

21世纪高等学校公共课计算机规划教材

VB语言程序设计 (第2版)

林卓然 编著

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



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

21 世纪高等学校公共课计算机规划教材

VB 语言程序设计

(第 2 版)

林卓然 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书以 Visual Basic 6.0 为语言背景,以程序结构为主线,介绍可视化程序设计的基本知识和编程方法。内容包括:认识 Visual Basic,程序设计基础,顺序结构程序设计,选择结构程序设计,循环结构程序设计,数组,过程,程序调试与错误处理,数据文件与文件管理,菜单及对话框,绘图及其他常用控件等。

本书概念清晰,层次分明,叙述简明易懂,注重实用性和可操作性。各章配有精心设计的习题和编程及上机调试题。本书为任课教师免费提供电子教案及习题参考答案。

本书适合作为高等学校计算机程序设计教材,也可作为各类 VB 培训班及全国计算机等级考试读者的学习参考书。

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,侵权必究。

图书在版编目(CIP)数据

VB 语言程序设计/林卓然编著. —2 版. —北京:电子工业出版社,2009.1

21 世纪高等学校公共课计算机规划教材

ISBN 978-7-121-07866-8

I. V… II. 林… III. BASIC 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 183068 号

策划编辑:童占梅

责任编辑:童占梅

印 刷: 北京市李史山胶印厂
装 订:

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:16 字数:403 千字

印 次:2009 年 1 月第 1 次印刷

印 数:4 000 册 定价:25.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

再版前言

本教材自 2003 年 2 月出版以来,得到了各高校教师及广大学生的好评和支持。在此对几年来关心、支持并对本教材提出意见和建议的教师及广大学生表示衷心感谢。

此次修订,继续保持上一版本的层次清楚、通俗易懂、便于教与学等特点,进一步优化了书中内容编排、叙述表达、习题选择等。对各章编程题也进行了大的改动,一是使每个实验要求更加明确,更具可操作性;二是增加修改程序的练习,培养学生分析程序和调试程序的能力,并为编写一个完整程序打下良好基础。修订教材还增加了常用算法,以提高学生解决实际问题的能力。

本书适合作为第一门程序设计语言学习的教材,它从最基本的计算机程序设计基础知识讲起,讲授 Visual Basic 程序设计的基本概念和基本方法。书中提供的大量例题都是经过上机验证过的,读者可以边看书,边在计算机上操作。

本书紧扣全国计算机等级考试《Visual Basic 程序设计》考试大纲及近年来命题重点,有利于提高学生应试和获证能力。

为帮助教师使用本教材,编者准备了这本书的教学辅助材料,包括各章节的电子教案、例题源程序文件、习题参考答案等,并发布在电子工业出版社华信教育资源网站上,其网址为 <http://www.huaxin.edu.cn>。

在本书的编写过程中,何丁海、彭金泉、阮文江、梁广德、李伟林等老师提出了宝贵意见,在此表示衷心感谢。

由于本人水平所限,加之计算机技术发展日新月异,书中错误在所难免,失误之处,敬请读者指正。作者电子邮件地址: puslzs@mail.sysu.edu.cn。

编著者
于中山大学

本书提供电子教案、例题源程序文件和习题参考答案,请从华信教育资源网 <http://www.huaxin.edu.cn> 注册下载。

目 录

第 1 章 认识 Visual Basic	(1)
1.1 VB 概述	(2)
1.1.1 VB 的发展过程	(2)
1.1.2 VB 的特点	(2)
1.1.3 VB 的启动和退出	(3)
1.2 对 VB 应用程序设计的初步认识	(4)
1.2.1 设计 VB 应用程序的步骤	(4)
1.2.2 第一个简单程序	(5)
1.2.3 第二个简单程序	(7)
1.3 对象和事件的基本概念	(10)
1.3.1 对象及其属性、方法和类	(10)
1.3.2 事件、事件过程及事件驱动	(11)
1.4 VB 的集成开发环境	(13)
1.4.1 主窗口	(13)
1.4.2 工具箱和控件	(14)
1.4.3 其他窗口	(16)
1.4.4 使用帮助系统	(17)
1.5 工程管理	(18)
1.5.1 工程中的文件	(18)
1.5.2 创建、打开和保存工程	(19)
1.5.3 添加、删除和保存文件	(19)
1.5.4 程序的运行	(20)
习题 1	(20)
编程及上机调试	(23)
第 2 章 程序设计基础	(25)
2.1 数据类型	(26)
2.2 常量与变量	(28)
2.2.1 常量	(28)
2.2.2 变量	(30)
2.3 表达式	(31)
2.3.1 算术表达式	(31)
2.3.2 字符串表达式	(32)
2.3.3 日期表达式	(33)
2.4 常用内部函数	(33)
2.4.1 数学函数	(33)
2.4.2 字符串函数	(36)

2.4.3 日期/时间函数	(37)
2.4.4 类型转换函数	(37)
2.5 程序代码编写规则	(38)
2.5.1 语句及语法	(38)
2.5.2 代码书写规则	(39)
2.5.3 结构化程序的基本结构	(40)
习题 2	(41)
编程及上机调试	(43)
第 3 章 顺序结构程序设计	(45)
3.1 赋值语句	(46)
3.2 注释、结束与暂停语句	(48)
3.3 使用 Print 方法输出数据	(49)
3.3.1 Print 方法	(49)
3.3.2 特殊打印格式	(50)
3.4 窗体	(51)
3.4.1 窗体的基本属性	(52)
3.4.2 窗体的事件	(53)
3.4.3 窗体的方法	(55)
3.4.4 焦点与 Tab 键序	(55)
3.5 基本控件	(57)
3.5.1 控件的公共属性	(57)
3.5.2 命令按钮	(58)
3.5.3 标签	(59)
3.5.4 文本框	(59)
3.6 使用对话框	(60)
3.6.1 输入对话框	(61)
3.6.2 消息对话框	(62)
3.7 程序举例	(65)
习题 3	(68)
编程及上机调试	(71)
第 4 章 选择结构程序设计	(73)
4.1 条件表达式	(74)
4.1.1 关系表达式	(74)
4.1.2 逻辑表达式	(75)
4.2 条件语句	(75)
4.2.1 If ... Then 语句	(76)
4.2.2 If ... Then ... Else 语句	(76)
4.2.3 IIf 函数	(78)
4.2.4 条件语句的嵌套	(78)
4.3 多分支语句	(80)
4.4 选择性控件	(82)

4.4.1 单选按钮	(82)
4.4.2 复选框	(83)
4.5 计时器控件	(85)
4.6 程序举例	(87)
习题 4	(91)
编程及上机调试	(94)
第 5 章 循环结构程序设计	(97)
5.1 循环语句	(98)
5.1.1 For ... Next 循环语句	(98)
5.1.2 Do ... Loop 循环语句	(100)
5.1.3 While ... Wend 循环语句	(103)
5.1.4 循环出口语句	(103)
5.2 多重循环	(104)
5.3 列表框与组合框	(107)
5.3.1 列表框	(107)
5.3.2 组合框	(111)
5.4 常用算法	(112)
5.5 程序举例	(115)
习题 5	(118)
编程及上机调试	(121)
第 6 章 数组	(123)
6.1 数组的概念	(124)
6.1.1 数组与数组元素	(124)
6.1.2 下标和数组的维数	(125)
6.2 数组的声明和应用	(125)
6.2.1 数组声明语句	(125)
6.2.2 Array 函数	(126)
6.2.3 数组的应用	(126)
6.3 动态数组	(129)
6.3.1 建立动态数组	(129)
6.3.2 数组刷新语句	(130)
6.4 For Each ... Next 循环语句	(131)
6.5 控件数组	(132)
6.5.1 控件数组的概念	(132)
6.5.2 控件数组的建立	(132)
6.5.3 控件数组的使用	(132)
6.6 程序举例	(134)
习题 6	(140)
编程及上机调试	(143)
第 7 章 过程	(146)
7.1 通用过程	(147)

7.1.1 Sub 过程	(148)
7.1.2 Function 过程	(151)
7.2 参数传递	(153)
7.2.1 形参与实参	(153)
7.2.2 按地址传递和按值传递	(153)
7.3 过程的嵌套调用	(154)
7.4 变量的作用范围	(155)
7.4.1 代码模块的概念	(155)
7.4.2 变量的作用域	(156)
7.4.3 变量的生存期	(157)
7.5 多窗体与 Sub Main 过程	(158)
7.5.1 多窗体处理	(158)
7.5.2 Sub Main 过程	(162)
7.6 程序举例	(163)
习题 7	(167)
编程及上机调试	(170)
第 8 章 程序调试与错误处理	(173)
8.1 VB 程序中的错误类型	(174)
8.2 程序工作模式	(175)
8.3 程序调试	(176)
8.3.1 静态检查与动态检查	(176)
8.3.2 使用 VB 调试工具	(177)
8.4 错误处理	(181)
8.4.1 错误处理的步骤	(181)
8.4.2 Err 对象	(181)
8.4.3 捕获错误语句	(181)
8.4.4 退出错误处理语句	(182)
习题 8	(183)
编程及上机调试	(183)
第 9 章 数据文件与文件管理	(187)
9.1 数据文件	(188)
9.1.1 数据文件的基本概念	(188)
9.1.2 顺序文件	(190)
9.1.3 随机文件	(192)
9.1.4 二进制文件	(194)
9.2 文件基本操作	(194)
9.3 文件系统控件	(196)
习题 9	(198)
编程及上机调试	(200)
第 10 章 菜单及对话框	(202)
10.1 键盘与鼠标事件	(203)

10.1.1 键盘事件	(203)
10.1.2 鼠标事件	(203)
10.1.3 拖放操作	(205)
10.2 菜单设计	(207)
10.2.1 下拉式菜单	(207)
10.2.2 菜单编辑器	(207)
10.2.3 菜单的 Click 事件	(209)
10.2.4 运行时改变菜单属性	(212)
10.2.5 弹出式菜单	(212)
10.3 对话框	(213)
10.3.1 通用对话框	(214)
10.3.2 自定义对话框	(219)
习题 10	(219)
编程及上机调试	(222)
第 11 章 绘图及其他常用控件	(223)
11.1 框架	(224)
11.2 滚动条	(226)
11.3 图形方法和图形控件	(227)
11.3.1 坐标系	(228)
11.3.2 图形方法	(228)
11.3.3 图片框	(229)
11.3.4 图像框	(231)
11.3.5 Shape 形状控件	(231)
11.3.6 Line 直线控件	(232)
11.4 GoTo 语句和几个定义语句	(232)
习题 11	(234)
编程及上机调试	(236)
附录 A 字符 ASCII 码表	(238)
附录 B 颜色代码	(239)
附录 C 习题参考答案	(241)

第¹章

认识 Visual Basic

本章导读

- VB 概述
- 对 VB 应用程序设计的初步认识
- 对象和事件的基本概念
- VB 的集成开发环境
- 工程管理

Visual Basic(以下简称 VB)是一种可视化程序设计语言,是目前在 Windows 操作平台上广泛使用的 Windows 应用程序开发工具。在深入学习 VB 编程之前,本章先介绍 VB 的特点、集成开发环境及面向对象的基本概念。

1.1 VB 概述

1.1.1 VB 的发展过程

Basic 语言于 1964 年诞生,其含义为“初学者通用的符号指令代码”,由于它简单易学而一直被大多数初学者作为首选入门的程序设计语言。随着计算机技术的发展,各种 Basic 语言新版本应运而生。1976 年前后开发出 DOS 环境下的 GW-Basic,20 世纪 80 年代中期又出现了多种结构化 Basic 语言,如 True Basic,Quick Basic,Turbo Basic,QBasic 等。

1988 年,美国微软公司(Microsoft)推出的 Windows 操作系统,以其友好的图形用户界面(GUI)、简单易学的操作方式和卓越的性能,赢得了广大计算机用户的喜爱,因此开发在 Windows 环境下的应用程序成为 20 世纪 90 年代软件开发的主导潮流。起初人们在开发 Windows 应用程序时遇到了很大困难,因为要编写 Windows 环境下运行的程序,必须建立相应的窗口、菜单、对话框等各种“控件”,程序的编制变得越来越复杂。

1991 年,微软公司推出的 VB 1.0,使这种情况有了根本的改观。微软公司总裁比尔·盖茨说,VB 1.0 是“用 Basic 语言开发 Windows 应用程序最强有力的工具”、“令人震惊的新奇迹”。VB 中的“Visual”的含义是“可视化”,指的是一种开发图形用户界面的方法。VB 采用的“可视化编程”是“面向对象编程”技术的简化版,它引入了面向对象和事件驱动的程序设计新机制,把过程化和结构化编程结合在一起,其解决问题的方式更符合人们的思维习惯,为开发 Windows 应用程序提供了强有力的开发环境和工具。

随着 Windows 操作平台的不断成熟,VB 版本也不断升级。自 VB 1.0 之后,微软公司又相继推出 VB 2.0,VB 3.0,VB 4.0,这些版本主要应用于 Windows 3.X 环境中 16 位应用程序的开发。1997 年,微软公司发布了 VB 5.0,它是一个 32 位应用程序开发工具,可以运行在 Windows 9.X 或 Windows NT 环境中。1998 年,微软公司推出 VB 6.0,2002 年又开发出 VB.NET 7.0。

为满足不同层次的用户需要,VB 6.0 提供了学习版、专业版和企业版三个版本。这些版本是在相同的基础上建立起来的,因此大多数应用程序可在这三种版本中通用。本书主要介绍中文版 VB 6.0 的基本功能,对这三个版本都适用。

1.1.2 VB 的特点

VB 是在原有的 Basic 语言的基础上发展而来的。它具有 Basic 语言简单易用的优势,同时增加了面向对象和可视化程序设计语言的特点。

(1)可视化的设计平台。VB 提供可视化的设计平台,把 Windows 界面设计的复杂性“封装”起来,开发人员不必再为界面的设计而编写大量程序代码,只需用系统提供的工具在屏幕上“画出”(放置)窗口、命令按钮、文本框等各种对象。VB 系统将自动产生界面设计代码,编

程人员只需编写实现程序功能的那部分程序代码,从而大大提高了程序设计的效率。

(2)面向对象的设计方法。VB应用面向对象的程序设计方法(OOP),把程序和数据“封装”起来成为一个对象,每个对象都是可视的,并为每个对象赋予应有的属性。

(3)事件驱动的编程机制。VB通过事件来执行对象的操作,通常由用户操作引发某个事件来驱动完成某种功能。例如,命令按钮是一个对象,当用户单击该按钮时,将产生(或称“触发”)一个“单击”(Click)事件,而在发生该事件时,系统将自动执行一段相应的程序(称为“事件过程”),用以实现指定的操作和达到运算、处理的目的。

在VB中,编程人员只需针对这些事件编写相应的处理代码(即事件过程),这样的代码一般较短,所以程序既易于编写又易于维护。

(4)结构化的设计语言。VB是在结构化的Basic语言基础上发展起来的,加上面向对象的设计方法,因此是更出色的结构化程序设计语言。

(5)友好的VB集成开发环境。VB提供了易学易用的应用程序集成开发环境。在该集成开发环境中,编程人员可以设计用户界面、编写代码和调试程序,直至把应用程序编译成可执行文件,直接在Windows环境下运行。

(6)具有强大的功能。VB可以对多种数据库系统进行数据访问。VB支持对象的链接与嵌入(OLE)、动态数据交换(DDE)、动态链接库(DLL)及Active等技术,它能够充分利用Windows资源,开发出集文字、声音、图像、动画、Web等对象于一体的应用程序。

1.1.3 VB的启动和退出

1. 启动VB

VB是Windows下的一个应用程序,因此可按运行一般应用程序的方法来运行它。启动VB的常用方法是:单击“开始”按钮,从开始菜单中选择“程序”项,再选择“Microsoft Visual Basic 6.0 中文版”级联菜单中的“Microsoft Visual Basic 6.0 中文版”程序。

当然,也可将VB系统程序的快捷方式放在桌面上,直接在桌面上双击该快捷方式图标来启动它。

启动VB后,作为默认方式,系统会首先弹出“新建工程”对话框,如图1.1所示。在对话框中,有三个选项卡:

(1)新建:列出了可以创建的应用程序类型,其中“标准EXE”用来建立一个VB应用程序,最终可生成一个标准的可执行文件(.exe文件)。

(2)现存:供选择和打开的现有工程。

说明:VB应用程序是以工程的形式组织的。一般情况下,一个工程就是一个应用程序。

(3)最新:列出最近使用过的工程。

直接单击对话框右下方的“打开”按钮,则可创建一个默认的“标准EXE”类型的应用程序,进入VB集成开发环境,如图1.2所示。

2. 退出VB

如果要退出VB,可单击VB主窗口中的“关闭”按钮或选择“文件”菜单中的“退出”命令,

VB 会自动判断用户是否修改了工程的内容,询问用户是否保存文件或直接退出。

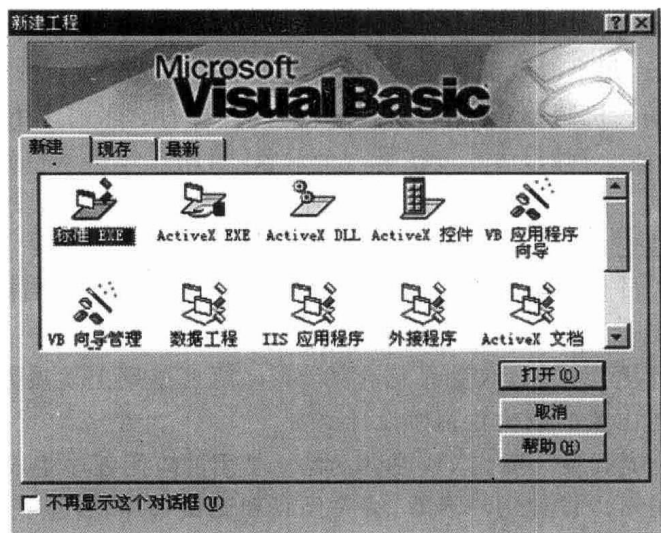


图 1.1 “新建工程”对话框

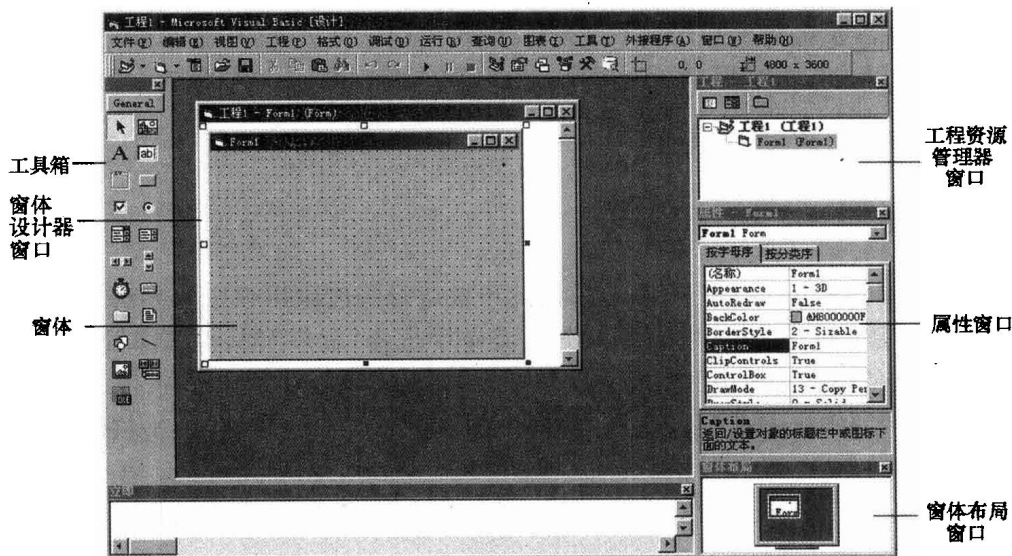


图 1.2 VB 集成开发环境

1.2 对 VB 应用程序设计的初步认识

1.2.1 设计 VB 应用程序的步骤

采用 VB 开发应用程序,一般可分为两大部分工作:设计用户界面和编写程序代码。所谓用户界面,是指人与计算机之间传递、交换信息的媒介,是用户使用计算机的综合操作环境。通过用户界面,用户向计算机系统提供命令、数据等输入信息,这些信息经过计算机处理后,又经过用户界面,把计算机产生的输出信息送回给用户。

VB 应用面向对象的程序设计方法,因此先要确定对象,然后才能针对这些对象进行代码编程。VB 编程中最常用的对象是窗体(即平时所说的窗口),各种控件对象必须建立在窗体上。用户界面设计又包括建立对象和对象属性设置两部分。

设计 VB 应用程序的大致步骤如下:

- (1) 建立用户界面的对象。
- (2) 设置对象的属性值。
- (3) 编写程序代码,建立事件过程。
- (4) 保存和运行应用程序。

为了使读者对 VB 程序设计有一个初步认识,以下举两个简单例子。

1.2.2 第一个简单程序

【例 1.1】 设计一个程序,在运行中当用鼠标单击窗体时,窗体上显示出“欢迎您来到 VB 世界!”字样。

该应用程序设计步骤如下。

1. 创建窗体

启动 VB 后,选择“标准 EXE”选项,进入 VB 集成开发环境。此时系统已经自动创建了一个窗体 Form1,如图 1.2 所示。

对于本例,用户界面无特殊要求,只要在系统默认提供的窗体上输出若干文字,因此不必专门设计用户界面。

2. 编写程序代码,建立事件过程

编写程序代码需要在“代码窗口”中进行。

在 VB 主窗口中选择“视图”菜单中的“代码窗口”命令,或双击 Form1 窗体,系统弹出与该窗体相对应的代码窗口,如图 1.3 所示。

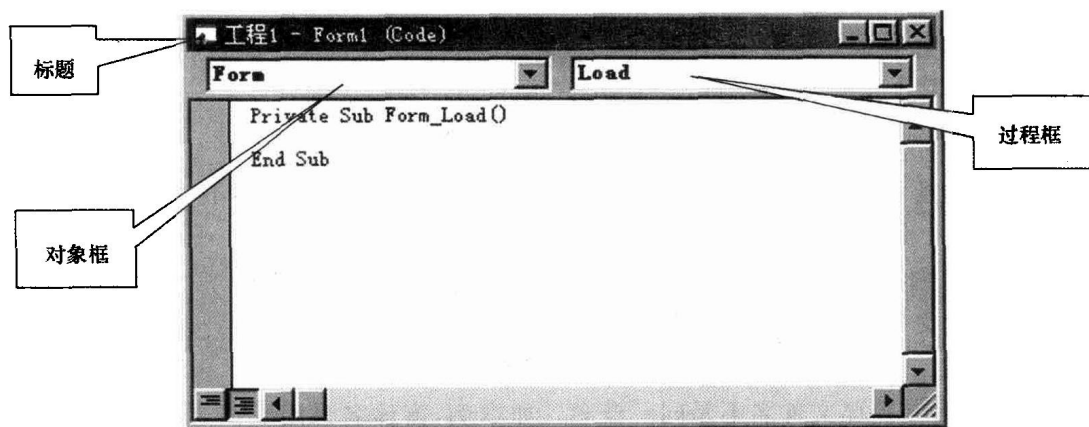


图 1.3 代码窗口

这个代码窗口有一个标题“工程 1—Form1(Code)”,表示当前工程名默认为“工程 1”,这与 Word 中默认第一个文档为“文档 1”一样。Form1 表示窗体名,圆括号内的 Code 表示代码窗口。

第二行左侧是一个对象框,其下拉列表框中列出了与当前窗体相联系的对象;第二行右侧是一个过程框,其下拉列表框中列出了与当前选中的对象相关的所有事件。

在对象框中选择对象 Form,在过程框中选择事件 Click(即单击)。当选择了对象和事件后,在代码窗口的编辑区中立即自动出现 Form_Click 事件过程的框架,如图 1.4 所示。

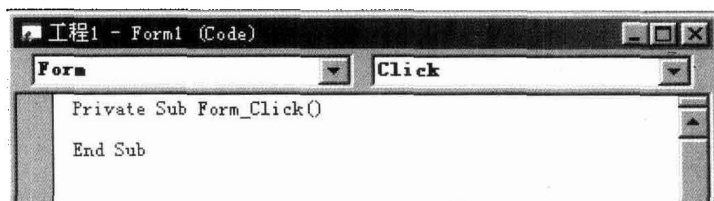


图 1.4 Form_Click 事件过程框架

说明: ① Private 意为“私有”,用来定义事件过程的类型,它表明该过程只能在本窗体中使用,应用程序中的其他窗体或模块不能调用它;② 关键字 Sub 和 End Sub 用于定义一个过程;③ Form_Click 表示事件过程名,它由两部分组成:对象名和事件名;④Form_Click 后面是一对圆括号,对于复杂的过程,圆括号内还可以有参数。

在已有两行代码之间插入一行代码,即

```
Print "欢迎您来到 VB 世界!"
```

该行代码的作用是在窗体上输出(即显示)后面的文字。代码窗口显示如图 1.5 所示。

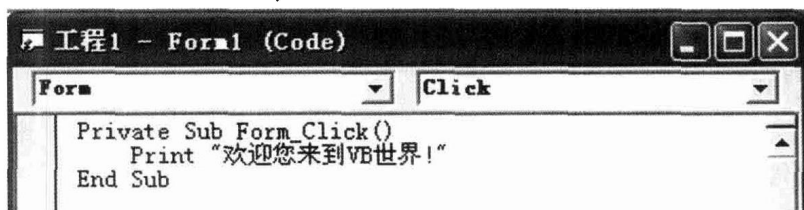


图 1.5 程序代码窗口

3. 保存工程

本例中只涉及一个窗体 Form1,因此,只需保存一个窗体文件和一个工程文件。保存文件的步骤如下:

(1) 选择“文件”菜单中的“Form1 另存为”命令,系统弹出“文件另存为”对话框,选择好保存位置(如“我的文档”文件夹)后输入文件名(如 vb0101.frm),然后单击“保存”按钮,即可保存窗体文件。

注意:窗体名与窗体文件名不是同一概念。如本例,窗体名采用默认名 Form1,而窗体文件名却是 vb0101.frm。

(2) 选择“文件”菜单中的“工程另存为”命令,系统弹出“工程另存为”对话框,选择好保存

位置(如“我的文档”文件夹)后输入文件名(如 vb0101.vbp),然后单击“保存”按钮。

4. 运行程序

单击工具栏上的“启动”按钮,或选择“运行”菜单中的“启动”命令,即可用解释方式运行程序。程序运行时显示一个空白窗体,当用户用鼠标单击该窗体时,就会发生单击窗体事件,系统会自动执行 Form_Click 事件过程,从而在窗体上输出“欢迎您来到 VB 世界!”字样,如图 1.6 所示。

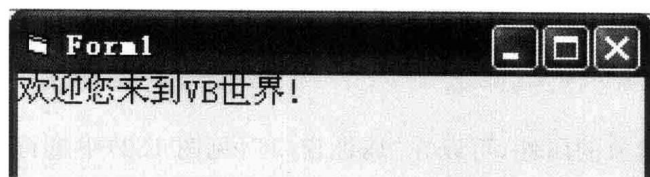


图 1.6 单击窗体时的显示信息

单击窗体右上角的“关闭”按钮,或单击工具栏上的“结束”按钮,即可结束程序的运行。

1.2.3 第二个简单程序

【例 1.2】 设计一个应用程序,由用户输入一个数,计算并输出该数的平方数。

分析:要创建的应用程序用户界面如图 1.7 所示。

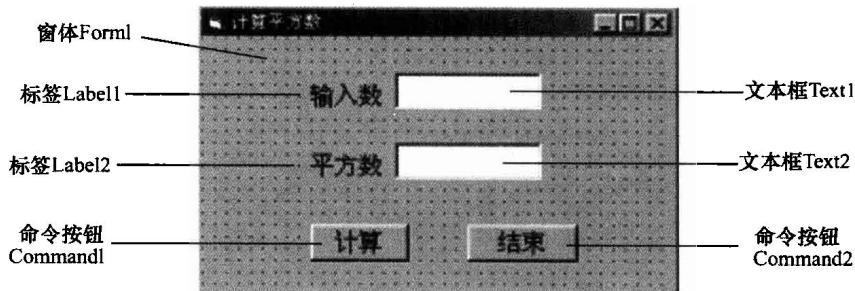


图 1.7 例 1.2 的设计界面

窗体上含有 6 个控件:2 个标签、2 个文本框和 2 个命令按钮。标签用于显示文字;文本框用于输入数据,也可显示数据;命令按钮用来执行有关操作。运行程序时,用户在“输入数”文本框中输入数据,当单击“计算”命令按钮时,则计算该数的平方数并显示在“平方数”文本框中。单击“结束”按钮,则结束程序的运行。

具体设计步骤如下。

1. 创建窗体

启动 VB 或选择“文件”菜单中的“新建工程”命令,从“新建工程”对话框中选择“标准 EXE”,系统会默认提供一个窗体(Form1)。用户可在此窗体上添加控件,以构建用户界面。

2. 在窗体上添加控件

设置控件的方法：在 VB 工具箱(如图 1.2 所示)中选择(单击)要添加的控件的按钮，此时鼠标指针变成“+”字形。将“+”字形指针移到窗体的适当位置，然后按下左键并拖动鼠标，可按所需大小放置一个控件。按照上述方法，可在窗体上添加以下控件：

- (1) 通过工具“Label”(图标“A”)放置两个标签框(简称标签)。
- (2) 通过工具“TextBox”(图标“abl”)放置两个文本框。
- (3) 通过工具“CommandButton”(图标)放置两个命令按钮。

3. 设置对象属性

设置窗体上控件对象的属性，可以在“属性窗口”(见图 1.2)中进行。通常，属性窗口(标题栏上显示有“属性—”)处于主窗口的右侧中部，用户也可以选择“视图”菜单中的“属性窗口”命令来显示属性窗口。

设置对象属性的方法：用鼠标单击窗体上要设置属性的对象，使其处于选定状态。此时属性窗口中会自动显示该对象的属性列表框，列表框左半边显示所选对象的所有属性名，右半边显示属性值。找到需设置的属性，然后对该属性值进行设置或修改。按照上述方法，可以设置以下对象的属性：

- (1) 设置窗体 Form1 的 Caption(标题名)属性为“计算平方数”。
- (2) 设置标签 Label1 的 Caption 属性为“输入数”。
- (3) 设置标签 Label2 的 Caption 属性为“平方数”。
- (4) 设置文本框 Text1 的 Text(文本内容)属性为空白。
- (5) 设置文本框 Text2 的 Text 属性为空白。
- (6) 设置按钮 Command1 的 Caption 属性为“计算”。
- (7) 设置按钮 Command2 的 Caption 属性为“结束”。

4. 编写程序代码，建立事件过程

(1) 双击当前窗体，或选择“视图”菜单中的“代码窗口”命令，系统弹出如图 1.3 所示的代码窗口。

(2) 按照例 1.1 中介绍的方法，输入命令按钮 Command1 的单击事件过程(Command1_Click)代码：

```
Private Sub Command1_Click()  
    Dim x As Single  
    x = Val(Text1.Text)  
    Text2.Text = x * x  
End Sub
```

说明：① Dim 语句的作用是定义(也称声明)一个数据类型为 Single(即单精度)的变量 x；② Val(Text1.Text) 的作用是将文本框 Text1 中的数字字符(用户输入的)转换为数