

高等学校计算机教材

# Visual Basic 教程

陈明 编

人民邮电出版社  
POSTS & TELECOMMUNICATIONS PRESS

高等学校计算机教材

# Visual Basic 教程

陈 明 编

人民邮电出版社

## 图书在版编目 (CIP) 数据

Visual Basic 教程 / 陈明编 . —北京: 人民邮电出版社, 2002.8

高等学校计算机教材

ISBN 7-115-09867-0

I. V... II. 陈... 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 (2002) 第 061491 号

## 内 容 简 介

本书主要介绍 Visual Basic 6.0 程序设计语言及应用的有关内容。主要包括: Visual Basic 概述、Visual Basic 6.0 集成环境、数据类型、表达式与运算、程序流程、Visual Basic 应用程序结构、常用控件、文件处理、多窗体及 MDI 窗体设计、界面设计等。本教材注重应用方法和可视化程序设计的介绍, 并提供大量编程示例, 可帮助读者快捷地开发出 Visual Basic 应用程序。全书内容系统, 循序渐进, 逻辑性强。

本书可作为高等院校非计算机专业的基础课教材, 也可作为从事软件开发和应用的工程技术人员的参考书。

高等学校计算机教材

### Visual Basic 教程

- 
- ◆ 编 陈 明  
责任编辑 滑 玉
  
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
读者热线 010-67180876  
北京汉魂图文设计有限公司制作  
北京隆昌伟业印刷有限公司印刷  
新华书店总店北京发行所经销
  
  - ◆ 开本: 787×1092 1/16  
印张: 20.25  
字数: 488 千字 2002 年 8 月第 1 版  
印数: 1-6 000 册 2002 年 8 月北京第 1 次印刷

---

ISBN 7-115-09867-0/TP · 2612

定价: 27.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

## 编者的话

Visual Basic 6.0 具有强大的可视化程序设计和面向对象程序设计的功能，近年来已得到了广泛地应用。Visual Basic 是高质量的先进软件开发工具，其主要特点是：

1. 简单易学
2. 简化了复杂的界面设计
3. 强大的多媒体功能
4. 网络功能支持了 Internet 编程
5. 具有所见即所得的功能
6. 支持用户开发的 ActiveX 控件

全书分为 10 章，介绍了 Visual Basic 6.0 的主要功能和使用方法。主要内容包括：

1. Visual Basic 概述
2. Visual Basic 6.0 集成开发环境
3. 数据类型
4. 表达式与运算符
5. 程序流程
6. Visual Basic 应用程序结构与调试
7. 常用控件
8. 文件处理
9. 多窗体及 MDI 窗体设计
10. 界面设计

本书在结构上为模块化，叙述内容深入浅出、并重于应用，尤其注重于界面设计和程序代码的设计。书中各章都附有内容小结和习题，可以帮助读者很好地巩固所学习的知识。

由于作者水平有限，书中不足之处在所难免，敬请读者批评指正。

陈明

2002 年 6 月

# 目 录

<b>第 1 章 Visual Basic 概述</b> .....	1
1.1 Visual Basic 的特点 .....	1
1.2 Visual Basic 基本概念 .....	2
1.3 Visual Basic 简单示例 .....	6
1.3.1 设计步骤 .....	6
1.3.2 运行程序 .....	8
1.4 小结 .....	9
习题 1 .....	9
<b>第 2 章 Visual Basic 6.0 集成开发环境</b> .....	11
2.1 启动和退出 Visual Basic 6.0 中文版 .....	11
2.2 VB 的集成开发环境 .....	13
2.2.1 标题栏 .....	13
2.2.2 菜单栏 .....	13
2.2.3 工具栏 .....	14
2.2.4 工具箱 .....	14
2.2.5 代码编辑器窗口 .....	16
2.2.6 项目窗口 .....	17
2.2.7 属性窗口 .....	18
2.2.8 窗体 .....	19
2.2.9 窗体布局窗口 .....	19
2.3 创建 Visual Basic 项目 .....	19
2.4 菜单的详细介绍 .....	25
2.5 应用程序开发实例 .....	34
2.5.1 创建应用程序界面 .....	34
2.5.2 设置属性 .....	35
2.5.3 编写代码 .....	36
2.5.4 保存应用程序 .....	38
2.5.5 调试、运行应用程序 .....	38
2.5.6 生成可执行文件 .....	38
2.6 小结 .....	39
习题 2 .....	39
<b>第 3 章 变量、常量、数组与数据类型</b> .....	40
3.1 变量与常量 .....	40
3.1.1 变量 .....	40

3.1.2 常量 .....	47
3.2 数值数据类型 .....	48
3.2.1 整型 .....	48
3.2.2 长整型 .....	49
3.2.3 单精度型 .....	49
3.2.4 双精度型 .....	49
3.2.5 货币型 .....	49
3.3 字符串 .....	50
3.3.1 可变长字符串 .....	50
3.3.2 定长字符串 .....	50
3.3.3 使用字符串应注意的问题 .....	50
3.4 Variant 数据类型 .....	52
3.5 Date 数据类型 .....	52
3.6 Decimal 数据类型 .....	53
3.7 数据类型转换 .....	53
3.8 数组 .....	53
3.8.1 固定大小数组 .....	54
3.8.2 动态数组 .....	55
3.9 记录 .....	57
3.10 小结 .....	60
习题 3 .....	61
<b>第 4 章 表达式和运算符 .....</b>	<b>62</b>
4.1 算术运算符 .....	62
4.2 字符串运算符 .....	65
4.3 关系运算符 .....	66
4.4 逻辑运算符 .....	68
4.5 特殊运算符 .....	69
4.5.1 Is 运算符 .....	69
4.5.2 Like 运算符 .....	70
4.6 运算符的优先级 .....	71
4.7 在运算中数据类型的转换 .....	72
4.8 小结 .....	73
习题 4 .....	73
<b>第 5 章 程序的流程 .....</b>	<b>74</b>
5.1 赋值语句 .....	74
5.2 注释语句 .....	74
5.3 程序续行 .....	75
5.4 一行中的多条语句 .....	75
5.5 程序流程 .....	75

5.5.1 顺序结构 .....	76
5.5.2 Goto 语句 .....	76
5.5.3 条件判断 .....	77
5.5.4 循环 .....	81
5.6 小结 .....	94
习题 5 .....	95
<b>第 6 章 Visual Basic 应用程序的结构与调试 .....</b>	<b>96</b>
6.1 模块 .....	96
6.1.1 VCR 应用程序的结构 .....	96
6.1.2 各种模块类型 .....	98
6.1.3 使用“代码编辑器” .....	100
6.2 过程 .....	101
6.2.1 过程概述 .....	101
6.2.2 Sub 过程 .....	102
6.2.3 Function(函数)过程 .....	105
6.3 过程的使用 .....	109
6.3.1 过程的创建与使用 .....	109
6.3.2 向过程传递参数 .....	112
6.4 常见标准函数 .....	115
6.4.1 数学函数 .....	115
6.4.2 日期函数 .....	115
6.4.3 字符串操作函数 .....	115
6.4.4 数据转换函数 .....	115
6.4.5 用户交互式函数 .....	117
6.4.6 其他常用函数 .....	123
6.5 程序的调试 .....	126
6.5.1 错误示例 .....	126
6.5.2 Visual Basic 6.0 中的错误类型 .....	127
6.5.3 Debug 窗口 .....	132
6.5.4 断点调试 .....	133
6.6 综合示例 .....	134
6.7 小结 .....	136
习题 6 .....	137
<b>第 7 章 常用控件 .....</b>	<b>138</b>
7.1 标签(Label) .....	138
7.1.1 常用属性 .....	138
7.1.2 常用事件 .....	139
7.1.3 例题 .....	139
7.1.4 Label 控件的其他用法 .....	145

7.2	文本框(Text Box)	149
7.2.1	MultiLine 属性	150
7.2.2	SelStart 属性	153
7.3	命令按钮(Command Button)	156
7.3.1	常用属性	156
7.3.2	示例	159
7.4	图片框(Picture Box)	167
7.4.1	图片框的特殊属性	167
7.4.2	Paint 事件	175
7.4.3	PaintPicture 方法	177
7.5	框架(Frame)	180
7.6	单选按钮(Option Button)	182
7.7	复选框(CheckBox)	188
7.8	列表框(ListBox)	189
7.8.1	列表框的主要使用方法	190
7.8.2	列表框的常用属性	191
7.9	组合框(ComboBox)	196
7.10	滚动条(ScrollBar)	200
7.11	计时器(Timer)	204
7.12	小结	209
	习题 7	209
<b>第 8 章</b>	<b>文件处理</b>	<b>211</b>
8.1	文件控件	211
8.1.1	DriveListBox 控件	211
8.1.2	DirListBox 控件	213
8.1.3	FileListBox 控件	218
8.1.4	CommonDialog 控件	220
8.2	利用文件控件来处理文件	222
8.2.1	利用文件控件的第一个示例	223
8.2.2	利用文件控件的第二个示例	225
8.3	Visual Basic 6.0 的文件处理语句与函数	230
8.3.1	Kill 语句	230
8.3.2	FileCopy 语句	231
8.3.3	Shell 函数	234
8.3.4	Rmdir 语句	236
8.3.5	Name 语句	238
8.4	文件系统	242
8.4.1	顺序文件	242
8.4.2	随机文件	248



8.4.3 二进制文件 .....	249
8.5 文件系统对象(FileSystemObject, FSO)编程 .....	250
8.5.1 创建 FileSystemObject 对象 .....	251
8.5.2 访问已有的驱动器、文件和文件夹 .....	251
8.5.3 使用驱动器 .....	252
8.5.4 使用文件夹 .....	252
8.5.5 使用文件 .....	253
8.6 小结 .....	256
习题 8 .....	256
<b>第 9 章 多窗体及 MDI 窗体的设计</b> .....	<b>258</b>
9.1 利用 ActiveX 控件进行窗体的界面设计 .....	258
9.2 多窗体程序设计 .....	268
9.2.1 多窗体程序设计常用的方法 .....	268
9.2.2 多窗体程序设计示例 .....	269
9.3 多文档界面(MDI)的程序设计 .....	279
9.3.1 MDI 应用程序设计示例 .....	279
9.3.2 MDI 应用程序中的菜单设计 .....	284
9.4 小结 .....	288
习题 9 .....	288
<b>第 10 章 界面设计</b> .....	<b>289</b>
10.1 认识菜单 .....	289
10.2 菜单编辑器 .....	290
10.2.1 启动菜单编辑器 .....	290
10.2.2 编辑菜单 .....	292
10.2.3 菜单的代码设计 .....	292
10.3 菜单的高级应用 .....	295
10.3.1 菜单的有效属性 .....	295
10.3.2 菜单的可见属性 .....	297
10.3.3 创建菜单控件数组 .....	298
10.3.4 菜单的复选(Checked)属性 .....	300
10.3.5 动态菜单 .....	301
10.4 弹出菜单和分隔条的设计 .....	303
10.5 工具栏的设计 .....	306
10.5.1 利用应用程序向导创建工具栏 .....	306
10.5.2 使用 ToolBar 控件创建工具栏 .....	307
10.6 小结 .....	310
习题 10 .....	311
参考文献 .....	312

# 第 1 章 Visual Basic 概述

本章主要介绍有关 Visual Basic(以下简称 VB)程序语言的基本特点、基本概念和一个 VB 程序语言的简单应用示例。在 Visual Basic 特点中,介绍了可视化设计、事件驱动编程、面向对象的程序设计等。在 VB 基本概念中,介绍了窗体、控件、属性、事件及方法等。最后,通过一个简单的示例说明 VB 的基本使用方法。

## 1.1 Visual Basic 的特点

VB 是一个快速可视化程序开发工具,是极有特色和功能强大的软件。VB 使用了面向对象的程序设计方法,大大地增强了程序的可重用性,进而简化了程序设计。VB 提供了众多的 ActiveX 控件,提高了软件开发人员的编程效率,VB 的主要特点如下。

### 1. 可视化设计

VB 具有“所见即所得”的功能。在程序设计时,头脑中所想象的应用程序界面,完全可以通过键盘与鼠标完成,而不用编制大量的代码。如果需要修改,也可利用键盘或鼠标完成,而底层的一些程序代码可由 VB 自动生成或修改。VB 为用户提供了大量的控件,这些控件对于熟悉 Windows 应用程序的用户而言是一点也不陌生,如“标签”、“组合框”、“命令按钮”、“文本框”、“单选框”等等,用户只需用鼠标或键盘把这些控件拖到所需的位置,并设置它们的大小、形状属性等,即可得到所需的应用程序界面。

### 2. 事件驱动编程

在 VB 中把窗体及控件称为对象。这些对象组成了与用户交互的图形界面(也称为用户接口或用户界面)。在设计应用程序时,必须考虑到用户如何与程序进行交互。用户通过鼠标和键盘与应用程序进行交互是最直接、最方便的,这时那些对象必须对鼠标、键盘操作所引起的事件作出响应。响应是指执行一段应用程序,它不沿预定的路径执行,而是在响应不同事件时执行不同的代码。因此,VB 编程重点是编写事件驱动过程的代码。

### 3. 面向对象的程序设计

VB 是面向对象的程序设计语言,它将代码和数据结合在一个个对象中,用户只需了解每个对象能完成何种任务,而不需知道对象是如何工作的。软件开发人员主要编写事件驱动过程的代码,使对象对用户的操作作出响应。但因 VB 没有多态等面向对象的特性,所以也有人称 VB 是一种准面向对象的程序设计语言。

### 4. VB6.0 是一种 32 位的开发工具

VB5.0 以前的开发工具很多是在 DOS 和 Windows 3x 下使用的,因而它们基本上是 16 位的开发工具,VB6.0 是运行在 Windows 9x 或 WindowsNT 操作系统下的,因此它是典型的 32 位开发工具。

### 5. 易学易用、开发周期短

VB 简单易学，界面友好，容易使用。VB 提供的大量功能强大的控件，能帮助开发人员在较短时间内用少量语句编写出应用程序，缩短了开发周期。

## 1.2 Visual Basic 基本概念

在 VB 中，对象是一个非常重要概念。世界上存在的任何事物都可以称之为对象。从天体中的太阳、月亮、地球，到生活中常见的物品，如报纸、衣服、裤子和鞋袜，都可以看做对象。对象是具体存在的，每个对象都是唯一的，它们有自己的个性。

类则是具有相同性质的对象的集合，对象可理解为类的具体化。如人可以称之为类，而每一个人，却只能看做是对象。当然，生活在北京的人叫北京人，他们作为一个具体存在的群体，也可以看做对象。

计算机内的零配件可以看做是对象，用户可以根据需要进行选用，把不同类型的主板、显示器、显卡、声卡、硬盘和软驱等组装在一起形成各不相同的计算机，可以满足不同用户的需要。同样，VB 中的对象也都是现成的，使用它们可以编写出各种各样的满足不同用户需要的应用程序。窗体、控件都被看作对象。任何一个对象都可由事件、属性和方法描述。

### 1. 窗体

窗体(Form)是一个对象，在大部分设计过程当中都要用到，它起着容器的作用，各种各样的控件都可以放在窗体中。在 VB 启动后，弹出的窗体如图 1-1 所示。

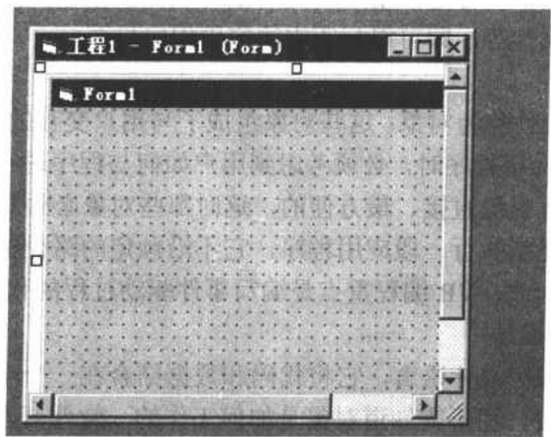


图 1-1 启动 VB 后的窗体

### 2. 控件

VB 具有丰富的控件。启动 VB 之后出现一个工具箱，如图 1-2 所示。工具箱中列出了 VB 中常用控件，这些控件是用来与用户交互、完成输入输出任务的部件，并按应用程序的要求，实现窗体和各控件之间相互配合，完成复杂任务。

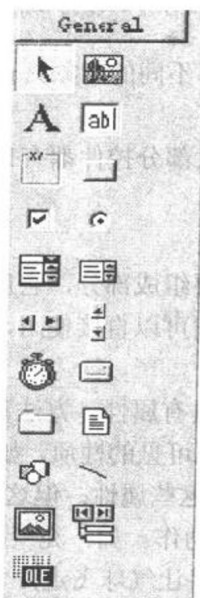



图 1-2 工具箱

工具箱中常用控件有：

- (1) 图标代表 PictureBox 控件，称为“图片”控件。
- (2) 图标代表 Label 控件，称为“标签”控件。
- (3) 图标代表 TextBox 控件，称为“文本框”控件。
- (4) 图标代表 Frame 控件，称为“框架”控件。
- (5) 图标代表 CommandButton 控件，称为“命令按钮”控件。
- (6) 图标代表 CheckBox 控件，称为“复选框”控件。
- (7) 图标代表 OptionButton 控件，称为“单选按钮”控件。
- (8) 图标代表 ListBox 控件，称为“列表框”控件。
- (9) 图标代表 ComboBox 控件，称为“组合框”控件。
- (10) 图标代表 HScrollBar 控件，称为“水平滚动条”控件。
- (11) 图标代表 VScrollBar 控件，称为“垂直滚动条”控件。
- (12) 图标代表 Timer 控件，称为“时钟”或“计时器”控件。
- (13) 图标代表 DriverListBox 控件，称为“驱动器列表框”控件。
- (14) 图标代表 DirListBox 控件，称为“目录列表框”控件。
- (15) 图标代表 FileListBox 控件，称为“文件列表框”控件。
- (16) 图标代表 Shape 控件。这个控件包含圆、正方形、矩形等形状的图形。
- (17) 图标代表 Line 控件。它是一个线条控件。
- (18) 图标代表 Image 控件。它是一个图形控件。
- (19) 图标代表 Data 控件。它是数据控件中的一种。
- (20) 图标代表 Ole 控件。

另外，图标用于对窗体设计时放弃控件选择，从而返回窗体的非设计状态，它不是控件。

### 3. 属性

对象的属性是指对象本身的特性，不同的对象具有不同的属性。

### 4. 事件

事件是指在对象上发生的事情，大部分控件都有自己的事件。当触发对象的事件时，事件就得到响应，完成特定的工作。

### 5. 方法

在 VB 中，方法是对象的一个重要组成部分，它反映了对象本身的功能，这些功能由特定的内部函数和过程完成。对象的方法可以直接使用，用户不需了解方法是如何实现的。

### 6. 面向对象编程

日常生活中的对象，如气球同样具有属性、方法和事件。气球的属性包括可以看到的一些性质，如它的直径和颜色；或一些不可见的性质，如用于描述气球状态的属性(充气的或未充气的)。通过定义，所有气球都具有这些属性；但这些属性的值也会因气球的不同而不同。

气球还具有本身所固有的方法和动作。如：充气方法(用氦气充满气球的动作)，放气方法(排出气球中的气体)和上升方法(放手让气球飞走)。所有的气球都具备这些方法。

气球还有预定义的、对某些外部事件的响应。例如，气球对刺破它的事件响应是放气，对放手事件的响应是升空。

对气球编程的 VB 代码如下所示，其作用是设置气球的属性：

```
Balloon.Color = Red
Balloon.Diameter = 10
Balloon.Inflated = True
```

首先是对象 **Balloon**，接着是属性 **Color**，然后是赋值 **Red**。重复这条语句，将属性 **Color** 换成另外一个不同的值，就可以改变气球的颜色。属性也可以在程序设计阶段在“属性”窗口中设置。

调用气球的方法是这样的：

```
Balloon.Inflate
Balloon.Deflate
Balloon.Rise 5
```

它的语法与属性的语法相似，对象(一个名词)后面紧跟着方法(一个动词)。在第三个例子中有一个附加项，称为一个参数，表示气球上升的高度。一些方法可以有一个或多个参数，它们可对执行的动作做进一步的描述。

气球对一个事件的响应如下：

```
Sub Balloon_Puncture()
    Balloon.Deflate
    Balloon.MakeNoise "Bang"
    Balloon.Inflated = False
    Balloon.Diameter = 1
End Sub
```

本例中，代码描述了当刺穿事件发生时气球的行为：调用 **Deflate** 方法，然后以“Bang”(发出的声响)为参数调用 **MakeNoise** 方法。因为气球已不再是充气状态，所以把 **Inflated** 属性置为 **False**，并给 **Diameter** 属性设置一个新值。

实际上对气球这个对象是不能编程的，而在 VB 中，可以对其对象(窗体或控件)进行编

程。可由程序员自己控制应更改哪些属性、调用哪些方法、对哪些事件做出响应，从而得到希望的外观和行为。

VB 的窗体和控件是具有自己的属性、方法和事件的对象。可以把属性看作一个对象的性质，把方法看作对象的动作，把事件看作对象的响应。

窗体是一种对象，由属性定义其外观，由方法定义其行为，由事件定义其与用户的交互。通过设置窗体属性并编写响应事件的代码，就能定义出满足应用程序需要的对象。

控件是包括在窗体对象内的对象。每种类型的控件都有自己的属性、方法和事件，以适用于特定的目的。一些控件最适合在应用程序中输入或显示文本，另一些控件能够访问其他的应用程序和处理数据。

### 7. 事件驱动

VB 是基于 Windows 开发环境下的开发语言，与 Windows 环境保持一定的相似性。所以需要了解在 Windows 环境下编程的特点以及与在其他环境下编程的一些根本性的差别。

#### (1) Windows 的工作机制

Windows 的工作机制简单地说是三个概念：窗口、事件和消息。

窗口可以看作带有边界的矩形区域。如，Windows 95 的“资源管理器”窗口、文字处理程序中的文档窗口或者弹出有提示信息的对话框。除了这些最普通的窗口外，还有许多其他类型的窗口，如命令按钮是一个窗口，图标、文本框、选项按钮和菜单栏等也都是窗口。

Microsoft Windows 操作系统通过给每个窗口指定一个唯一的标识(窗口句柄或 hWnd)来管理所有的窗口。操作系统连续地监视每一个窗口的活动或事件的信号。事件可以通过单击鼠标或按下按键的操作而产生，也可以通过程序的控制而产生，甚至可以由另一个窗口的操作而产生。

可以想象，处理各种窗口、事件和消息的所有可能的组合将有很大的工作量。幸运的是，VB 摆脱了所有的底层消息处理。

#### (2) 事件驱动模型

在传统的或“过程化”的应用程序中，应用程序自身控制了代码的执行过程和代码的执行顺序。执行时从第一行代码开始并按应用程序中预定的路径执行，必要时还要调用过程。

在事件驱动的应用程序中，代码不是按照预定的路径执行，而是在响应不同的事件时执行不同的代码片段。事件可以由用户操作触发，也可以由来自操作系统或其他应用程序的消息触发，甚至由应用程序本身的消息触发。这些事件的顺序决定了代码执行的顺序，因此应用程序每次运行时所经过的代码的路径都是不同的。因为事件的顺序是无法预测的，所以在代码中必须对执行时的“各种状态”做一定的假设。当做出某些假设时，应该组织好应用程序的结构，以确保该假设始终有效。在执行中，代码也可以触发事件。

#### (3) 交互式开发

传统的应用程序开发过程可以分为三个步骤：编码、编译和测试代码。但是 VB 与传统的语言不同，它使用交互式方法开发应用程序，使三个步骤之间不再有明显的界限。

在大多数语言里，如果编写代码时发生了错误，则在开始编译应用程序时该错误就会被编译器捕获。此时必须查找并改正该错误，然后再次进行编译，对每一个发现的错误都要重复这样的过程。VB 在程序员输入代码时便进行解释，即时捕获并突出显示大多数语法或拼写错误。除即时捕获错误以外，VB 也在输入代码时部分地编译该代码。当准备运行和测试

应用程序时，只需极短时间即可完成编译。如果编译器发现了错误，则将错误突出显示于代码中。这时可以更正错误并继续编译，而不需从头开始编译。由于 VB 的交互特性，代码运行的效果可以在开发时进行测试，而不必等到编译完成以后。

#### (4) 事件驱动应用程序的工作方式

事件是窗体或控件识别的动作。在响应事件时，事件驱动应用程序执行 VB 代码。VB 的每一个窗体和控件都有一个预定义的事件集。如果其中有一个事件发生，而且在关联的事件过程中存在代码，则 VB 调用该代码。

尽管 VB 中的对象自动识别预定义的事件集，但要判定它们是否响应具体事件以及如何响应具体事件则是编程的责任了。事件过程与每个事件对应，为了让控件响应事件，需要把代码写入这个事件的事件过程之中。

对象所识别的事件类型多种多样，多数类型为大多数控件所共有。例如，大多数对象都能识别 Click 事件，如果单击窗体，则执行窗体的单击事件过程中的代码；如果单击命令按钮，则执行命令按钮的 Click 事件过程中的代码。但其中的实际代码几乎完全不一样。

事件可由用户引发(例如键盘操作)、可由系统引发(例如定时器事件)，也可由代码间接引发(例如当代码装载窗体时的 Load 事件)。如果在相应的事件过程中存在代码，就执行代码。应用程序等待下一次事件。许多事件会伴随其他事件发生。例如，在 DbClick 事件发生时，MouseDown、MouseUp 和 Click 事件也会发生。

## 1.3 Visual Basic 简单示例

利用 VB 设计应用程序不是一件困难的事情，下面通过一个简单的 VB 应用程序设计示例，说明 VB 编程的一般过程，使读者对 VB 程序有一个初步的认识。

### 1.3.1 设计步骤

#### 1. 启动 VB

单击“开始”按钮，在“程序”菜单中找到 VB 程序项并单击，即可启动 VB。

然后从弹出的“新建工程”对话框中选择“新建”选项卡中的“标准 EXE”项，单击“打开”按钮，进入 VB6.0 集成开发环境。

此时将产生一个工程，这个工程只有一个窗体(Form1)。

#### 2. 向窗体添加“标签”控件

在“工具箱”中双击“标签”控件，将“标签”控件添加到窗体上，此时可以在窗体上拖动“标签”控件，将“标签”控件调整到适当位置，也可调整它的大小，进而得到如图 1-3 所示的结果。

#### 3. 向窗体添加“命令按钮”控件

与步骤 2 相似，向窗体添加一个“命令按钮”控件，如图 1-4 所示。

#### 4. 修改“标签”控件和“命令按钮”控件的属性

在“属性窗口”找到“命令按钮”控件的“Caption”属性。该属性用于设置对象的标题，如图 1-5 所示，将它的值改为“走进 VISUAL BASIC”。接着在“属性窗口”找到“标签”控

件的“Caption”属性，将其值改为“VB 是一门面向对象语言”，如图 1-6 所示。

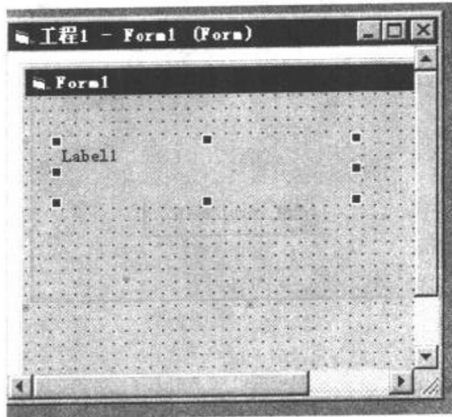


图 1-3 添加“标签”控件

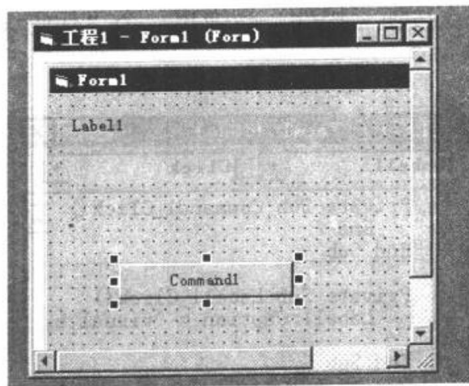


图 1-4 添加“命令按钮”的窗体

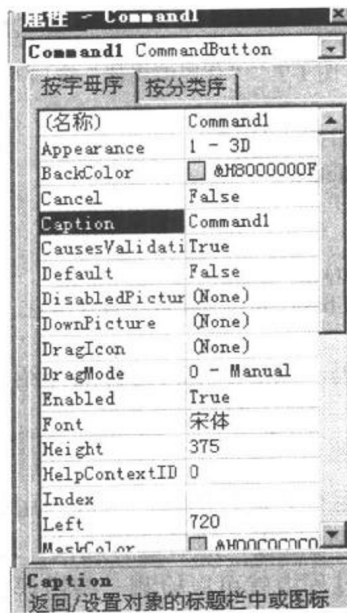


图 1-5 属性窗口





图 1-6 窗体设计结果

### 5. 添加事件响应代码

双击“窗体”上的“命令按钮”控件，会弹出一个“代码编辑器”窗口，如图 1-7 所示。

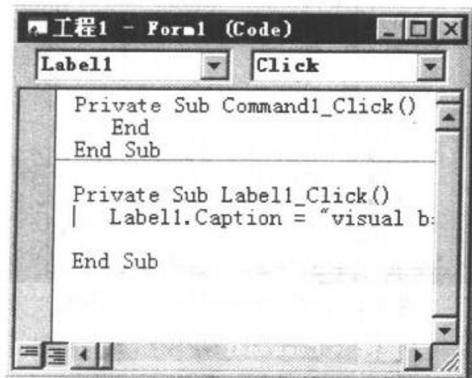


图 1-7 “代码编辑器”窗口

在“代码编辑器”窗口中添加如下代码：

```
Private Sub Command1_Click()
    End
End Sub

Private Sub Label1_Click()
    Label1.Caption="visual basic 是可视化程序设计语言。"
End Sub
```

读者可暂不关心如何编写代码，以后章节将会详细介绍。

### 1.3.2 运行程序

应用程序设计好以后，可以运行并进行测试。一般当用编辑器编辑好一个程序后，都要进行修改、编译、再修改、再编译的过程，然后进行连接，最后生成需要的可执行文件。VB 是一种交互式语言，在编写代码过程中就可以检测一些错误。此例执行过程简单，编写代码未发现错误。单击“工具栏”上的“▶”启动按钮，如图 1-8 所示。