计一个应用程序,这个命令按钮最终代表着程序所需执行的特殊操作,例如保存记录、计算一个数值、从屏幕

上清除数值,显示新的选项对话框或从当前程序中退出等。在这一章的应用程序例子中,读者将看到代表这些操作的按钮。在运行程序时,需要执行其中一种操作,只要简单地单击适当的命令按钮即可。

当开始往应用程序中添加控件时,用户希望能够指定每一个控件的功能——它的外观和定位、名字和行为。这些以及其它功能在 Visual Basic 中称为属性。每一个属性有一个名字和缺省值。有趣的是,用户已经在 Form1 中改变了命令按钮的四个属性:重新定义了控件的大小,改变了 Left 和 Top 属性。用户可以直接用鼠标改变这些特殊的属性。其它属性通过一个称作属性(Properties)窗口的 Visual Basic 的重要特性来控制。

1.1.3 控件及其属性

工具框中的每一个控件都有自己的属性集。另外,应用程序的窗体也有属性。要修改控件或窗体的外观、定位或行为,用户需要改变一组适当的属性。一些属性,如 Top 和 Left 是大多数或全部控件的通用属性,而另外一些有特殊用途的属性则只与特殊控件有关。属性窗口提供给用户简单的技术来操纵处理这些属性。用户可以利用属性窗口执行以下任何一种操作:

- 显示适用于给定控件或窗体的属性表。
- 确认选定控件或窗体的当前特定属性集。
- 显示有效属性集表。
- 改变选定窗体或控件的属性列集。

属性窗口最初定位在 Visual Basic 屏幕的右下角。用户只需简单地按下 F4 键就可激活这个窗口,并用鼠标重新定义窗口的位置和大小,就象对待其它窗口一样。图 1-4 显示的是属性窗口,带有对应 Form1 的属性表。

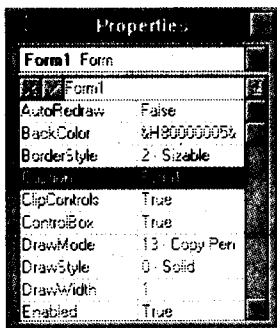


图 1-4 属性窗口

属性窗口分两列表格显示。左列显示对应当前选定的窗体或控件的属性,右列显示每一个属性的设置。表格上面为设置框,当前显示为 Form1。每个属性都有一组确定的设置值,显示依附于此设置框的下拉式列表中。另外,设置框还可以接收键盘输入作为设置值。在设置框上是对象框,此框的下拉式列表中显示了当前窗体的全部控件。在用户选择了一个控件和属性后,就可以利用设置框改变选定的属性和设置值。每一个给定控件的属性都有一个缺省

值——在操作中保留的设置值,除非用户要明确地改变它。

正如读者已经看到的,定义一个给定控件或窗体大小和位置的四个属性值已经在 Visual Basic 菜单之下的工具条右边显示出来了。这四个属性被恰当地命名为 Left, Top, Width 和 Height。两个显示属性设置值的框,如图 1-5 所示,命名为位置框和尺寸框。

- 位置框显示选定控件或窗体的位置坐标,即 Left 和 Top 属性设置值。对一个控件来说,坐标表示的位置与包含窗体的左上角位置有关;对一个窗体来说,坐标与屏幕的左上角位置有关。在这两种情况中,第一个坐标都是水平位置,第二个都是垂直位置。
- 尺寸框包括选定控件或窗体的大小,即宽度和高度设置值。宽度是第一维,高度是第二维。

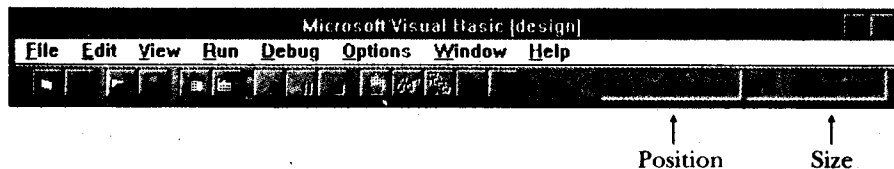


图 1-5 位置框和尺寸框

练习:使用属性窗口

在看过关于属性窗口的简单介绍之后,用户现在可以改变一下已经放置在 Form1 中的命令按钮的两个属性。首先,在按钮里显示一个新的名字(重新激活按钮时,当前显示为缺省的 Command1)定义这个名字的属性叫作 Caption。其次,用斜体显示这个名字。控制显示特性的属性称为 FontItalic。改变 Caption 属性需要用户输入新的文字,而 FontItalic 属性则有固定的有效设置表。用户可以从设置框的下拉式列表中选择设置。

依照下列步骤,改变上述两个属性:

1. 单击 Command1 命令按钮来选定控件。按钮周围出现尺寸把柄。
2. 按下 F4 功能键激活属性窗口。最初被选定的属性是 Caption。设置框显示命令按钮的 Caption 属性缺省值为 Command1。当选定控件时,如要改变 Caption 属性值,只需简单地从键盘上输入新文字。
3. 输入 Start 作为命令按钮的 Caption 属性值,然后按下回车键完成此操作。这个新名字就出现在设置框和命令按钮里。
4. 利用垂直滚动条上下移动属性网格,浏览一下命令按钮的属性。然后,单击 FontItalic 属性。这时,用户的屏幕如图 1-6 所示。
5. 单击设置框右边的下箭头图标,下拉一张 FontItalic 属性的有效设置值表。这张表包括两个设置值:False 和 True。当前设置为 False,单击 True,当前设置值立刻改为斜体字。

对于添加到应用中的许多新的控件来说,用户只需在设计模式时,对其中几个改变缺省设置值。然后,用户将会看到程序在运行时,作为对发生事件的响应,会自动改变属性值。

到现在为止,用户已经操作过 Visual Basic 环境中的几个元素:工具框、属性窗口、窗体和控件。现在,我们来研究一下至今还隐藏在 Form1 后面的窗口——项目窗口。

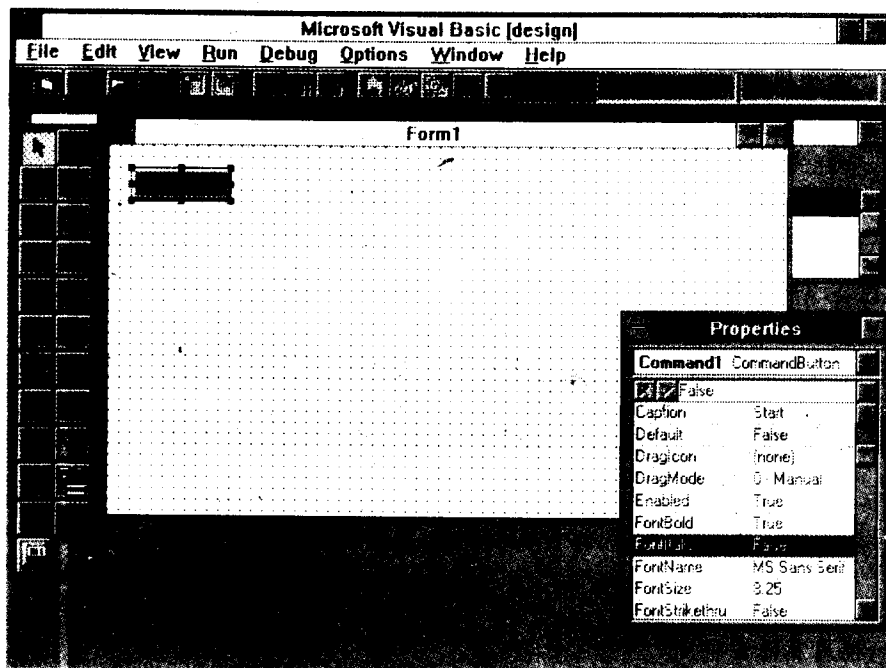


图 1-6 使用属性窗口

1.1.4 项目窗口

项目窗口列出了当前创建的应用程序的所有文件。请按照下列步骤显示项目窗口。

1. 单击 Visual Basic 菜单条的 Windows, 出现下拉式菜单, 上面列出了各种 Visual Basic 工作窗口。

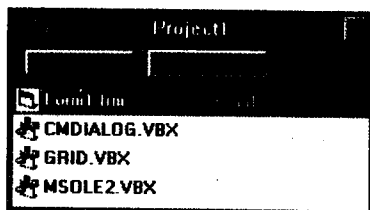


图 1-7 项目窗口

2. 单击 Project, 显示项目窗口。
3. 把窗口的底边向下拉一点, 这样用户就可以看到窗口包括的全部文件。

完成这一操作后, 屏幕上的项目窗口如图 1-7 所示。正如读者所看到的, 当前项目的缺省值命名为 Project1, 包括四个文件, 分别为 Form.frm, CMDIALOG.VBX, GRID.VBX 和 MSOLE2.VBX。窗口内分三列显示文件信息: 第一列显示代表文件类型的图标。图标右边

是保存文件的文件名,如果文件没有保存在盘上,则由缺省值显示。第三列显示了项目表中所有窗体的 Name 属性的设置值。

项目包括窗体、模块、定制控件。

- 就象用户所看到的一样,窗体是应用程序的可视化部分,可以在对话框或窗口中显示文本、数据或图形。
- 模块是用来保存代码的一段程序,代码即 Visual Basic 程序的声明、结构和过程。
- 定制控件定义的工具框控件并不构造到默认的 Visual Basic 环境中。在 Visual Basic 3.0(标准版)中,通用对话框、网格控件、对象链接与嵌入控件就是作为定制控件实现的。这些控件的属性在 CMDIALOG.VBX, GRID.VBX 和 MSOLE2.VBX 中有定义。

这三个文件由 Visual Basic 安装程序拷贝到 \WINDOWS\SYSTEM 目录下。

如果应用程序需要使用这些定制控件,VBX 文件则必须包含在应用程序的项目窗口中。如读者稍后将看到的一样,当应用程序不使用这些控件时,用户可以从项目列表中删除这些 VBX 文件。

窗体和模块

这是两个可供用户书写和保存应用程序代码的地方。

- 一个窗体通常包含了相应控件的过程。有两个窗口可用来显示窗体的组成部分:就象用户所见到的,窗体的窗口中显示了已经放入的控件。代码窗口显示用户为窗体编写的声明和过程。对开发过程来说,代码窗口就相当于全功能的文本编辑器。
- 一个模块中的过程可以用在多个窗体中。与窗体的代码窗口相似,模块窗口也可用作输入和修改代码的编辑器。

读者以后还会发现此代码区更多的内容。现在,让我们主要看一下项目窗口中用于窗体和模块的工具。首先,我们注意到项目窗口顶行有两个命令按钮:View Form 和 View Code。利用它们,用户就可以立即访问文件部分,正如读者在以下练习中学到的一样。

练习:使用项目窗口

View Form 和 View Code 按钮与当前项目窗口中的选定文件相对应。用户可以用鼠标单击文件名或移动上下箭头键来选定文件。

现在,确认选定了 Form1,就可以接着按下列步骤执行:

1. 单击 View Code,显示这个窗体的代码窗口。代码窗口如图 1-8 所示。因为用户还没有为这个窗体书写任何代码,所以代码窗口是空的。请注意窗口标题条下的标为 Object 和 Proc 的框;在开发程序时,它们可以帮助用户定位代码段的各个不同部分。
2. 按下 Alt+F4,关闭代码窗口。
3. 下拉 File 菜单,选择 New Module 命令。与之相应,Visual Basic 为当前项目添加一个模块文件并打开模块的代码窗口,此窗口与窗体的代码窗口相似。
4. 按下 Alt+F4,关闭模块的代码窗口。

Visual Basic 的项目文件采用不同的扩展名:

- 窗体文件的扩展名为 FRM。
- 模块文件的扩展名为 BAS。

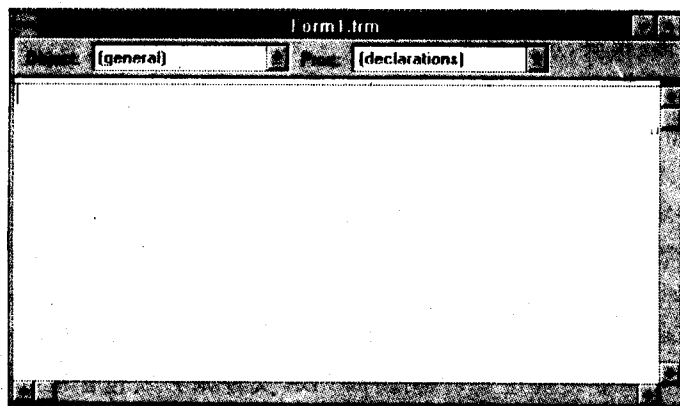


图 1-8 Form1 代码窗口

- 特殊的项目文件的扩展名为 MAK。这种文件可以记往应用程序的不同部分。
- 定制控件的扩展名为 VBX。

正如用户所看到的,文件名列表随着添加到应用程序中的窗体和模块的增多而加长。(用户刚刚加入到项目中的模块已经列入项目窗口,名为 Module1.bas)。用户可以利用 File 菜单中的命令来改变程序中的文件。

1.1.5 Visual Basic 的菜单与工具条

Visual Basic 菜单系统提供了用户在开发应用程序中所用到的工具。在系统中,有八个下拉式菜单:File,Edit,View,Run,Debug,Options,Windows 和 Help,如图 1-9 到 1-16 所示。

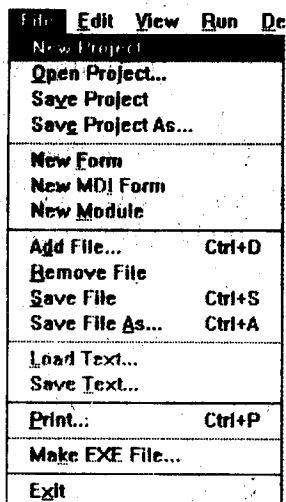


图 1-9 File 菜单

象在 Windows 的其他应用程序中一样,用户可以用鼠标单击菜单名或按下 Alt 键,再按菜单名的第一个字母来下拉菜单。举例来说,按下 Alt 键,再按 F,显示出 File 菜单。一旦菜单显示出来,用户可以用鼠标单击命令名或从键盘敲入适当的值来选定一个命令。另外,有一些菜单命令可以用快捷键执行而无需下拉菜单。

下面是每个菜单的命令和工具的小结:

- File 菜单中命令的执行与项目的文件有关或者无关。举例来说,New Project 可以使用户开始一个新的项目。Open Project 命令可以用一次操作打开一个已经存在项目的所有文件。同样的,Save Project 命令能够把当前项目的所有文件保存到盘上(使用 Save Project As 可以为当前项目起一新的名字)。File 菜单也包括了为当前应用程序添加新窗体和模块的命令(New Form, New MDI Form 和 New Module)。在以后的练习中,用户将会用到这些命令。用户也可以把一个文件从项目中移出(Remove File),在盘上单独地保存窗体和模块(Save File 或 Save File as),并且可以把文本文件装入项目或使项目部分作为文本文件存储(Load Text 或 Save Text)。Print 命令允许用户打印一个打开文件的窗体和代码。Make EXE File 命令由当前的应用程序部分产生唯一的 EXE 文件。当用户开始打开 MAK 文件的现存项目时,File 菜单显示一张最近使用表,表中包含了用户已经打开的至少四个项目的名字。用户只需简单地在表中选定文件名,就可快速地重新打开文件。最后,用 Exit 命令关闭 Visual Basic 窗口,结束当前的程序操作。

Edit View Run Debug	
Undo	Ctrl+Z
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Paste Link	
Delete	Del
Find...	Ctrl+F
Find Next	F3
Find Previous	Shift+F3
Replace...	Ctrl+R
Bring to Front	Ctrl+Z
Send to Back	Ctrl+Y
Align to Grid	

图 1-10 Edit 菜单

- Edit 菜单包括了各种各样的用于书写 Basic 程序的工具。Undo 命令允许用户在一次错误的操作后,重新加载原先的文本。用户可以使用 Paste Link 命令在 Visual Basic 窗体和其它应用程序的文档之间建立一种动态链接。对于定位指定文本项或执行查找与替换操作,Find 和 Replace 命令是非常有效的。
- View 菜单提供了查看代码、产生新过程、滚动过程的命令,还有一个命令项(Toolbar)用于显示或隐藏 Visual Basic 的工具条。