

Visual Basic

程序设计教程

田希山 师金华 主编

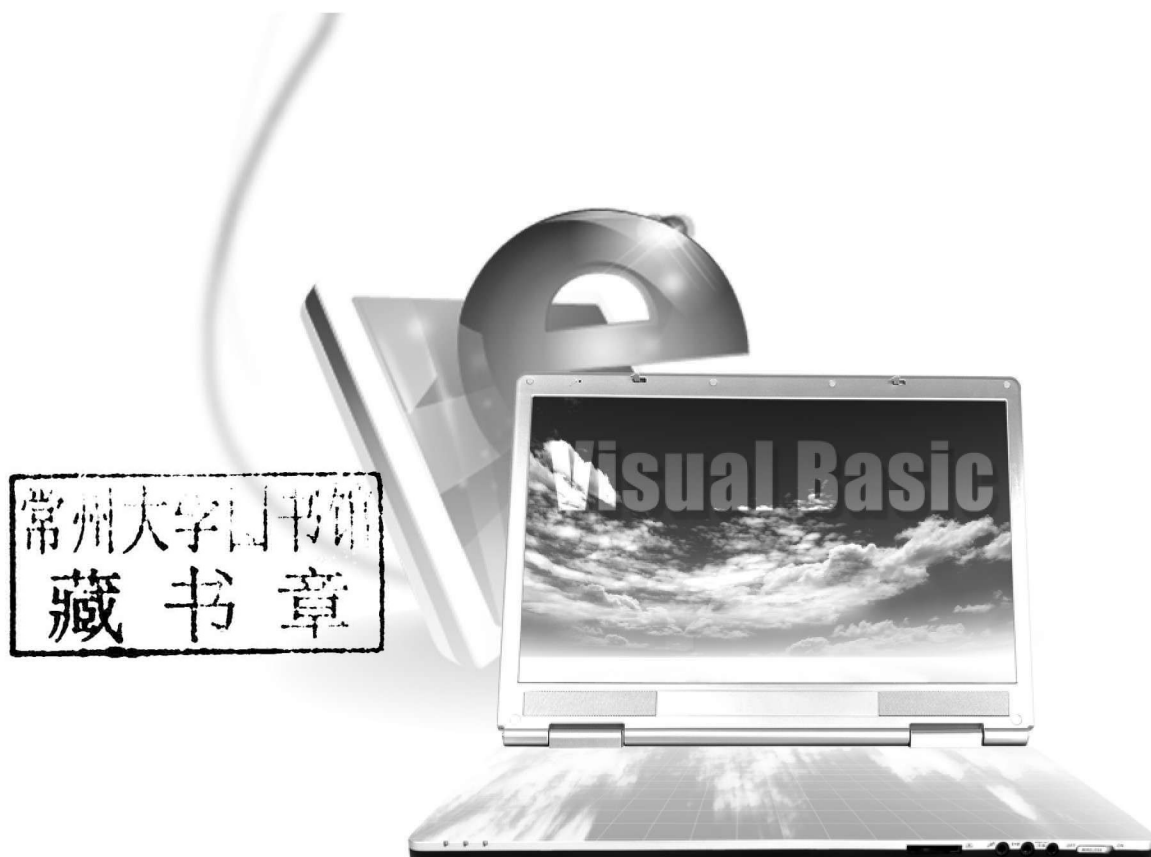


黄河出版传媒集团
宁夏人民出版社

Visual Basic

程序设计教程

田希山 师金华 主编



黄河出版传媒集团
宁夏人民出版社

图书在版编目 (CIP) 数据

Visual Basic 程序设计教程 / 田希山, 师金华主编.

—银川: 宁夏人民出版社, 2012. 12

ISBN 978-7-227-05369-9

I. ①V… II. ①田…②师… III. ①BASIC 语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字 (2012) 第 309913 号

Visual Basic 程序设计教程

田希山 师金华 主编

责任编辑 丁 佳 马宗明

封面设计 海 飞

责任印制 张国祥

黄河出版传媒集团
宁夏人民出版社 出版发行

地 址 银川市北京东路 139 号出版大厦 (750001)

网 址 www.yrpubm.com

网上书店 www.hh-book.com

电子信箱 renminshe@yrpubm.com

邮购电话 0951- 5044614

经 销 全国新华书店

印刷装订 宁夏捷诚彩色印务有限公司

开 本 880mm×1230mm 1/16 印 张 19.25 字 数 470 千

印刷委托书号 (宁) 0010720 印 数 2000 册

版 次 2012 年 12 月第 1 版 印 次 2012 年 12 月第 1 次印刷

书 号 ISBN 978- 7- 227- 05369- 9/TP·17

定 价 35.00 元

版权所有 侵权必究

目 录

第 1 章 Visual Basic 程序设计入门	1
1.1 Visual Basic 语言	1
1.2 VB 集成开发环境	10
1.3 窗 体	20
1.4 练 习	23
第 2 章 程序设计基础	26
2.1 数据类型	26
2.2 常量与变量	29
2.3 运算符与表达式	34
2.4 常用内部函数	36
2.5 程序语句	40
2.6 练 习	42
第 3 章 顺序结构程序设计	46
3.1 数据的输出	46
3.2 数据的输入	59
3.3 常用的其他基本语句	69
3.4 使用框架控件	71
3.5 练 习	72
第 4 章 选择结构程序设计	76
4.1 条件表达式	76
4.2 单条件选择语句 If	78

4.3 多分支条件选择语句 Select Case	89
4.4 相关控件	94
4.5 习 题	103
 第 5 章 循环结构程序设计	108
5.1 循环结构的概念	108
5.2 循环结构	108
5.3 列表框控件和组合框控件	124
5.4 习 题	136
 第 6 章 数组	146
6.1 数组的概念	146
6.2 一维数组	148
6.3 二维数组	153
6.4 多维数组的声明和引用	157
6.5 动态数组	157
6.6 数组的特殊操作	161
6.7 控件数组	163
6.8 记录类型和枚举类型	167
6.9 习 题	171
 第 7 章 过程	179
7.1 事件过程与通用过程	179
7.2 过程的建立和调用	179
7.3 参数的传递	188
7.4 过程的嵌套与递归	195
7.5 变量的作用域与生存期	198
7.6 过程的作用域	202
7.7 习 题	203
 第 8 章 图形设计	211
8.1 图形设计基础	211
8.2 图形属性与事件	216

8.3 形状控件和直线控件	220
8.4 绘图方法	222
8.5 习 题	232
 第 9 章 界面设计	 234
9.1 菜单的设计	234
9.2 工具栏设计	240
9.3 对话框设计	244
9.4 多文档界面	255
9.5 习 题	257
 第 10 章 文件	 261
10.1 文件概念	261
10.2 文件操作语句与函数	263
10.3 文件系统控件	269
10.4 文件的基本操作	272
10.5 文件系统对象	280
10.6 习 题	288
 第 11 章 数据库访问技术	 295
11.1 数据库的基本概念	295
11.2 Data 控件	301
11.3 ADO 数据对象访问技术	305
11.4 制作报表	314
11.5 习 题	322

第 1 章 Visual Basic 程序设计入门

1.1 Visual Basic 语言

Basic 是英文 Beginner's All-Purpose Symbolic Instruction Code(初学者通用符号指令代码)的缩写,是专门为初学者设计的计算机语言。它诞生于 1964 年,最初问世时只有几十条语句,由于简单易学而受到了用户的普遍欢迎,很快就应用于各个领域。

Visual 的意思是“视觉的,可视的”,也就是直观的编程方法,Visual Basic(简称为 VB)是 Microsoft 公司为开发 Windows 应用程序而提供的强有力的开发环境和工具,具有很好的图形用户界面(Graphic User Interface,简称为 GUI)的程序设计语言。VB 应用程序的开发以对象为基础,并运用事件驱动机制实现对 Windows 操作系统的事件响应。VB 提供了大量控件,可用于设计界面和实现各种功能,用户可以通过拖放操作完成界面设计,不仅大大减轻了工作量、简化了界面设计过程,而且有效地提高了应用程序的运行效率与可靠性。

1.1.1 Visual Basic 的特点

1. 可视化设计平台

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。程序员不必再为界面的设计而编写大量的程序代码,只需按程序设计的要求,用系统提供的工具在屏幕上“画出”各种对象,VB 将自动产生界面设计代码,程序员的编程工作仅编写的是实现程序功能的那部分代码,也就是针对某一“对象”要完成的事件过程的代码,从而大大提高了编程的效率。

2. 面向对象的设计方法

VB 采用了面向对象的程序设计思想。它的基本思路是把复杂的程序设计问题分解为一个个能够完成独立功能的相对简单的对象集合,所谓“对象”就是一个可操作的实体,如窗体、窗体中的命令按钮、标签、文本框等。面向对象的编程就好像小朋友搭积木一样,程序员可以根据程序和界面设计要求,直接在屏幕上“画出”窗口、菜单、按钮等不同类型的对象,并为每个对象设置相应的属性。

3. 事件驱动的编程机制

在 Windows 环境下,程序是以事件驱动方式运行的,每个对象都能响应多个不同的事件。所谓事件,是指 VB 预先设置的、能够被对象识别的动作。对象感应到某一事件发生时所执行的程序代码称为事件过程。传统的程序设计语言面向过程,而 VB 语言面向对象,通过事件来触发执行相应的事件过程

(“事件驱动”),从而执行对象的操作以便达到运算、处理的目的。例如,单击一个命令按钮,就触发了按钮的 Click(单击)事件,该事件中的代码就会被执行。若用户未进行任何操作(未触发事件),则程序就处于等待状态。整个应用程序就是由彼此独立的事件过程构成。

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,具有高级程序设计语言的语句结构,接近于自然语言和人类的逻辑思维方式,语句简单易懂,它的编辑器可以自动进行语法错误检查。因此更是一种结构化的程序设计语言。

5. 集成式的开发环境

VB 为编程提供了一个集成开发环境。在这个环境中,编程者可设计界面、编写代码、调试程序,直至把应用程序编译成可在 Windows 环境中运行的可执行文件,并为它生成安装程序。VB 的集成开发环境为编程者提供了很大的方便。

6. 充分利用 Windows 资源

VB 提供的动态数据交换 (DDE) 的编程技术,可以在应用程序中实现与其他 Windows 应用程序建立动态数据交换、在不同应用程序之间进行通信。VB 提供的对象链接与嵌入 (OLE) 技术将每个应用程序都看作一个对象,将不同的对象连接起来,从而得到具有声音、图像、动画、文字等各种信息的集合式文件。另外,VB 提供的动态链接库技术,可方便地调用 C/C++ 语言或汇编语言编写的函数,也可调用 Windows 的应用程序接口 (.API) 函数。

7. 开放的数据库功能与网络支持

VB 具有很强的数据库管理功能,不仅可以管理 MS Access 格式的数据库,还能访问其他外部数据库,如 FoxPro、Paradox 等格式的数据库。另外,VB 还提供了开放式数据库连接 (ODBC) 功能,可以通过直接访问或建立链接的方式使用并操作后台大型网络数据库,如 SQL Server、Oracle 等。在应用程序中,可以使用结构化查询语言 (SQL) 直接访问 Server 上的数据库,并提供简单的面向对象的库操作命令、多用户数据库的加锁机制和网络数据库的编程技术,为单机上运行的数据库提供 SQL 网络接口,以便在分布式环境中快速而有效地实现客户/服务器 (Client/Server) 方案。

1.1.2 面向对象的程序设计

传统的编程方法使用的是面向过程、按顺序进行的机制,其缺点是程序员始终要关心什么时候要发生什么事情,所以处理 Windows 环境下的事件驱动方式工作量太大。VB 采用的面向对象、事件驱动的编程机制,程序员只需要编写相应用户动作的程序,如:移动鼠标、单击鼠标等,而不必考虑每个步骤执行的精确次序,编写的代码相对较少。

VB 使用的可视化编程方法,是面向对象编程技术的简化版。它提供多种控件支持可视化编程,利用它们可以快速创建强大的应用程序用户界面而无需涉及不必要的细节。

在 VB 环境中,程序员不仅可以利用控件来创建对象,而且还可以建立自己的控件,这是 Windows 环境下编程的新概念。

1. 对象、属性和方法

(1) 对象

可以把对象 (Object) 想象成日常生活中的各种物体,例如:一只气球、一把椅子、一部手机等都是

对象。

而在 VB 程序中,对象是 VB 应用程序的基础构件,是指 VB 可以访问的实体。

如工具箱中的控件、窗体、菜单、应用程序的部件、外部文件、程序中的变量以及数据库等都是对象。这些对象都具有属性(数据)和行为方式(方法)。

(2) 属性

每个对象都有一组特征,属性就是对象的特征也就是对象的性质。不同的对象有不同的属性,属性值决定了对象的外观和行为。

对于 VB 中的每一个对象而言,都有一组特定的属性。有些属性属于多数对象的公共属性,如 BackColor 属性定义对象的背景颜色;有些属性则是属于特定对象的专有属性,如只有计时器对象才有 Interval 属性,该属性用来确定计时器控件两次报时的时间间隔。

每一个对象属性都有一个缺省值(默认值),如果不明确的改变该值,程序就使用它。通过修改对象的属性能够控制对象的外观。对象属性的设置一般有两条途径:

① 如果在界面设计时设置对象的属性,需要使用属性窗口。这时只要在属性窗口的左列中选中要修改的属性,然后在右列中键入新的属性值即可设置对象的属性。

② 如果要在程序运行中动态地更改对象的属性,可以使用 VB 的赋值语句(第三章会详细讲解)在代码窗口中通过编程设置,其格式为:

对象名.属性名 = 属性值

其中“对象名.属性名”是 VB 中引用对象属性的方法,如下述代码可以设置标签控件 Label1 的标题为“欢迎学习 VB 程序设计”:

```
Label1.Caption = "欢迎学习 VB 程序设计"
```

(3) 方法

方法就是要执行的动作,它是规定好的用于完成某种特定功能的特殊过程。用户对具体的实现过程并不关心,关键是最终收到的效果。VB 的方法与事件过程类似,它可能是函数,也可能是过程,它用来完成某种特定功能而不响应某个事件。如打印(Print)方法、显示窗体(Show)方法、移动窗体(Move)方法等。每个方法都能完成某项功能,但具体的实现步骤和细节用户看不到、也不能修改,用户能做的就是按照约定调用它们。

方法只能在代码窗口中使用,其用法依赖于方法所用的参数的个数以及它是否具有返回值。下面的格式可以调用方法:

对象名.方法名 [参数]

如在图片框 Picture1 上以 (60,80) 为圆心、100 为半径画一个圆,可用如下命令:

```
Picture1.Circle( 60,80) ,100
```

Picture1 是图片框的名称,Circle 是 VB 中的方法名。

2. 对象的事件与事件过程

(1) 事件

事件(Event)就是对象所发生的事情。在 VB 中,事件是指由系统事先设定的、能被对象识别和响应的动作。

每个控件都可以对一个或多个事件进行识别和响应,如窗体的载入事件(Load)、鼠标的单击事件(Click)、鼠标的双击事件(DoubleClick)等。不同的对象能够识别不同的事件。如窗体对象能识别载入事件

(Load), 而其他对象则不可以识别这一事件。当事件发生时, VB 将检测两条信息, 即发生的是哪种事件和哪个对象接受了事件。总之, 事件是一种预先定义好的特定动作, 由用户或系统激活, 在多数情况下, 事件是通过用户的交互操作产生的。

(2) 事件过程

事件过程是用来完成事件发生后所要执行的操作。

当某一个对象察觉到某一事件发生时(如 Click 等), 就会对事件产生响应, 即执行一段程序代码, 所执行的这段程序代码就称为事件过程。

通常, VB 的控件都可以识别一个或一个以上的事件, 所以对一个对象至少能够建立和使用一个事件过程, 来对用户或系统的事件做出相应的反应。但是不一定每个事件过程对应用程序都是必需的, 只要根据应用程序的需要来建立需要的事件过程就可以了。

(3) 事件驱动程序设计

程序开始执行时, 先等待某个事件的发生, 然后再去执行处理此事件的事件过程。事件过程要经过事件触发才会被执行, 这种动作模式就称为事件驱动程序设计(Event Driven Programming)。也就是说, 由事件控制整个程序的执行流程。

VB 程序的执行是由事件来驱动的, 以下是事件驱动应用程序的典型工作过程:

- ① 启动应用程序, 装载和显示窗体。
- ② 对象等待事件的发生。
- ③ 事件发生后, 如果在相应的事件过程中存在程序代码, 就执行代码。
- ④ 应用程序等待下一次事件。

如此周而复始地执行, 直到整个程序结束。

1.1.3 Visual Basic 程序设计的步骤

VB 的对象已经被抽象为窗体和控件, 不再需要编写大量的代码去描述界面元素的外观和位置, 只需要编写相应用户动作的那部分程序代码, 大大简化了程序设计。因此, VB 可视化编程一般有三个步骤: 建立用户界面、设置对象属性和编写程序代码。

下面通过编写一个简单的 VB 应用程序, 由用户输入正方形的边长, 计算并输出正方形的面积来详细说明可视化编程的一般步骤。

1. 启动 VB

安装 VB 后, 在“开始”菜单的“程序”组中将多出一个“Microsoft Visual Basic 6.0 中文版”菜单选项, 这时就可以启动 VB 了, 步骤如下:

(1) 单击 Windows 任务栏的“开始”按钮, 从“程序”组中选择其中的“Microsoft Visual Basic 6.0 中文版”, 如图 1-1 所示。

启动 VB 后, 显示如图 1-2 所示的“新建工程”窗口, 提示用户选择应用程序类型。在选项卡“新建”中, 列出了 VB6.0 能够建立的应用程序类型; 在选项卡“现存”中, 列出了当前文件夹中所保存的工程文件; 在选项卡“最新”中列出了最近编辑过的工程文件。

(2) 一般在“新建”标签中选择默认的“标准 EXE”工程, 单击“打开”即可进入 VB6.0 应用程序集成开发环境, 如图 1-3 所示。



图 1-1 启动 VB6.0



图 1-2 “新建工程”窗口

在集成开发环境中集中了许多不同的功能,如程序设计、编辑、编译和调试等,所有的任务就是在这一环境中完成的。

设计工程直接面对的是窗体,因此用户的主要工作就是在“窗体设计器”中完成窗体的设计工作。系统默认的窗体只有一个 Form1,其窗体名称和标题属性均默认为 Form1。

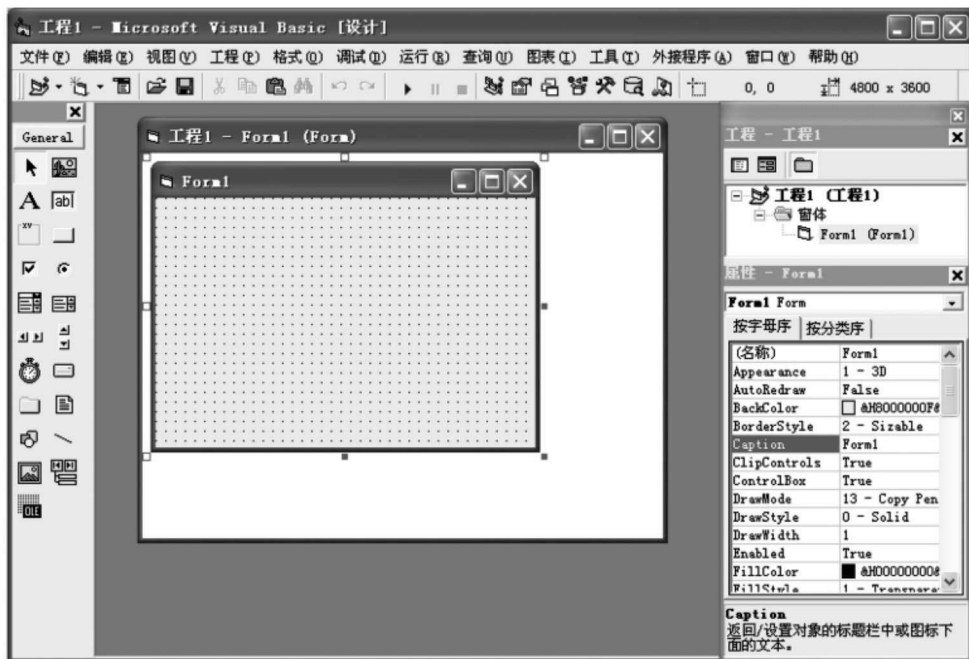


图 1-3 VB 集成开发环境

2. 界面设计

(1) 添加控件

在窗口左边的工具箱中,选中 **A** (标签控件 Label),在窗体上用鼠标拖动,画出这个控件 Label1。用同样的方法画出另外一个标签控件 Label2、两个文本框 Text1、Text2 文本框 (TextBox) 在工具箱中图标为 **abl**) 和两个命令按钮 Command1-Command2 命令按钮 (CommandButton) 在工具箱中图标为 **bl**)。若在窗体 Form1 上画出了程序所需的控件,同类型的控件序号将依次自动增加。添加控件后的程序界面如图 1-4 所示。

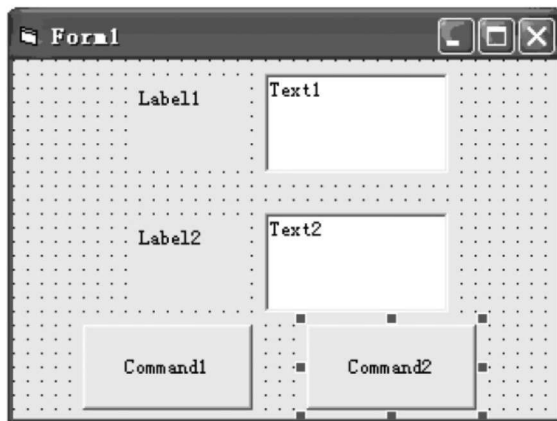


图 1-4 示例界面设计之一

(2) 设置属性

设置对象属性可以在属性窗口中进行,一般采用的操作方法是:用鼠标单击选中该对象,在该对象的属性窗口中对属性值作修改。设置属性实际上是对原有属性的修改,对象的任何属性用户不做设置时则该属性值取 VB 的缺省值(默认值)。

① 设置窗体的属性

设置窗体如 Form1 的属性的方法是:单击窗体的空白区域(不是其他任何控件对象),在属性窗口中找到标题属性 Caption,其缺省值为 Form1,可以将其值改为“计算正方形的面积”。

当然,窗体的其他属性也可根据程序的需要进行设置,如窗体的名称属性 Name、运行时窗体的背景颜色、边框风格、窗体的大小等。

② 设置窗体中其他控件的属性

单击选中某个控件,在该控件的属性窗口中根据需要逐一设置控件的各属性。

若选中标签控件 Label1,将其 Caption 属性设置为“边长”,AutoSize 属性(标签控件可以根据控件的内容自动调整大小)设置为 True”。然后用鼠标单击字体属性(Font)右边的...”按钮,从弹出的“字体”对话框(如图 1-5 所示)中设置字体类型为“宋体”、字体大小为“小五”。



图 1-5 “字体”对话框

选中标签控件 Label2,将其 Caption 属性设置为“面积”,AutoSize 属性设置为 True”,设置字体类型为“隶书”,字体大小为“三号”。



图 1-6 示例界面设计之二

选中文本框控件 Text1,将其 Text 属性设置为空字符串,设置字体类型为“黑体”、字体大小为“三号”。用同样的方法设置文本框控件 Text2。

将两个命令按钮的标题 (Caption) 属性分别设置为“计算”和“结束”。

所有控件的其他属性均采用缺省值。属性设置后的窗体如图 1-6 所示。

3. 编写代码

在“视图”菜单栏中选择“代码窗口”(打开后的代码窗口如图 1-15、1-16 所示),单击“对象”下拉列表框右边的箭头按钮,从中选择 Command1 对象。在“过程”的事件下拉列表框中选择 Click 事件,在代码窗口中输入下列代码。命令按钮的 Click 事件需要在运行中单击命令时被执行。

```
Private Sub Command1_Click( )
```

```
    Dim A As Single
```

```
    X=val( Text1.text)
```

```
    Text2.text=X*X
```

```
End Sub
```

用同样的方法,输入命令按钮 Command2 的单击 (Click) 事件过程代码:

```
Private Sub Command2_Click( )
```

```
    End
```

```
End Sub
```

事件过程的首尾两行“Private Sub Command1_Click()”和“End Sub”由系统自动给出。

程序中的逗号、引号等符号都必须是英文状态下的半角符号,中文标点符号只能出现在字符串当中。如执行语句 Print “欢迎学习 VB 教程!”,窗体上显示“欢迎学习 VB 教程!”。

4. 调试运行

单击工具栏上的“启动”按钮或按功能键 F5,即运行工程。在第一个文本框中输入正方形的边长,用鼠标单击“计算”按钮,窗体显示如图 1-7 所示,单击“结束”按钮,则结束程序的运行。



图 1-7 示例之运行结果

另外,单击工具栏上的“结束”按钮也可结束程序运行,返回“窗体设计器”窗口。

在程序运行时,可以反复输入正方形的边长值,然后单击“计算”,得到正方形的面积。

如果应用程序的运行结果不符合设计的要求,则需要修改程序。修改程序包括修改对象的属性和代码,也可以添加新的对象和代码,或者调整控件的大小等,直到满足设计需要为止。

5. 退出 VB

在退出 VB 之前,需要保存设计好的应用程序,即将程序以文件的形式保存到磁盘上,在 VB 中称为保存工程。一般是先将程序写入磁盘,然后再调试程序;当然,也可先对程序进行调试和运行,再写入磁盘。

(1) 保存工程

常用下面两种方法保存工程:

① 单击“文件”菜单中的“保存工程”或“工程另存为”命令。

② 单击工具栏上的“保存工程”按钮。

如果要保存从未保存过的新建工程,系统则打开“文件另存为”对话框,如图 1-8 所示。



图 1-8 “文件另存为”对话框

由于一个工程含有多种文件,如工程文件、窗体文件、程序模块文件等,这些文件集合在一起才能构成应用程序。因此,建议用户在保存工程时,将同一工程所有类型的文件都存放在同一文件夹中,以便日后修改和管理程序文件。在“文件另存为”对话框中首先选择保存的位置,然后选择文件夹或选中图标按钮创建新文件夹。

保存工程时,窗体文件和工程文件等需要分别保存。系统会要求用户输入文件名、选择文件的保存类型。窗体文件的保存类型为“窗体文件(*.frm)”,默认窗体文件名为“Form1”,窗体文件存盘后系统会自动弹出“工程另存为”对话框,工程文件的保存类型为“工程文件(*.vbp)”,默认工程文件名为“工程1.vbp”。

如果想保存磁盘上已有但修改后的工程文件,直接单击工具栏上的“保存工程”按钮或“文件”菜单中的“保存工程”命令。此时,系统不会弹出“工程另存为”对话框。所以,若要将已经存在的工程保存到其他文件夹中或用其他文件名保存,则需要选择“工程另存为”命令。

(2) 工程的编译

当完成工程的全部文件之后,可以将此工程转换成可执行文件(.exe)。

在 VB 中对程序(工程)的编译非常简单,在“文件”菜单中选择“生成工程1.exe”命令。在打开的“生成工程”对话框中选择程序所保存的文件夹和文件名,然后单击“确定”按钮即可生成 Windows 中的应用程序,以后此工程即可脱离 VB 环境直接在 Windows 下运行(不要求在该系统中安装 VB 应用软件)。

当完成一个应用程序的设计后,如果还有其他的设计任务,则可选择“文件”菜单的“新建工

程”命令。如果要退出 VB,可单击图 1-3 中 VB 窗口内标题栏右上角的“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,并询问用户是否保存文件或直接退出。

1.2 VB 集成开发环境

1.2.1 集成开发环境概况

如图 1-3 所示的 VB6.0 的集成环境与 Microsoft Office 家族中的软件类似,除了 Microsoft 应用软件常规的标题栏、菜单栏、工具栏外,还可以根据需要添加或删除多种独立窗口,如对象窗口、属性窗口、代码窗口、立即窗口等。在窗口的任何位置上单击鼠标右键可显示快捷菜单等。

VB 提供的集成开发环境 (IDE),将应用程序的开发、测试、查错等功能集于一体,大大降低其操作的复杂性,提高了开发应用程序的效率。

1.2.2 标题栏

标题栏是 VB6.0 集成开发环境窗口顶部的水平条。标题栏中显示了当前工程的名称以及工作模式。

VB 提供了设计模式、运行模式和中断模式三种工作模式。各模式特点如下:

1. 设计模式 设计应用程序的模式。在该模式下,窗体标题栏显示“设计”字样,用户既可以进行界面设计也可以进行代码编写。
2. 运行模式 运行应用程序的模式。在该模式下,窗体标题栏显示“运行”字样,用户既不能进行界面设计也不能进行代码编写。
3. 中断模式 暂时中断应用程序的运行。在该模式下,窗体标题栏文本显示“break”字样,用户只能编辑代码,而不能编辑界面。

1.2.3 菜单栏

VB6.0 菜单栏中包括 13 个下拉式菜单。菜单栏提供了开发应用程序所需的各种选项,包括“文件”、“编辑”、“视图”、“工程”、“格式”、“调试”、“运行”、“工具”、“外接程序”、“窗口”、“帮助”等菜单项,包含了 VB 编程的常用命令。

1.“文件”菜单项

提供了用于文件操作的全部选项,如新建工程、打开工程、保存工程、显示最近的工程以及生成可执行文件等功能。

2.“编辑”菜单项

提供设计模式下编辑界面及代码的功能。

3.“视图”菜单项

包括显示或隐藏集成开发环境中各种窗口,如代码窗口、对象窗口、工程资源管理器窗口、属性窗口、工具箱、工具栏等的操作选项。

4.“工程”菜单项

包括用于多窗体程序设计的添加窗体选项、向工具箱添加控件的部件选项,用于设置某些工程属性的选项等。

5.“格式”菜单项

提供对界面设计的辅助控制,如控件的对齐、间距等格式化功能。

6.“调试”菜单项

提供程序调试、查错等功能。

7.“运行”菜单项

提供程序启动、中断和停止等功能。

8.“查询”菜单项

提供与数据库有关的查询功能。

9.“图表”菜单项

提供与图表有关的操作功能。

10.“工具”菜单项

提供集成开发环境下工具的扩展功能。

11.“外接程序”菜单项

提供 VB 外接程序及外接程序管理器等功能。

12.“窗口”菜单项

提供 VB 子窗口在主窗口中的排列方式等功能。

13.“帮助”菜单项

提供 VB 的联机帮助功能。

1.2.4 工具栏

菜单栏的下面就是工具栏,工具栏提供快速访问常用命令的方法。当鼠标指针指向工具栏各按钮时,系统会自动显示出该按钮的作用的中文提示,单击工具栏上的按钮,即可执行相应的操作。VB 常用的工具栏有标准工具栏、编辑工具栏、调试工具栏等。要显示或隐藏这些工具栏,可以选择“视图”→“工具栏”命令选取所需工具栏或右击标准工具栏,在弹出的快捷菜单中选取。标准工具栏中主要按钮的图标及其功能见表 1-1。

在标准工具栏的最右边是数字显示区,分别用来显示窗体的当前位置和大小,其单位是 twip(1 厘米=567 twip,1 英寸=1440 twip)。数字显示区包含两个部分,左数字区显示的是对象的左上角的坐标位置(窗体工作区的左上角为坐标原点,即(0,0)位置),右数字区显示的是对象的高度(向下递增)和宽度(向右递增),即对象的大小。