

VB 程序设计 简明教程

林卓然 编著

中山大学出版社

·广州·

版权所有 翻印必究

图书在版编目(CIP)数据

VB 程序设计简明教程/林卓然编著. —广州:中山大学出版社, 2002. 1
ISBN 7-306-01857-4

I. V…

II. 林…

III. BASIC 语言 - 程序设计 - 教材

IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 083773 号

中山大学出版社出版发行

(地址:广州市新港西路 135 号 邮编:510275

电话:020-84111998, 84037215)

广东新华发行集团股份有限公司经销

广州市番禺区市桥印刷厂印刷

(地址:广州番禺区市桥环城西路 201 号 邮编:511400 电话:020-84881937)

787 毫米×1092 毫米 16 开本 14.25 印张 358 千字

2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷

印数:1-5 000 册 定价:19.00 元

如发现因印装质量问题影响阅读,请与承印厂联系调换

内 容 简 介

本书是一本 Visual BASIC 程序设计的入门教材。它以 VB 6.0 为语言背景，介绍面向对象的初步概念和可视化程序设计的基本方法，内容包括：VB 语言基础，数据输入输出，分支结构设计，循环结构设计，数组，过程，数据文件的存取，VB 在网页制作中的应用，程序调试等。

本书叙述上力求深入浅出，简明易懂。各章后面配有练习题和上机实习题。

本书可作为高等院校本、专科各专业有关课程的教材，也可作为学习和开发 VB 应用程序的参考书。

前 言

Visual BASIC (简称 VB) 是 Microsoft 公司推出的一种 Windows 应用程序开发工具, 它既继承了 BASIC 语言具有的简单易学、操作方便等特点, 又引入了面向对象的编程机制和可视化程序设计方法, 大大降低了开发 Windows 应用程序的难度, 有效地提高了应用程序开发的效率。因此, VB 在国内外各个领域应用十分广泛, 已经成为普通用户首选的程序设计语言。

本书以 VB 6.0 为语言背景, 讲授程序设计的基本思想和基本方法, 重点培养学生使用可视化技术进行编程和程序设计的能力。因为 VB 程序设计涉及的内容比较多, 既有控件的使用及用户界面的设计, 又有程序代码的编写, 为了突出重点, 本书把使用 VB 控件分散到各个章节来介绍, 使之与程序设计紧密结合起来。书中还安排了大量的例题和上机实习题, 帮助读者边学边用, 以提高学习效果。

本书主要根据广东省高校计算机水平考试《VB 程序设计》考试大纲而编写的。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, 包括各章节的电子讲稿、应用程序文件及相关素材文件, 并发布在中山大学计算机基础教育中心的教学网站上, 其网址为:

<http://cec.zsu.edu.cn>

老师们可进入“课程介绍”栏目, 单击本书书名, 然后从打开的文件夹中下载所需文件。需要本书习题解答的老师, 可直接与编者联系, 编者将通过电子邮件发送。同学们也可以进入“文虎考堂”栏目, 通过该栏目提供的网上考试系统来进行自我测试。

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。

编者电子邮件地址: Puzlzl@zsu.edu.cn

编著者

2001 年 11 月

目 录

第一章 Visual BASIC 概述	(1)
1.1 VB 简介	(1)
1.1.1 BASIC 语言的发展	(1)
1.1.2 VB 的主要特点	(2)
1.1.3 VB 的启动和关闭	(2)
1.2 设计 VB 应用程序的初步认识	(4)
1.2.1 设计 VB 应用程序的大致步骤	(4)
1.2.2 第一个简单程序	(4)
1.2.3 第二个简单程序	(6)
1.3 对象与事件的基本概念	(8)
1.3.1 对象、类与方法	(9)
1.3.2 事件、事件过程及事件驱动	(10)
1.4 VB 的集成化开发环境	(11)
1.4.1 窗体编辑器窗口	(11)
1.4.2 工具箱和控件	(11)
1.4.3 工程资源管理器窗口	(14)
1.4.4 属性窗口	(14)
1.4.5 代码窗口	(15)
1.4.6 窗体布局窗口	(15)
1.5 工程的管理	(15)
1.5.1 工程中的主要文件类型	(16)
1.5.2 创建、打开和保存工程	(16)
1.5.3 在工程中添加窗体和模块	(17)
1.5.4 从工程中删除文件	(17)
1.5.5 生成可执行文件	(17)
习题	(18)
编程及上机调试	(19)
第二章 VB 语言基础	(20)
2.1 基本数据类型	(20)
2.2 变量与常量	(22)
2.2.1 变量	(22)

2.2.2 常量	(23)
2.3 表达式	(23)
2.3.1 算术表达式	(23)
2.3.2 字符串表达式	(24)
2.3.3 日期表达式	(25)
2.4 常用内部函数	(25)
2.4.1 数学函数	(25)
2.4.2 字符串函数	(27)
2.4.3 日期与时间函数	(28)
2.4.4 类型转换函数	(29)
2.5 编写代码规则	(30)
2.5.1 程序语句及语法	(30)
2.5.2 代码书写规则	(30)
2.5.3 结构化程序的基本结构	(31)
习题	(32)
编程及上机调试	(33)
第三章 赋值与输入输出	(35)
3.1 赋值语句	(35)
3.2 数据输出	(37)
3.2.1 Print 方法	(37)
3.2.2 特殊打印格式	(39)
3.2.3 位置属性和字体属性	(40)
3.3 常用基本语句	(42)
3.3.1 卸载语句 Unload	(42)
3.3.2 结束语句 End	(42)
3.3.3 暂停语句 Stop	(42)
3.3.4 强制显式声明变量语句 Option Explicit	(43)
3.3.5 清除方法 Cls	(43)
3.4 窗体	(44)
3.4.1 窗体的基本结构	(45)
3.4.2 窗体的基本属性	(45)
3.4.3 窗体的事件	(45)
3.4.4 窗体的方法	(46)
3.5 利用文本框输入数据	(47)
3.5.1 使用文本框控件	(48)
3.5.2 焦点与 Tab 键序	(49)
3.6 使用对话框	(49)

3.6.1 输入对话框·····	(50)
3.6.2 消息对话框·····	(50)
习题·····	(53)
编程及上机调试·····	(54)
第四章 分支结构设计·····	(56)
4.1 关系表达式和逻辑表达式·····	(56)
4.1.1 关系表达式·····	(56)
4.1.2 逻辑表达式·····	(57)
4.2 条件语句·····	(58)
4.2.1 If...Then 语句·····	(58)
4.2.2 If...Then...Else 语句·····	(58)
4.2.3 If 函数和 Choose 函数·····	(60)
4.2.4 条件语句的嵌套·····	(61)
4.3 多分支语句·····	(65)
4.4 选择性控件·····	(67)
4.4.1 单选按钮·····	(68)
4.4.2 复选框·····	(69)
4.5 计时器控件·····	(71)
习题·····	(73)
编程及上机调试·····	(75)
第五章 循环结构设计·····	(76)
5.1 For...Next 循环语句·····	(76)
5.2 Do...Loop 循环语句·····	(79)
5.2.1 前测型 Do...Loop 循环·····	(79)
5.2.2 后测型 Do...Loop 循环·····	(80)
5.3 循环出口语句·····	(81)
5.4 多重循环·····	(83)
5.5 列表框与组合框·····	(85)
5.5.1 列表框·····	(86)
5.5.2 组合框·····	(89)
5.6 程序举例·····	(91)
习题·····	(96)
编程及上机调试·····	(99)
第六章 数组·····	(101)
6.1 数组的概念·····	(101)

6.1.1 数组的引入	(101)
6.1.2 数组与数组元素	(101)
6.1.3 下标和数组的维数	(102)
6.2 静态数组和动态数组	(102)
6.2.1 静态数组	(103)
6.2.2 动态数组	(107)
6.3 For Each...Next 循环语句	(108)
6.4 控件数组	(109)
6.4.1 控件数组的概念	(109)
6.4.2 控件数组的建立	(110)
6.4.3 控件数组的使用	(110)
6.5 程序举例	(111)
习题	(118)
编程及上机调试	(122)
第七章 过程	(124)
7.1 通用过程	(124)
7.1.1 Sub 过程	(124)
7.1.2 Function 过程	(128)
7.1.3 Sub Main 过程与多窗体	(130)
7.2 参数传递	(132)
7.2.1 形参与实参	(132)
7.2.2 按地址传递和按值传递	(134)
7.3 变量的作用域	(135)
7.3.1 局部变量	(135)
7.3.2 模块级变量	(136)
7.3.3 全局变量	(136)
7.4 过程的嵌套与递归调用	(137)
7.4.1 过程的嵌套调用	(137)
7.4.2 过程的递归调用	(137)
7.5 程序举例	(139)
习题	(145)
编程及上机调试	(148)
第八章 数据文件的存取	(150)
8.1 数据文件的基本概念	(150)
8.1.1 数据文件的组成	(150)
8.1.2 文件类型	(150)

10.3.3	菜单的 Click 事件	(183)
10.3.4	运行时改变菜单属性	(186)
10.3.5	快捷菜单	(186)
10.4	工具栏和状态栏	(187)
10.4.1	Active X 控件简介	(187)
10.4.2	工具栏 (ToolBar)	(188)
10.4.3	状态栏 (StatusBar)	(191)
10.5	对话框	(192)
10.5.1	通用对话框 (CommonDialog)	(193)
10.5.2	自定义对话框	(194)
10.6	滚动条	(195)
10.7	文件系统控件	(196)
10.8	控件拖放	(198)
10.9	多媒体应用	(200)
10.9.1	使用 Animation 控件	(200)
10.9.2	使用 Multimedia 控件	(200)
10.10	多文档界面 (MDI)	(202)
第十一章	在 Web 网页制作中使用 VBScript	(203)
11.1	HTML 语言简介	(203)
11.1.1	HTML 文档的基本结构	(203)
11.1.2	常用的一些标记	(204)
11.2	网页脚本语言 VBScript	(205)
11.2.1	脚本的概念	(205)
11.2.2	在 HTML 文档中加入 VBScript 代码	(205)
11.2.3	编制动态网页	(207)
11.3	ASP 网页制作	(210)
11.3.1	基本概念	(210)
11.3.2	编写 ASP 程序	(210)
	编程及上机调试	(212)
附录一	ASCII 码	(213)
附录二	颜色代码	(213)

第一章 Visual BASIC 概述

在深入学习 Visual BASIC (简称 VB) 之前, 本章先介绍一下 VB 编程的一些基本概念。

1.1 VB 简介

1.1.1 BASIC 语言的发展

BASIC 自 1964 年问世以来, 由于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 新版本应运而生。1975 年前后开发出 DOS 环境下的 GW - BASIC, 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。1991 年美国微软公司 (Microsoft) 推出 VB1.0 版本, 之后该公司又相继推出 VB2.0、VB3.0、VB4.0, 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 年微软公司推出 VB5.0, 它是一个 32 位应用程序的开发工具, 可以运行在 Windows 9.X 或 Windows NT 环境中。1998 年微软公司发布了 VB 的最新版本 VB 6.0。

VB 中的 “Visual” 在字面上的含义是 “看见的、视觉的”, 因此它被称为 “可视化” 程序设计语言。它为开发 Windows 应用程序而提供了强有力的开发环境和工具, 并具有很好的图形用户界面 (GUI)。VB 采用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集合在一起。应用 VB 可以方便、快捷地完成 Windows 应用程序的开发, 大大地提高了应用程序的开发效率。

如今 VB 的功能已十分强大, 它可以实现 Windows 的绝大部分功能, 如多任务、多文档界面 (MDI)、对象的链接与嵌入 (OLE)、动态数据交换、动态链接库 (DLL)、子程序的调用等, 尤其是动态链接技术, 使得 VB 可以调用 Windows 系统的各种资源。应用 VB 可以方便地完成小型应用程序, 到大型的数据库应用系统、多媒体信息处理, 甚至通过 Internet 访问遍及全球的分布式应用程序等各项任务。

VB 6.0 提供了学习版、专业版和企业版。学习版是 VB 6.0 的基本版, 主要是为初学者学习而设计的。专业版为专业编程人员提供了功能完备的开发工具, 专业版中包含了学习版的所有功能。企业版除包括专业版的全部功能外, 还包括一个 Back Office 工具, 它使得专业编程人员能够创建更高级的分布式高性能应用程序。

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 的主要特点

VB 是一种以对象为基础，以事件驱动作为运行机制的可视化程序设计语言，它的两个主要特点是：

1. 可视化的设计平台

采用传统的程序设计语言编程时，一般需要通过编写程序来设计程序的界面（如界面元素的外观和位置），在设计过程中看不到程序界面的实际显示效果。VB 提供可视化设计平台，把 Windows 界面设计的复杂性“封装”起来，程序员只需从 VB 工具箱中选用界面元素（称为控件，如文本框、命令按钮等），放到窗口上的相应位置，即可设计图形用户界面。这种设计方法就好比目前盖房子，所用的门、窗、水泥、钢筋、砖等都是现成的材料，而用传统的 BASIC 就如同早先的工匠，一砖、一瓦、一木都要自己动手。

VB 与 Visual C++、Visual FoxPro 等一样，都是可视化开发工具。这些工具的共同特点是，提供了 Windows 界面下一些常用界面元素。所谓“可视化程序设计”，它指的是应用可视化开发工具，开发图形用户界面应用程序的方法。

2. 事件驱动的编程机制

传统的程序设计语言程序由一个主程序和若干个过程及函数组成，程序运行时总是从主程序开始，由主程序调用各过程和函数。程序设计者在编程时必须将整个程序的执行顺序十分精确地设计好。程序按指定的过程执行。因此传统的语言称为面向过程的语言。

VB 不同于传统的面向过程的设计语言，它将应用程序划分成相互关联的多个对象，并且建立起与这些对象相关的事件过程，通过用户操作触发某个事件来执行相应的事件过程，从而达到运算、处理的目的。

一个 VB 程序包含若干个过程，但它没有传统意义上的主程序。每个事件过程都由相应的“事件”触发而执行的。各事件的发生顺序是任意的。这样就使程序设计工作变得比较简单。人们只需针对一个事件编写一段过程即可。

1.1.3 VB 的启动和关闭

1. VB 的启动

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：单击“开始”按钮，从开始菜单中选择“程序”，再选择“Microsoft Visual Studio 6.0 中文版”级联菜单中的“Microsoft Visual Basic 6.0 中文版”程序，就可以启动 VB。

当然也可将 VB 程序以快捷方式放在桌面上，这样就可以直接在桌面上双击该图标来启动它。

启动 VB 后，做为默认方式，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（默认类型为“标准 EXE”）。

(2) 现存：供选择和打开现有的工程。

说明：一个 VB 应用程序称为一个工程。

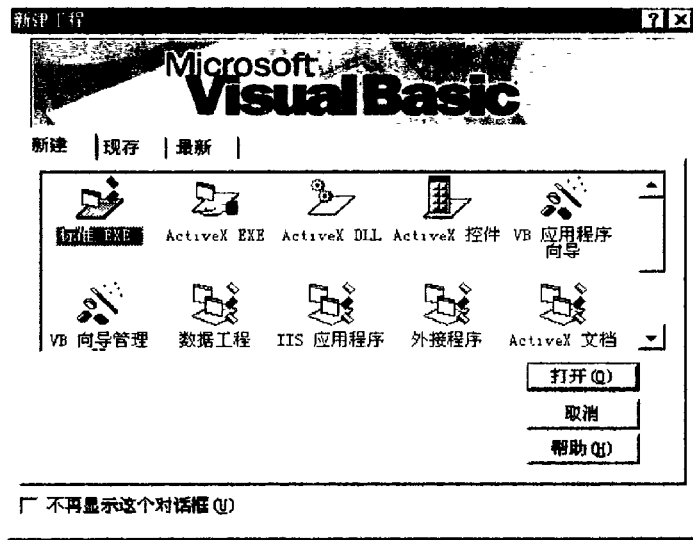


图 1.1 “新建工程”对话框

(3) 最新：列出最新使用过的工程。

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。

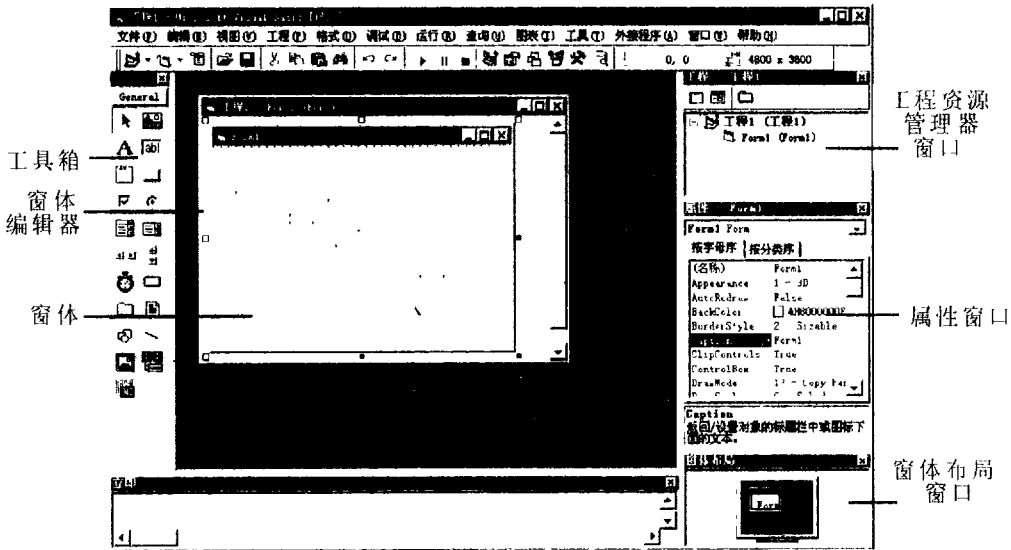


图 1.2 VB 集成开发环境

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。

1.2 设计 VB 应用程序的初步认识

1.2.1 设计 VB 应用程序的大致步骤

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 采用面向对象的编程机制，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“VB 欢迎您”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（图 1.2）。

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（见图 1.2）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 form1 窗体相对应的代码窗口，如图 1.3 所示。

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

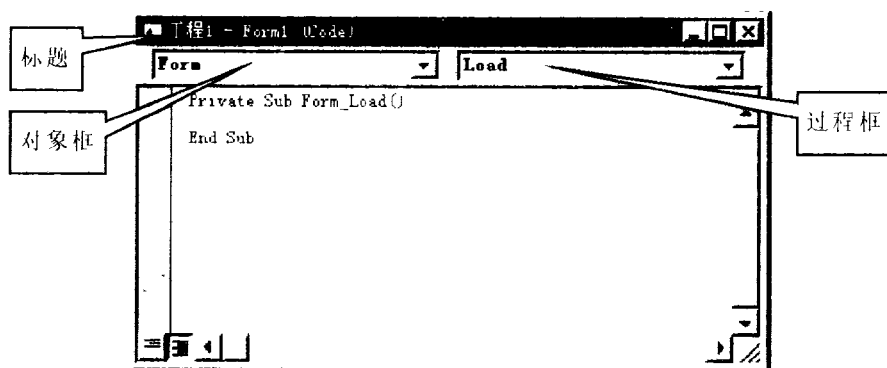


图 1.3 代码窗口

事件后，在代码窗口的编辑区中立即自动出现 Form_Click 事件过程的模板，如图 1.4 所示。

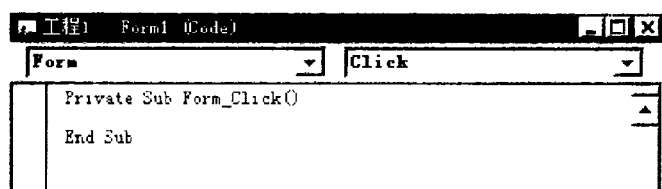


图 1.4 Form_Click 事件过程框架

说明：① 关键字 Private（中文含义是“私用”）表示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。

在已有两行代码之间插入一行代码，即

```
Print "VB 欢迎您!"
```

代码窗口显示如图 1.5 所示。

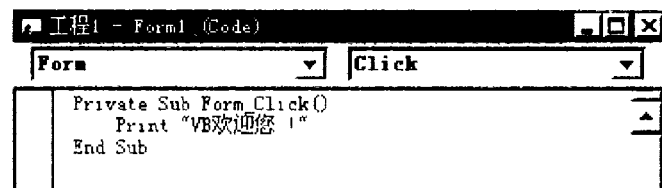


图 1.5 程序代码

3. 保存工程

本例中只涉及一个窗体（系统默认提供的 Form1），因此，只需保存一个窗体文件和工程文件。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），输入文件名（如 Vb0101.frm），然后单

击“保存”按钮，即可保存窗体文件。

(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），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“VB 欢迎您！”字样，如图 1.6 所示。



图 1.6 单击窗体时显示的情况

单击工具栏上的“结束”按钮，或选择“运行”菜单中的“结束”命令，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，使之能输入一个数，然后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。窗体上含有两个标签、两个文本框和两个命令按钮。两个标签分别用于显示文字“输入数”和“平方数”，两个文本框用于输入数据和显示计算结果。运行时，用户在“输入数”文本框中输入某一个数，当单击“计算”按钮时，则在“平方数”文本框中显示该数的平方数。单击“结束”按钮，则结束程序的运行。

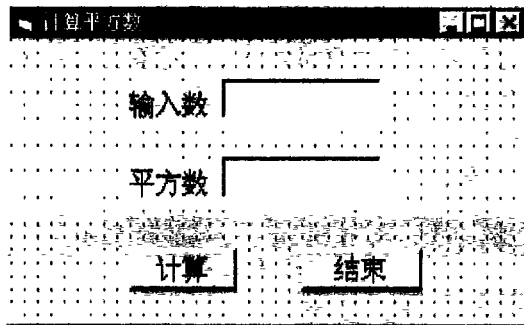


图 1.7 例 1.2 的用户界面

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. 在窗体上添加界面的控件

设置控件的方法：在 VB 工具箱中选择（单击）要添加的控件的按钮，此时鼠标指针

变成“+”字型，将“+”字型指针移到 Form 窗体的适当位置，然后按下左键并拖动鼠标，则可按所需大小画出一个控件。按照上述方法，可在 Form1 窗体上添加以下控件：

(1) 通过工具“Label”（图标“A”）画出两个标签框（简称标签）：

标签 Label1：用于显示文字“输入数”；

标签 Label2：用于显示文字“平方数”。

(2) 通过工具“TextBox”（图标“abl”）画出两个文本框：

文本框 Text1：用于输入数；

文本框 Text2：用于显示计算结果（平方数）。

(3) 通过工具“CommandButton”（图标□）画出两个命令按钮：

命令按钮 Command1：用于计算输入数的平方，并把结果显示在文本框 Text2 中；

命令按钮 Command2：用于结束应用程序的运行。

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”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，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。

(5) 设置文本框 Text2 的 Text 属性为空。

(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 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 上述 Dim 语句的作用是声明变量 x，定义 x 的数据类型为 Single（即单精度）；② Text1.Text 表示文本框 Text1 中的文本内容，若用户在文本框 Text1 中输入数据