

© 高等学校计算机基础教材



computer

Visual Basic 程序设计

主 编 潘地林

副主编 黄洪超 王永国



高等教育出版社

Higher Education Press

前 言

Visual Basic 是一种易学易用、功能强大的高级编程语言,在实际编程中得到广泛使用,适合作为高等学校程序设计语言的入门语言。随着高校招生规模的不断扩大与计算机基础课程教学改革的不断深入,基础差、就业难、课时少、用不上的矛盾日益突出,同时市场上适合初学者学习、方便教学兼顾等级考试且资源配套、实用性较强的特色教材也很少。为此,安徽省教育厅有关部门,从 2004 年 7 月开始着手,结合高等学校计算机基础课程教学改革与课程建设,组织省内部分高等学校有关学科的老师,进行了几门重点课程的试题库建设工作。目前,题库建设已初步完成,为相关教材和实训指导书的编写打下良好的基础。

本书紧扣 2005 年安徽省教育厅制定的《全国高等学校(安徽考区)计算机基础教育教学(考试)大纲》,并参考了全国计算机等级考试和部分省市计算机水平考试大纲,由长期从事计算机基础教学与考试题库建设的“Visual Basic 程序设计”项目组的一线教师编写,是一本程序设计的入门教材,在内容编排、叙述表达、课后习题和实验题选择等方面做了精心设计,目的是让非计算机专业的学生了解和掌握程序设计的基本原理和方法,从而具备基本的应用程序开发能力。

本书编写的指导思想是,吸取已有教材的优点,结合教学实践、试题库建设经验,努力把知识点融入到实际项目的开发中去,立足于“理论够用,操作熟练,重在实践”的要求,做到通俗易懂,循序渐进。本书每一章后面都按照考试大纲的要求附有一定数量的典型习题,供读者练习,以便巩固所学知识。

全书共 8 章,主要内容包括:Visual Basic 程序设计概论、Visual Basic 语言基础、用户界面设计、过程与变量的作用域、菜单设计、文件管理、数据库编程及多媒体程序设计等。

本书配套的《Visual Basic 程序设计实训与考试指导》(附光盘)辅导书,根据主教材知识点,精心设计和安排了 13 个实训,每个实训都给出了具体的操作步骤、详细的分析和参考源代码,并给出了主教材中的全部习题解答,便于读者理解和巩固所学内容。在配套光盘中给出了本书所有习题的程序代码,便于读者上机调试。此外,光盘中还包括题库练习系统和 Visual Basic 上机模拟考试系统。试题库练习系统内嵌了经过多年研制的大型题库,题库中收集了近 1 000 道试题,其中涵盖了 Visual Basic 语言的各主要知识点,既可固定组卷,也可随机组卷,还可以计算机自动评分,方便读者学习和考前复习自测。

另外,为了适应新的教学需求,方便学校统一测试,提高教学质量,我们还研发了整套无纸化考试系统,该系统包括组卷、上机考试、阅卷、服务器设置、考试监控等模块。

本书由潘地林主编,黄洪超、王永国任副主编。具体分工为:潘地林编写第 1 章、第 2 章;黄洪超编写第 3 章;王永国编写第 4 章;汪采萍、马春编写第 5 章、第 6 章;王小立编写第 7 章;陈清华编写第 8 章。参加本书部分内容和题库编写的还有黄玉春、陈万顺、杨飞等。全

II 前言

书由潘地林统稿。

本书建议授课学时为 32~40 学时,上机实训 24~30 学时,课外上机实训 30 学时。

本书既可以作为高等学校本科及专科学生 Visual Basic 语言程序设计课程的教材,也可以作为教师教学和读者自学的参考书,还可以供各类参加等级考试的人员参考。

本书在出版过程中,一直得到安徽省教育厅相关领导的指导与支持,教科所袁振发同志、合肥工业大学孙家启教授共同主持并参与了试题库的建设工作,高等教育出版社相关领导及策划/编辑,在此一并表示感谢。

由于编者水平有限,书中不足之处,敬请广大读者批评指正。

编 者

2006 年 4 月



Visual Basic 程序设计概论

【内容概述】

本章简要介绍了 Visual Basic 程序设计的基本概念和入门知识,主要包括 Visual Basic 特色、系统的安装与启动、集成开发环境、程序编写的基本概念和基本步骤以及程序的调试方法等。

【教学目标】

- 了解 Visual Basic 程序设计语言的特点,熟悉 Visual Basic 的集成开发环境。
- 了解 Visual Basic 编程的关键概念,掌握编程的一般步骤。
- 了解程序调试的基本概念和方法。
- 了解生成可执行文件和制作安装盘的方法和步骤。

1.1 概 述

Visual Basic(简称 VB)是美国微软公司推出的在 Windows 环境下使用的应用软件开发系统,是近年来得到迅速推广和应用的一种计算机高级语言,其特点是适合于面向对象程序设计。使用 VB 可以既快又简单地开发 Windows 环境下的应用软件。

Visual 意为“可视化的”,在此是指一种开发图形用户界面(GUI)的方法。在 VB 中引入了窗体和对象的概念,窗体和每个控件都由若干个属性来控制其外观形状、工作方法。这样,在使用 VB 编程时就不须编写大量的代码去描述用户界面元素的外观和位置,而只要把预先建立的控件添加到用户窗体上即可。BASIC 是 Beginner's All-purpose Symbolic Instruction Code(初学者通用符号指令代码)的缩写,它是一种计算机高级编程语言。Visual Basic 是一种基于 BASIC 的可视化程序设计语言,或者说它是 BASIC 语言的可视化版本。Visual Basic 一方面继承了 BASIC 程序设计语言所具有简单易用的特点,另一方面在其编程系统中采用了面向对象、事件驱动的编程机制,用一种巧妙的方法把 Windows 的编程复杂性封装起来,提供了一种所见即所得的可视化程序设计方法。专业人员可以用 VB 实现其他任何 Windows 编程语言的功能,而初学者只要掌握几个关键词就可以建立实用的应用程序。对于以前没学过任何一种计算机编程语言的人来说,VB 的语法是最容易被接受的。

本教材是面向编程初学者的 Visual Basic 入门教程。学完本书后,读者应能够熟悉 Visual



Basic 6.0 的编程环境,掌握 Visual Basic 语言的基本语法和编程技术以及基本控件的使用方法,初步具备开发简单 Visual Basic 应用程序的能力,为今后继续深入学习 Visual Basic 编程和在自己专业中实际应用计算机技术打下必需的基础。

目前,很多常用的应用软件都内嵌了 VBA(Visual Basic for Applications, Visual Basic 的一个子集)作为二次开发工具,例如 Office 2000、AutoCAD 2000 等。VBScript(Visual Basic 的另一个子集)是广泛使用的脚本语言,被广泛应用于 Web 程序的开发过程中,使用 VBScript 并结合 HTML 代码可以快速完成 Web 应用程序的开发工作。同样,Visual Basic 对数值计算、数据库应用、图形图像处理、多媒体和通信技术等都具备强大的开发应用功能,能满足各行各业应用软件开发的需求。

1.2 Visual Basic 的发展及编程特色

Visual Basic 最早是由微软公司在 1991 年推出的,刚推出的 Visual Basic 存在一些缺陷,功能也相对较少。经过微软公司的不断努力,于 1992 年、1993 年、1995 年和 1997 年相继推出了 VB 2.0、VB 3.0、VB 4.0 和 VB 5.0 这 4 个版本。在 1998 年秋季,随着 Windows 98 的发行,微软公司又推出了功能更强、更完善的 VB 6.0 版本,该版本在创建自定义控件、对数据库的访问以及对 Internet 的访问等方面都得到进一步加强、完善和提高。本书主要介绍 VB 6.0 中文版。

VB 6.0 包括 3 个版本,分别为学习版(Learning)、专业版(Professional)和企业版(Enterprise)。

① 学习版:VB 6.0 的基本版,包括所有的内部控件、连接网络、数据绑定等控件,适用于初学者。

② 专业版:主要针对计算机专业开发人员,除了具有学习版的全部功能外,还包括 ActiveX 和 Internet 控件开发工具之类的高级特性。

③ 企业版:VB 6.0 的最强版本,除具有专业版的全部功能外,还包括一些特殊的工具。

这些版本是在相同的基础上建立起来的,以满足不同层次的用户需要,对大多数用户来说,专业版已经可以满足需要。本书的各章节适用于 Visual Basic 6.0 专业版和企业版(中文版)。

Visual Basic 是从 BASIC 语言发展而来的,对于开发 Windows 应用程序而言,VB 是目前所有程序开发语言中最简单、最容易使用的编程语言。作为计算机程序设计语言,VB 具有以下编程优势和特点。

1. 可视化的程序设计

Visual Basic 率先采用了可视化的程序设计方法。利用系统提供的大量可视化控件,可以方便地以可视化方式直接绘制用户图形界面,并可直观地调整界面的风格和样式,从而克服了以前必须用大量代码去描述界面元素的外观和位置的传统编程模式。

用 Visual Basic 开发程序,就像搭积木盖房子一样,系统提供的可视化控件如同盖房子要用的各种构件,通过不同控件的搭配组合,可方便地构造出所需的应用程序。

2. 面向对象的程序设计思想

Visual Basic 采用了面向对象的程序设计思想。面向对象的程序设计是伴随 Windows 图形界面的诞生而产生的一种新的程序设计思想,与传统程序设计有着较大的区别。所谓“对象”就是现实生活中的可见实体。同样,在 Visual Basic 中,用来构成用户图形界面的可视化控件,也可视为是一个对象。不同的对象,在程序中所赋给它的功能是不同的。例如,在图形界面上有两个



命令按钮,一个用来实现数据的统计计算,另一个用来实现数据的打印,这两个按钮就可以视为是两个不同的对象。为了实现这两个对象各自不同的功能,应该分别针对这两个对象编写程序代码,这种编程的思想和方法即为“面向对象的程序设计”。

3. 事件驱动的编程机制

Visual Basic 采用了事件驱动的编程机制。在 Visual Basic 中,对象与程序代码通过事件及事件过程来联系,对象的活跃性则通过它对事件的敏感性来体现。一个对象(控件)通常可以感知和接收多个不同类型的事件,每个事件均能驱动一段程序(事件过程),完成对象响应事件的工作,从而实现一个预编程的功能。例如,命令按钮是编程常用的一个对象,若用鼠标在其上面单击,便会在该对象上产生一个鼠标单击事件(Click)。与此同时,Visual Basic 系统就会自动调用执行命令按钮对象的 Click 事件过程,从而实现事件驱动的功能。

Visual Basic 编程没有明显的主程序概念,程序员所要做的就是面向不同的对象分别编写它们的事件过程。若希望某对象在某事件发生后能做出响应,只需在该对象的事件过程中编写相应的程序代码即可。整个 VB 应用程序就是由这些彼此相互独立的事件过程构成,事件过程的执行与否以及执行的顺序由操作时用户所引发的事件来决定,若用户未触发任何事件,则系统将处于等待状态。

4. 高度的可扩充性

Visual Basic 是一种具有高度可扩充性的语言,除自身具有强大的功能外,还为用户扩充其功能提供了多种途径,这主要体现在以下 3 个方面。

① 支持第三方软件开发商为其开发的可视化控件:Visual Basic 除自带许多功能强大、实用的可视化控件以外,还支持第三方软件开发商为扩充其功能而开发的可视化控件,这些可视化控件对应的文件扩展名为 OCX。只要拥有控件的 OCX 文件,就可将其加入到 VB 系统中,从而大大增强了 VB 的编程功能。

② 支持访问动态链接库(Dynamic Link Library,DLL):Visual Basic 在对硬件的控制和低级操作等方面功能相对较弱,为此,VB 提供了访问动态链接库的功能。可以利用其他语言(如 Visual C++ 语言),将需要实现的功能编译成动态链接库,然后提供给 VB 调用。

③ 支持访问应用程序接口(API):应用程序接口(Application Programming Interface,API)是 Windows 环境中可供任何 Windows 应用程序访问和调用的一组函数。在微软的 Windows 操作系统中,包含了 1 000 多个功能强大、经过严格测试的 API 函数,供程序开发人员编程时直接调用。Visual Basic 提供了访问和调用这些 API 函数的能力,充分利用这些 API 函数,可大大增强 VB 的编程能力,并可实现一些用 VB 语言本身不能实现的特殊功能。

5. 支持大型数据库的连接与存取操作

Visual Basic 提供了强大的数据库管理和存取操作的能力,利用数据控件可以访问多种数据库系统。Visual Basic 6.0 中新增加了功能强大的 ADO(ActiveX Data Object)控件,利用它可以轻松地开发出各种大型的客户/服务器应用程序。

另外,Visual Basic 还支持动态数据交换、对象的链接与嵌入等新型的编程技术。

1.3 Visual Basic 的安装与启动

Visual Basic 6.0 系统程序在发布时是经过压缩存储在光盘上的,使用前必须先将这些系统



文件解压、复制到硬盘上,这一过程通常称为安装,其具体的解压和复制工作由系统提供的相应安装程序(一般为 Setup.exe)完成。

1.3.1 Visual Basic 的运行环境

Visual Basic 6.0 是一个 32 位的应用程序开发工具,其运行环境必须是 Microsoft Windows 95/98/Me/2000/XP 或 Windows NT。硬件要求 Pentium 或更高的处理器,或任何运行 Microsoft Windows NT Workstation 的 Alpha 处理器。企业版的安装需要 140 MB 的硬盘空间,其帮助系统约需要 67 MB 的硬盘空间。

1.3.2 Visual Basic 的安装与卸载

运行 Visual Basic 的安装程序时,系统将会在指定的硬盘上为 Visual Basic 创建安装目录。下面以 Visual Basic 6.0 企业版的安装为例,介绍如何安装 Visual Basic。

① 将 VB 6.0 的安装光盘插入 CD-ROM 驱动器,并运行安装程序 Setup.exe。

② 在进入安装程序后,用户要阅读一份“最终用户许可协议”,此时要单击“同意”按钮,才能进行下一步的安装。

③ 此时,系统会要求用户输入姓名、公司名称和 CD-KEY,输入完毕后,系统会要求选择安装 VB 6.0 的驱动器和文件夹,用户也可直接单击“确定”按钮,将系统安装至默认文件夹。

④ 在进行这些必要的步骤之后,将显示一个安装类型选择窗体,在该窗体中有两种安装方式供用户选择,分别是典型安装和自定义安装。

自定义安装是一种较好的安装方式,用户可根据需要选择要安装的组件;典型安装包含了 VB 的一些常用组件。一般情况下,可选择典型安装,单击“典型安装”按钮,即可开始应用程序的安装,安装完毕后会 Windows 的开始菜单中添加 Microsoft Visual Studio 6.0 程序组。

⑤ Visual Basic 6.0 具有自动卸载的功能,当不需要该软件时,可以很方便地将其从硬盘上删除。操作方法为:进入“控制面板”,双击窗口中的“添加/删除程序”图标,在打开的对话框中选择“Microsoft Visual Basic 6.0 中文企业版”选项,单击“添加/删除”按钮,按系统的提示进行操作即可卸载。所有组件被删除后,需要重新启动 Windows 系统以便更新系统。

1.3.3 启动与退出 Visual Basic

1. 启动 Visual Basic

Visual Basic 安装成功后,其启动步骤如下:

① 单击 Windows 的“开始”按钮。

② 在弹出的菜单中选择“程序”。

③ 用鼠标指针移向“Microsoft Visual Basic 6.0 中文版”选项。

④ 在弹出的菜单中选择“Microsoft Visual Basic 6.0 中文版”选项,即可启动 Visual Basic 6.0。

2. 退出 Visual Basic

打开 Visual Basic 的“文件”菜单,选择其中的“退出”命令,即可退出 Visual Basic。



1.4 Visual Basic 的集成开发环境

Visual Basic 拥有一个集成式的开发环境,所有的图形界面设计和代码的编写、调试、运行、编译均在该集成环境中完成。为使读者能尽快熟悉和掌握 Visual Basic 的集成开发环境,本节将对 Visual Basic 开发环境中的主窗口、窗体窗口、工具箱窗口、对象浏览器窗口、工程资源管理器窗口、属性窗口、窗体布局窗口以及代码编辑器窗口进行详细介绍。

启动 Visual Basic 后,系统首先弹出“新建工程”对话框,其中列出了 VB 6.0 能够创建的各种应用程序类型,如图 1-1 所示。系统默认创建工程类型为标准 EXE 文件,初学者可以直接单击对话框的“确定”按钮选择默认的程序,进入 Visual Basic 集成开发环境,如图 1-2 所示。

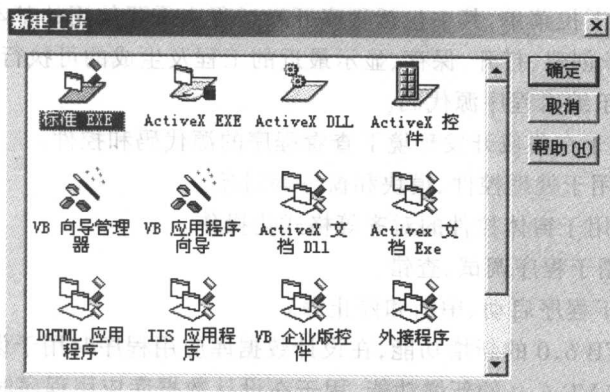


图 1-1 “新建工程”窗口

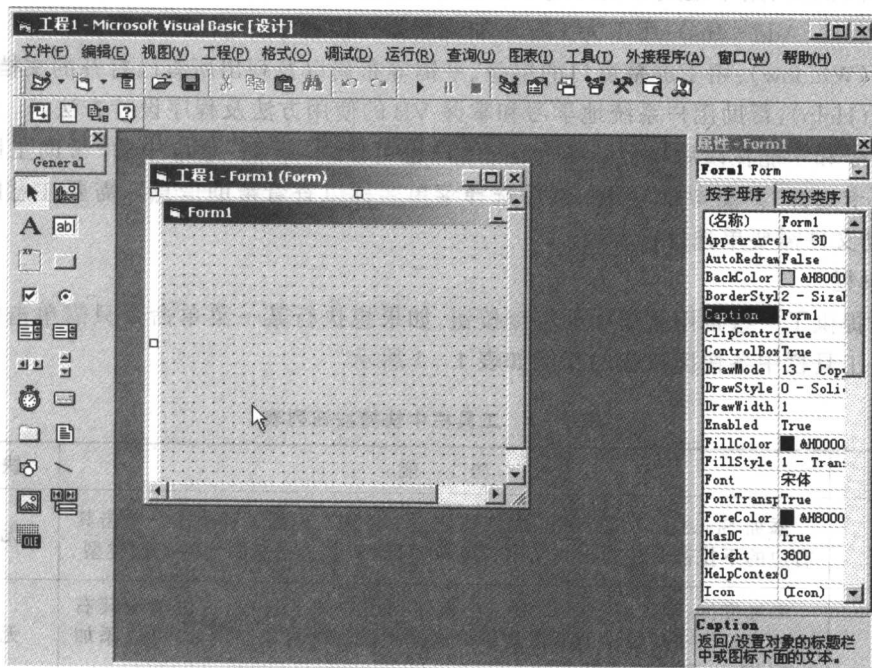


图 1-2 Visual Basic 的集成开发环境



1.4.1 主窗口

主窗口由标题栏、菜单栏和工具栏等组成。

1. 标题栏

标题栏中的标题为“工程 1—Microsoft Visual Basic [设计]”,说明此时集成开发环境处于设计模式,在进入其他状态时,方括号中的文字将做相应的变化。

VB 有 3 种工作模式:设计(Design)模式、运行(Run)模式和中断(Break)模式。

同 Windows 系统的用户界面一样,标题栏的最左端是窗口控制菜单;标题栏的右端是最大化、最小化和关闭按钮。

2. 菜单栏

菜单栏包括 13 个下拉菜单,其中包括程序开发过程中需要使用的基本命令。

- 文件(File):用于创建、打开、保存、显示最近的工程及生成的可执行文件。
- 编辑(Edit):用于编辑程序源代码。
- 视图(View):用于在集成开发环境下查看程序的源代码和控件。
- 工程(Project):用于处理控件、模块和窗体等对象。
- 格式(Format):用于窗体控件的对齐等格式化操作。
- 调试(Debug):用于程序调试、查错。
- 运行(Run):用于程序启动、中断和停止等。
- 查询(Query):VB 6.0 的新增功能,在设计数据库应用程序时用于设计 SQL 属性。
- 图表(Diagram):VB 6.0 的新增功能,用于在设计数据库应用程序时编辑数据库。
- 工具(Tools):用于集成开发环境下工具的扩展。
- 外接程序(Add - Ins):用于为工程添加或删除外接程序。
- 窗口(Window):用于屏幕窗口的层叠、平铺等布局以及列出所有已打开的文档窗口。
- 帮助(Help):帮助用户系统地学习和掌握 VB 的使用方法及程序设计方法。

各种命令和功能在以后将会陆续介绍,用户也可使用“帮助”菜单中的“帮助主题”命令,在“目录”选项卡选择“界面参数”主题,然后选择菜单名称以查看帮助信息。需要注意的是,Visual Basic 的帮助文件需要单独进行安装。

3. 工具栏

工具栏提供了一些常用菜单项的快捷按钮,如果想执行某一菜单命令只需单击相应的快捷按钮即可。工具栏中各快捷按钮的功能如表 1-1 所示。

表 1-1 工具栏中快捷按钮列表

图 标	功 能	快 捷 键
	添加 Standard EXE 工程:用来添加一个新的工程到工程组中。单击其右边的下拉箭头将弹出一个下拉菜单,可以从中选择想添加的工程类型	无
	添加窗体:默认情况下添加一个窗体到用户的工程中,也可单击其右边的下拉箭头从弹出的下拉菜单中选择想添加的对象。例如,可以添加 MDI 窗体、用户控件等	无



续表

图 标	功 能	快 捷 键
	菜单编辑器:用来显示菜单编辑器对话框	Ctrl + E
	打开工程:用于打开存在的工程文件	Ctrl + O
	保存工程:用于保存当前工程	无
	启动:开始运行当前工程	F5
	中断:中断当前运行的工程	Ctrl + Break
	结束:结束运行当前的工程	无
	工程资源管理器:打开“工程资源管理器”窗口	Ctrl + R
	属性:打开“属性”窗口	F4
	窗体布局窗口:打开“窗体布局”窗口	无
	对象浏览器:打开“对象浏览器”窗口	F2
	工具箱:打开“工具箱”窗口	无
	数据视图:打开数据视图窗口	无
	可视化部件管理器:打开可视化部件管理器	无

Visual Basic 6.0 采用了新型的平面式工具栏,当鼠标移动到某个按钮上时,系统会自动弹出相应的功能提示。在工具栏的末端,显示的是窗体的左上角的坐标位置和窗体目前的宽度和高度。在 Visual Basic 中,默认的坐标度量单位采用的是一种名为 Twip(缇)的新型单位,该单位与屏幕分辨率无关。

$$1\text{Twip} = 1/567\text{ cm} = 1/20\text{ point(点)}$$

1.4.2 窗体窗口

窗体窗口如图 1-2 中间部分所示。窗体窗口具有标准窗口的一切功能,可移动、改变大小及缩成图标。窗体是 VB 应用程序的主要部分,用户通过与窗体上的控制部件交互得到相应结果。一个程序可以拥有许多窗体窗口,每个窗体窗口必须有一个唯一的窗体名称,建立窗体时默认名为 Form1、Form2……。

在设计状态下窗体是可见的,窗体的网格点间距可以通过“工具”菜单中的“选项”命令,在“通用”选项卡的“窗体设置网格”中输入“宽度”和“高度”来改变。运行时可通过属性控制窗体的可见性(窗体的网格始终不显示)。

1.4.3 工程资源管理器窗口

在 Visual Basic 中,把开发一个应用程序视为一项工程,用创建工程的方法来创建一个应用



程序,利用工程资源管理器窗口来管理一项工程。因此,工程资源管理器窗口中包含了创建一个应用程序的所有属性以及所需的所有文件的列表,如图 1-3 所示。VB 6.0 采用层次化管理方式显示各类文件,而且也允许同时打开多个工程(这时以工程组的形式显示)。

在工程资源管理窗口中,工程的所有文件分类按层次结构图的方式显示,通过单击含“+”的节点,可展开一层,单击含“-”的节点,可折叠分支。若要打开某窗体,只需用鼠标双击该窗体文件即可。

创建好工程以后,可通过 VB“文件”菜单下的“保存工程”命令进行存盘,将其保存到一个工程文件中。在 VB 中,工程文件的扩展名为 .vbp,以后若要打开该工程,也是通过打开该工程文件来实现的。完成工程的全部文件之后,就可以通过“文件”菜单下的“生成工程”命令,将工程编译生成可执行的 EXE 文件。

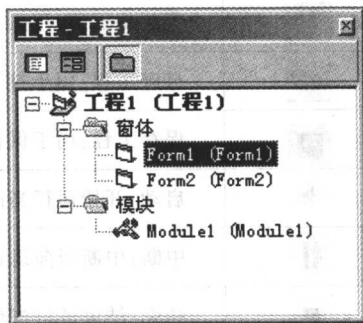


图 1-3 工程资源管理器窗口

注意:工程文件保存的仅是该工程所需的所有文件的一个列表,并不保存用户图形界面和程序代码。用户图形界面、各控件的属性设置值以及程序代码等,均保存在各窗体对应的窗体文件中(窗体文件的扩展名为 .frm),因此保存工程时同时也要保存窗体。

1.4.4 属性窗口

在 Visual Basic 中,属性窗口显示了一个对象在设计阶段的所有有效属性,通过属性窗口,可以设置或修改对象的属性值,如图 1-4 所示。在属性窗口中,属性可按字母顺序显示,也可按分类顺序显示,默认为按字母顺序显示,可通过单击排列方式选项卡进行切换。

选中一个对象后,按 F4 快捷键或单击工具栏上的属性按钮,即可弹出该对象的属性窗口。在属性列表的第 1 列,显示的是属性名称,第 2 列显示的是对应属性的当前取值,单击选中要修改的属性后,即可在第 2 列的对应位置输入或选择属性的具体取值。选中某项属性后,在属性窗口的底部会有对该属性功能的简单说明。



图 1-4 “属性”窗口

1.4.5 代码编辑器窗口

用户图形界面设计完毕后,第 2 阶段的工作是针对要响应用户操作的对象编写程序代码。在 Visual Basic 中,专门为程序代码的书写提供了一个代码编辑窗口。选中要编程的对象,按 F7 键,即可弹出该对象的代码编辑窗口,然后即可在该窗口的事件过程中书写程序代码。此外,也可以通过直接双击要编程的对象弹出代码编辑窗口,如图 1-5 所示。

在编辑窗口中,通常会显示该对象的一个默认事件过程框架,在如图 1-5 所示的编辑窗口中,就显示了窗体对象的 Load 事件过程。若要更改编程的对象,或者更改对象所要响应的



事件,可通过代码编辑窗口顶部的两个下拉列表框来实现,左边的列表框用于选择要编程的对象,右边的列表框用于选择该对象所要响应的事件。单击列表框右边的下拉按钮,即可弹出相应的列表选项,如图 1-6 所示。对象和对象要响应的事件确定后,代码编辑区中的事件过程框架就会自动产生,然后即可在事件过程框架中编写实现具体功能的程序代码。编写完毕后,单击编辑窗口的“关闭”按钮,将其关闭即可。若要观察程序的运行效果,可按 F5 键或单击工具栏上的“启动”按钮,运行该程序。

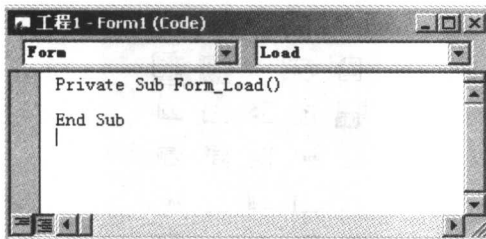


图 1-5 Visual Basic 代码编辑窗口

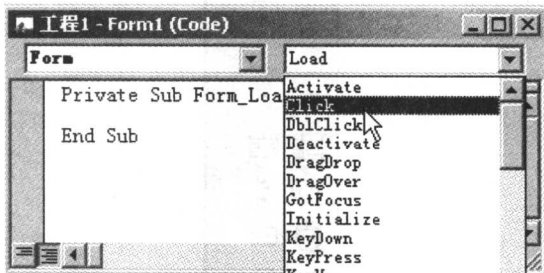


图 1-6 事件的选择方法

【例 1-1】 试编程实现:当用户单击窗体 Form1 时,在窗体上显示“You Click Me!”的提示信息。

分析:用户单击的对象是窗体 Form1,单击的结果是在窗体上显示一个字符串信息,因此需要针对窗体对象编程;而信息是在单击事件发生后显示的,因此也应对窗体对象的单击事件编程。

双击窗体打开代码编辑窗口,在窗口顶部右边的组合框中选择单击事件 Click,然后在 Click 事件过程中编写实现信息显示的程序代码,如图 1-7 所示。

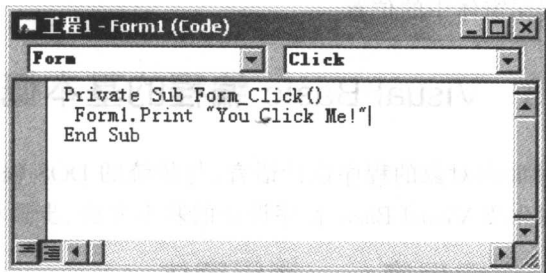


图 1-7 事件过程代码的编写

代码编写完毕后,按 F5 键运行应用程序。若要结束程序的运行,可单击工具栏上的“结束”按钮.

1.4.6 工具箱

开发环境左边是 Visual Basic 的工具箱,其中含有许多可视化的控制对象(控件),如图 1-8 所示。用户可以从工具箱中选取所需的控件,并将其添加到窗体中,以绘制所需的图形用户界面。工具箱中的控件越多,VB 的开发能力也就越强。

Visual Basic 启动时,一般仅在工具箱中会装载一些基本的控件,如图 1-8 所示。若要新增



控件,可通过 VB“工程”菜单下的“部件”命令来实现。单击“部件”命令,打开“部件”对话框,在“控件”列表框中,选中要添加的控件,然后单击“确定”按钮,被选中的控件就会添加到工具箱中。添加了其他控件的工具箱如图 1-9 所示。

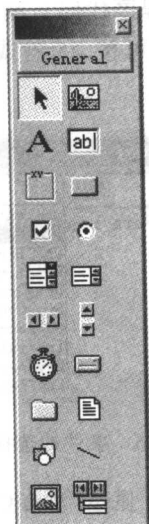


图 1-8 标准工具箱

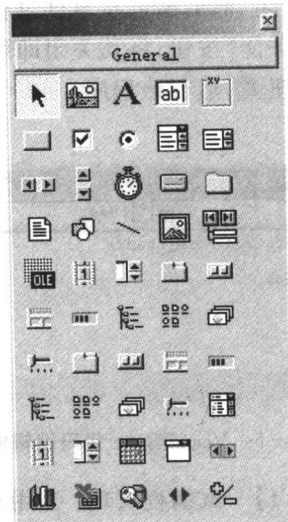


图 1-9 添加控件后的工具箱

有关常用控件将在第 3 章详细介绍,其他类型的高级控件和 ActiveX 控件将在第 6、7、8 章中进行介绍。若要在窗体中加入某控件,只要从工具箱中选中该控件然后在 Form 窗体中按下鼠标左键并拖动,即可将控件添加到窗体中。可用鼠标左键按住控件四周的边框调整其大小,也可用鼠标拖动控件改变其在窗体中的位置。

1.5 Visual Basic 编程的基本概念

Visual Basic 作为一种面向对象的程序设计语言,与传统的 DOS 程序设计有较大的区别,本节将通过具体的编程实例介绍 Visual Basic 程序设计的基本方法、步骤和相关的概念。

1.5.1 编写第一个应用程序——使用窗体

【例 1-2】编写一个 VB 程序,在运行时若用鼠标单击窗体,在窗体上会显示出“我的第一个 VB 程序!”文字,如图 1-10 所示。

要用 VB 实现一个任务,必须解决两类问题:

① 设计一个用户操作界面,输入或输出信息都在这个界面中进行。

② 设计程序代码。使程序运行后能按规定的目标和步骤进行操作,以达到题目的要求。

当用户创建一个新的工程文件时,VB 会自动给

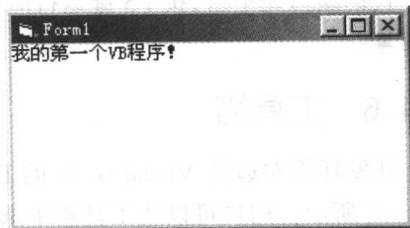


图 1-10 程序运行界面



出默认名为 Form1 的窗体。窗体是用来设计应用程序用户界面的,在其上面可以添加各种控件,最终形成美观实用的用户界面。对本例来说,用户界面无特殊要求,只要求在窗口中输出一行文字,因此不必专门设计用户界面,也不必使用工具箱中的工具,只需编写程序代码,使其输出所要求的文字即可。

通常,需要在程序代码窗口中编写程序。下面介绍一下进入代码窗口的两种方法:

① 双击当前窗体(双击一个控件也可以进入该控件所对应的代码窗口)。

② 选中当前窗体后,按 F7 键。

双击窗体,进入与该窗体相对应的代码窗口(见图 1-5)。

此代码窗口有一个标题栏,显示窗体的名字(图 1-5 中“工程 1”之后的“Form1”表示窗体的名字,Code 表明是代码窗口)。标题栏下面有两部分,左边是用于显示对象的下拉列表框(称为对象框),包含所有与当前窗体相联系的对象。由于是双击窗体进入代码窗口的,所以对象框中显示的是 Form。如果现在要对其他对象进行编码,可单击该对象框右侧的下拉按钮打开显示对象的下拉列表框,从中选择所需要的对象。

右边显示了与当前选中的对象相关的所有事件(称为过程框)。单击右侧的下拉按钮,打开下拉列表框,用鼠标单击所需的事件名,就可以对所选择的对象和事件进行编码(见图 1-6)。

本例题要求在运行时若单击窗体就应显示出一行字符,所以可按以下方法进行操作:

① 在对象框中选择 Form(窗体)。

② 在过程框中选择 Click(单击)事件(Click)。当选择了对象和事件后,在代码窗口会立即出现相应的 Form_Click()过程的框架:

```
Private Sub Form_Click()
```

```
End Sub
```

此时,就可以在 Private Sub Form_Click()与 End Sub 之间输入程序语句。在此,输入以下语句:Print“我的第一个 VB 程序!”,如图 1-11 所示。

以上 Print 语句的作用是将双引号中的内容原样输出到窗体上。在 VB 中,Print 被称为一种“方法”。关于“方法”的概念将在 1.5.2 节进行介绍。

至此,已经编写了一个对窗体单击事件的响应过程,即在运行程序时,若用户单击窗体,系统就会执行以下过程:

```
Private Sub Form_Click()  
Print "我的第一个 VB 程序!"  
End Sub
```

其中,关键字 Private(私有)表示该过程只能在本窗体文件中被调用,不能被应用程序中的其他窗体或模块所调用。关键字 Sub 是过程的标志;Form_Click()是过程名,由两部分组成:对象和事件名,两者之间用下划线连接;End Sub 表示过程结束。

在编写好程序后可通过 VB 中提供的几种运行程序的方法(参见 1.5.4 节)运行,其中之一是从菜单栏的“运行”菜单中选择“启动”命令。

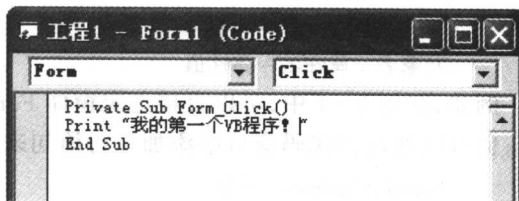


图 1-11 编写程序代码



本程序进入运行状态后,用鼠标单击窗体,窗体上就出现一行“我的第一个 VB 程序!”,再单击一次就会再显示一行。当用户从“运行”菜单中选择“结束”命令后,程序结束运行。

1.5.2 Visual Basic 对象的属性、事件和方法

1. 对象的概念

在现实生活中,任何一个实体都可以视为一个对象(Object)。例如,一只气球是一个对象,一台计算机也是一个对象。在 VB 中,工具箱中包含了许多工具,例如文本框、标签框、命令按钮等,它们以不同的图标形式排列在工具箱中,用这些工具可在窗体上画出各种各样的图形。工具箱中的每一个图标代表一个控件(Control),这些控件就是一种对象。

除了控件对象外,VB 还提供了系统对象,例如打印机、屏幕、应用程序以及数据库等。此外,窗体也是对象。

2. 对象的属性

每个对象都有一组特征,称为属性。不同的对象有不同的属性,例如气球所具有的属性包括它的直径、颜色及状态(充气的或未充气的)等。此外,还有一些不可见的性质,如它的寿命等。通过定义,所有气球都具有这些属性,但这些属性也会因气球的不同而不同。

在可视化编程中,每一种对象都有一组特定的属性。有许多属性可能为大多数对象所共有,如 BackColor 属性定义对象的背景色。也有一些属性只局限于个别对象,如只有命令按钮才有 Cancel 属性,该属性用来确定命令按钮是否为窗体默认的“取消”按钮。

对象的每个属性都有一个默认值,可根据具体的要求修改此值。通过修改对象的属性能够控制对象的外观和行为。设置对象的属性一般有两种方法:

① 选定对象,然后在属性窗口中找到相应的属性直接进行设置。这种方法的特点是简单明了,每当选择一个属性时,在属性窗口的下部就显示该属性的一个简短提示;缺点是不能设置所有需要的属性。

② 在代码中通过编程设置,格式如下:

对象名.属性名=属性值

例如,在例 1-1 中,可以直接选定窗体 Form1,然后在属性窗口中将其 Caption 属性改为“例题”;也可以通过在代码窗口中添加一行语句来实现:

Form1.Caption = "例题"

3. 对象的事件

事件(Event)就是对象上所发生的事情。在 VB 中,事件是预先定义好的、能够被对象识别的动作,如单击(Click)事件、双击(DblClick)事件、装载(Load)事件、鼠标移动(MouseMove)事件等,不同的对象能够识别不同的事件。当事件发生时,VB 将检测两条信息,即发生的是哪种事件及哪个对象接收了事件。

每种对象都能识别一组预先定义好的事件,但并非每一种事件都会产生结果。为了使对象能够对某一事件做出响应,就必须编写事件过程。

事件过程是一段独立的程序代码,在对象检测到某个特定事件时执行这些代码(响应该事件)。一个对象可以响应一个或多个事件,因此可以使用一个或多个事件过程对用户或系统的事件做出响应。例如,在例 1-2 中,窗体 Form1 可以识别许多事件,但此时只编写了单击(Click)



事件的事件过程,因此程序运行时,只对单击(Click)事件做出响应。

事件过程的名字由对象名和事件名两部分组成,两者之间用一条下画线连接,其一般格式为:

```
Private Sub 对象名_事件名()  
...  
End Sub
```

其中,Sub 是定义过程开始的语句,End Sub 是定义过程结束的语句,关键字 Private 表示该过程是私有过程。具体编程时,用户可以在过程开始语句和结束语句之间添加实现具体功能的程序代码。在例 1-2 中,在单击事件过程中添加了在窗体上输出字符串的功能语句:Print "我的第一个 VB 程序!"。

4. 事件的方法

VB 中的对象除拥有自己的属性和事件外,还拥有属于自己的方法(Method)。“方法”是 VB 中的一个术语,所谓“方法”实际上是 VB 为程序设计人员提供了一种特殊的过程或函数,用来完成一定的操作或实现一定的功能。在 VB 中已将一些通用的过程和函数编写好并封装起来,作为方法供用户直接调用,这给用户带来了很大的方便。例如,Print 就是一种方法,它用来输出信息的专用过程。

方法是面向对象的,所以调用方法的形式与调用一般的过程或函数不同,应该指明是哪个对象调用的。其调用格式如下:

对象名.方法名[可选参数项]

例如,Form1.Hide 表示由 Form1 对象调用 Hide 方法,其执行结果是将窗体 Form1 隐藏起来。同样,Form1.Print "我的第一个 VB 程序!"的作用是将字符串“我的第一个 VB 程序!”显示在窗体 Form1 内,其中“我的第一个 VB 程序!”是可选参数。如果写成 Printer1.Print "我的第一个 VB 程序!"则是会在打印机上打印出该字符串。

在上面的调用格式中,如果省略了对象名,则隐含指当前对象。例如,在例 1-2 中的 Form1_Click()事件过程中直接使用:

```
Print "我的第一个 VB 程序!"
```

调用 Print 方法的,在此就隐含表示要在窗体 Form1 上输出该字符串。一般来说,为了提高程序的可读性,最好在方法名之前加上对象名。此外,还应注意每一种对象所能调用的方法是不完全相同的。

1.5.3 编写第二个应用程序——使用控件

【例 1-3】 设计一个程序,用户界面由 3 个命令按钮和一个文本框组成。当用户单击其中一个命令按钮时,在文本框中显示文本内容;单击另一个命令按钮时,清除文本框中的内容;单击第三个命令按钮时,结束程序的运行。

根据题目要求,设计用户界面,如图 1-12 所示。

首先,进入 VB 编程环境,建立一个新工程,再将一个文本框和 3 个命令按钮添加到窗体 Form1 上。在窗体上添加控件的方法是:用鼠标单击工具箱中的控件图标,然后松开鼠标左键将光标移到窗体上,这时光标由箭头变成十字形。将十字形光标移到窗体上,按住鼠标左键斜向拖动至所需要的尺寸,然后释放左键,即可将一个控件添加到窗体上。



图 1-12 设计程序界面

在窗体上添加了上述几个控件后,即可开始设置窗体和各控件的属性。本例题中需要设置的属性如表 1-2 所示。

表 1-2 例 1-3 的对象属性设置

对 象	属 性	设 置
窗体	Caption (名称)	Form1 Form1
文本框	(名称) Text FontName FontSize	Text1 置空 隶书 三号
命令按钮 1	Caption (名称) FontSize	显示 cmdShow 五号
命令按钮 2	Caption (名称) FontSize	清除 cmdClear 五号
命令按钮 3	Caption (名称) FontSize	退出 cmdEnd 五号

在本例中,窗体的 Caption(标题)和名称属性均使用了系统给定的默认值,故不需要重新设置。

设置控件的属性时,首先单击窗体上某一控件,使其成为当前控件(注意:当前控件的四周有 8 个蓝色小方块),然后在属性窗口中找到需要设置的属性,再指定属性值。例如,若把命令按钮 Command1 的 Caption(标题)属性设为“显示”,就应该先单击命令按钮 Command1,使其成为当前控件,再到属性窗口中找到 Caption 属性。可以看到,系统事先为命令按钮 Command1 设置的属性值(称为默认值)是 Command1。为了改变系统给定的标题,在此可以删除字符 Command1 并用汉字重新输入“显示”,这时可以看到命令按钮 1 上的文字已由 Command1 改变为“显示”,从而完成了该命令按钮的 Caption 属性设置。

从属性表中找到“名称”属性,其属性值为 Command1(系统默认值),将其删除,修改为 cmdShow。