



教育部职业教育与成人教育司推荐教材

桌面编程之 Visual Basic

主编 徐国强 姜秋明

 中国财政经济出版社
China Financial & Economic Publishing House

教育部职业教育与成人教育司推荐教材

jiaoyubuzhiyejiaoyuyuchengrenjiaoyusituijianjiaocai

桌面编程之 Visual Basic

主编 徐国强 姜秋明
审稿 郭振华 鲁春发

中国财政经济出版社

图书在版编目 (CIP) 数据

桌面编程之 Visual Basic/徐国强、姜秋明主编. —北京: 中国财政经济出版社, 2006.10
教育部职业教育与成人教育司推荐教材

ISBN 7-5005-8456-3

I. 桌… II. 徐… III. BASIC 语言-程序设计-高等学校: 技术学校-教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 085006 号

中国财政经济出版社 出版

URL: <http://www.cfeph.cn>

E-mail: cfeph@cfeph.cn

(版权所有 翻印必究)

社址: 北京市海淀区阜成路甲 28 号 邮政编码: 100036

发行电话: 88190616 传真: 88190655

慧美印刷厂印刷 各地新华书店经销

787×1092 毫米 16 开 12.25 印张 289 000 字

2006 年 11 月第 1 版 2006 年 11 月北京第 1 次印刷

定价: 16.00 元

ISBN 7-5005-8456-3/TP·0106

(图书出现印装问题, 本社负责调换)

本教材的正版图书封底上贴有“中国财政经济出版社 教育分社”防伪标识。根据标识上提供的查询网站、查询电话和查询短信, 输入揭开防伪标识后显示的产品数字编号, 即可查询本书是否为正版图书。版权所有, 翻印必究, 欢迎读者举报。举报电话: 010—88190654。

出版说明

为了进一步贯彻落实《国务院关于大力推进职业教育改革与发展的决定》和全国职业教育工作会议的精神,适应中等职业教育发展的趋势,满足各类职业技术学校培养技能型紧缺人才的实际需要,我们组织编写了中等职业教育计算机应用与软件技术专业教学用书。从2005年秋季开学起,这些教材将陆续提供给各类职业技术学校使用。

该系列教材是根据教育部提出的“以综合素质培养为基础,以能力培养为主线”为指导思想,以教育部新近颁布的计算机应用与软件技术专业教学指导方案为依据,结合中等职业教育的教学培养目标而编写的,经教育部职业教育与成人教育司批准立项,并由专家审定,作为教育部职业教育与成人教育司推荐教材出版。新教材全面贯彻素质教育思想,从社会发展对高技术应用性人才的需求出发,在内容的构建上结合专业岗位(群)对职业能力的需要来确定教材的知识点、技能点和素质要求点,并注重新知识、新技术、新工艺、新方法的应用,注重对学生的创新精神和实践能力的培养。新教材在理论体系、组织结构和阐述方法等方面均作了一些新的尝试,以适应中等职业教育教学改革,满足各类中等职业技术学校的教学需要。在此,我们真诚的希望各类职业技术学校在使用教材的过程中,能够总结经验,及时提出修改意见和建议,使之不断完善和提高。

2006年6月

前 言

Visual Basic 是运行于 Windows 平台上的可视化集成开发环境。它是 Microsoft 公司开发的 Microsoft Visual Studio 系列的一部分。Visual Basic 集程序的界面设计、代码编辑、编译、连接和调试等功能于一体，为编程人员提供了一个方便而又完整的开发界面和许多有效的辅助开发工具，目前，从许多初学者到专家都在使用它，也是中等专业学校计算机桌面编程的首选。本教程采用 Visual Basic 6.0。

根据教育部对中等专业学校计算机和软件开发设计教学的要求，在内容取舍和教学层次上，我们本着“实用、够用、好用”和“先会后懂、先会后精”的原则，精心安排了编程的教学内容。从一个可用的简单界面编程开始，到编写程序代码，再到设计完整的应用程序，手把手教学生如何使用 Visual Basic 开发软件。全篇内容简单明了，通俗易懂，易于学生学习和实践。教学内容包括用户界面设计、变量与表达式、控制语句、编写过程、多媒体程序设计、窗体设计、网络编程、数据库编程、VB.net 知识。通过这些知识的学习，可以迅速提高编程能力，并且把它应用到自己的工作和学习中。

本书由徐国强、姜秋明主编。参与编写的人员为：徐国强（第 1、2、5 章）；姜秋明（第 7、8、9 章）；万正平（第 3 章）；郭亚庆（第 4 章）；奚建胜（第 6、10 章）。由于时间紧，编者水平有限，书中难免有不当和错误之处，恳请广大使用者批评指正。

编 者

2005 年 4 月

目 录

第 1 章 编程起步

1

- 第 1 节 Visual Basic 轻松入门
- 第 2 节 对象、属性、事件和方法
- 第 3 节 轻松编程——画出一个程序
- 第 4 节 图形控件和初级菜单

1

7

10

15

第 2 章 变量与表达式

22

- 第 1 节 基本数据类型
- 第 2 节 常量和变量
- 第 3 节 常用内部函数
- 第 4 节 Visual Basic 中的运算

22

24

28

31

第 3 章 Visual Basic 基本结构与流程控制

34

- 第 1 节 结构化的程序设计
- 第 2 节 顺序结构语句
- 第 3 节 分支结构语句
- 第 4 节 循环结构语句

34

36

38

43

第 4 章 Visual Basic 中的过程

51

- 第 1 节 Visual Basic 中的过程概述
- 第 2 节 如何编写过程
- 第 3 节 调用过程中的参数传递
- 第 4 节 变量的作用域

51

55

58

59

第5章 窗体界面设计 61

第1节	窗体对象	61
第2节	文本编辑器界面设计	64
第3节	多文档界面 (MDI) 应用程序	69
第4节	窗体的高级设计	73
第5节	使用工具栏控件	75
第6节	状态栏控件和对话框的使用	80

第6章 使用数组与字符串 86

第1节	数组声明与运用	86
第2节	二维数组的定义与运用	91
第3节	记录与数组	93
第4节	数组排序与字符串操作	94
第5节	常用的字符串操作	98
第6节	在文本框中进行字符串操作	99

第7章 多媒体程序设计 109

第1节	ActiveX 控件简介	109
第2节	设计多媒体播放器的界面	110
第3节	编写代码	113

第8章 网络编程 123

第1节	网络编程概述	123
第2节	TCP (UDP) /IP 编程技术	124
第3节	创建 Internet Explorer 中运行的应用程序	129
第4节	创建 FTP 浏览器	131

第9章 数据库编程——档案管理信息系统 134

第1节	概述	134
第2节	需求分析	135
第3节	UML 系统建模	137
第4节	系统配置	139
第5节	数据库分析	141

第 6 节	界面设计	145
第 7 节	代码分析/实现	156

第 10 章 Visual Basic.Net 175

第 1 节	Visual Basic6.0 与 Visual Basic.Net 之间的不同	175
第 2 节	Visual Basic.Net	176
第 3 节	升级到 Visual Basic.Net	184

第 1 章

编程起步

编程和软件开发似乎是件很高深的学问，然而在人们普遍采用 Windows，Unix 等窗口操作系统，电脑的操作发展到了一个傻瓜化的时代时，桌面软件开发也由于 Visual Basic 系统开发程序（俗称 VB）的出现，而像儿童搭积木一样方便、直观。更吸引人的是，Visual Basic 不仅简单易学，而且功能强大，可以满足专业开发人员的绝大部分编程需求。自 VB 问世以来，微软公司对它不断改进，应用了许多新技术，使初学者和专家都能自如地运用它来进行创造性的工作。其实编程也就是这么简单的事，只要一步一步去认识它，就会发现，桌面软件开发的学习是一个并不艰难的事情，而且很多人就是这样走进编程世界的。

第1节

Visual Basic 轻松入门

一、初识 Visual Basic 6.0

VB 是一个功能强大的开发工具，它给我们一个高效、迅速的开发环境，让我们能轻松编写出想要的程序。本书的学习和使用采用 Visual Basic 6.0。

Visual Basic 6.0 是 32 位应用程序开发工具，最低配置至少是 CPU/奔腾 166、内存/32M、硬盘/70M、支持 VGA 的监视器及光驱等必须外部设备和 Windows 中文操作系统。系统环境可以是 Windows/95/98/NT/2000/XP。

（一）安装和启动 Visual Basic 6.0

找来一张 VB6.0 的安装光盘，放入 CD-ROM 驱动器，运行 Setup.exe，然后按照提示，一步步安装好 Visual Basic 6.0 系统程序。完成安装后就可以从 Windows 启动菜单中启动 Visual Basic 6.0 了。当然，要是你经常使用，还可以将它放到桌面快捷启动方式中去。

现在启动 Visual Basic，首先会弹出一个【新建工程】对话框，询问用户要创建的工程类别。现在单击【打开】按钮，即进入 Visual Basic 集成开发环境。

(二) 认识 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编写、调试、运行、编译均在该集成环境中完成（见图 1-1）。

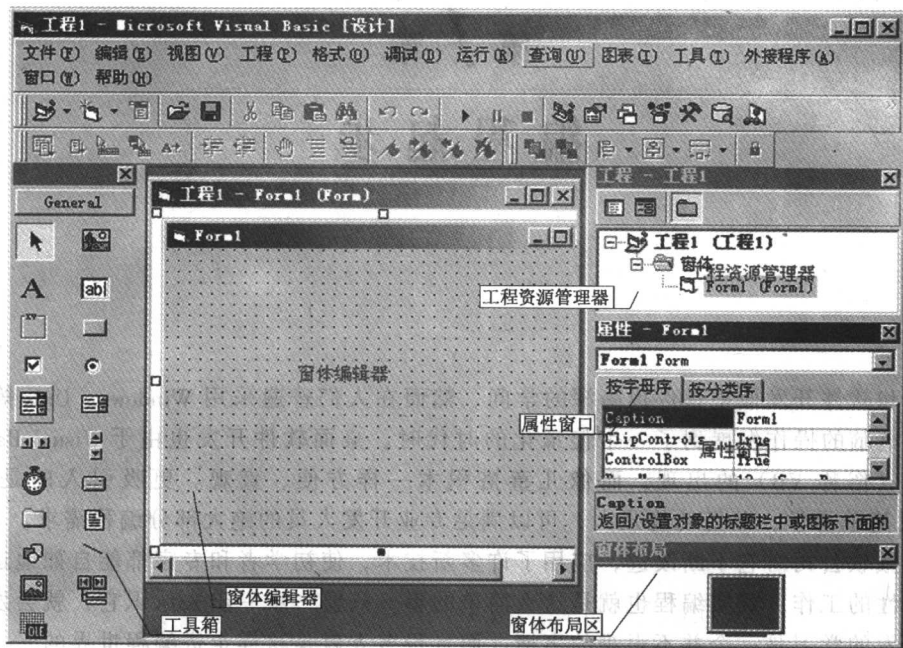


图 1-1 Visual Basic 集成开发环境

- 左面是 VB 的工具箱，里面的各种图标是我们开发程序时经常用到的标准控件。
- 正中间是窗体编辑器，又称对象窗口。我们程序的界面，大部分是在它上面设计的。
- 右上方是工程资源管理器，它能让我们对整个工程进行整体性的管理。
- 右中处的是属性窗口，陈列着程序中所涉及的各种窗体和控件的属性。
- 右下方是窗体布局区，它能够调节程序运行时窗体在屏幕中的位置。

二、完整 VB 应用程序的建立过程

文本显示器是一个最简单的 Visual Basic 应用程序，通过该程序的编制过程，用户可以了解到 Visual Basic 编程最基本的方法和步骤。同时，创建 Visual Basic 应用程序是很简单的事，很多人的软件开发都从下面这个程序开始的。我们也来看看下面这个程序的创建，实现我们的第一次编程。

我们把第一个程序称为“你好，Visual Basic”，在文本显示器中，单击【问候】按钮，界面上显示“你好，Visual Basic”，单击【退出】按钮，界面回到空白的集成环境中。

(一) 建立应用程序界面

打开 Visual Basic 6.0，在弹出的【新建工程】对话框中，选择【标准 EXE】选项，单击【打开】按钮。在窗体设计器中会打开标题栏是 Form1 的窗口，在 VB 中它叫做窗体，如图 1-2 所示（VB 启动时，系统会自创一个工程，该步可跳过）。以后要用多个窗体时，通过 VB 菜单下的【添加窗体】选项来实现。

在屏幕的左下角可以看到工具箱。将鼠标指向某一控件上，并停滞一秒钟，将会弹出一

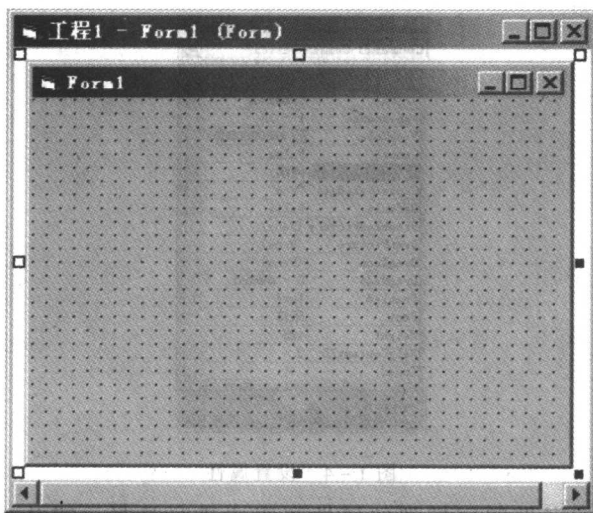


图 1-2 新建窗体

个 tip，指出该控件的名字。首先移动鼠标，寻找到命令按钮控件图标，单击后把鼠标指针移动到窗体，此时鼠标指针变成十字形。将十字光标放到适当位置并拖动光标到所需大小，这样就在窗体中添加了命令按钮控件。这里，我们绘出 Command1、Command2 两个命令按钮控件和一个 Timer 控件。如图 1-3 所示。

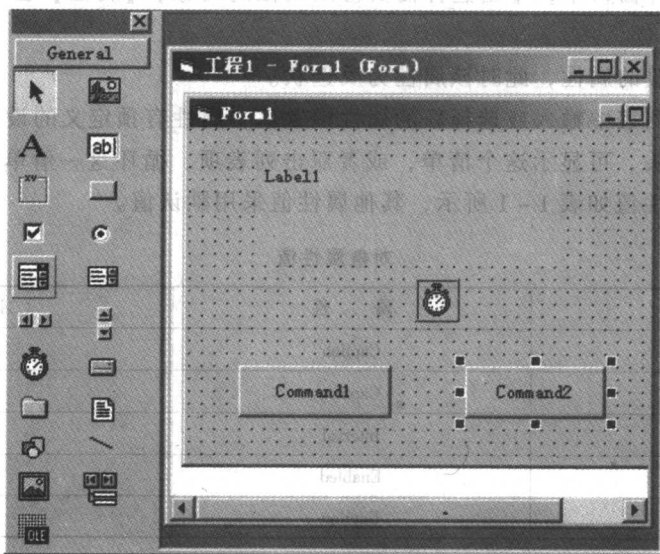


图 1-3 绘出所需要的控件

(二) 对象属性的设置

在建立用户界面后，下一步的工作为窗体的各个控件设置属性了，使它们具有各自的外观及特性。见图 1-4。

【属性】窗口内容包含如下元素。

- 对象框：显示所设置属性的对象名。单击对象框右边的箭头，可显示当前窗体的对象到表。

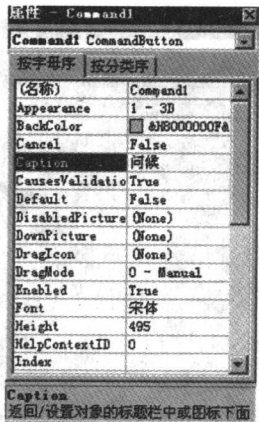


图 1-4 设置属性

- 排序方式：从按字母顺序排列的属性列表中进行选择，或从按逻辑诸如（与外观、字体或位置相关的）分类页的层次结构视图进行选择。
- 属性列表：属性各列表中显示所选对象的全部属性名，在属性值列表中可编辑和察看设计值。

现在按如下步骤设置控件的属性。

- (1) 在窗体设计窗口中，单击选择需要修改属性的对象，【属性】窗口将显示所选窗体和控件的属性值。
- (2) 选中要修改的属性，此时该属性为反选状。
- (3) 在属性值列表中输入或选择新的属性设置值，属性有预定义的设置值清单，单击设计框右边的向下箭头，可显示这个清单，或者双击列表项，循环这一清单。

需要修改的属性值如表 1-1 所示，其他属性值采用默认值。

表 1-1 对象属性值

对 象	属 性	设置值
Command1	Caption	问候
Command2	Caption	退出
Timer1	Interval	500
	Enabled	False
Label1	Caption	(空)

下面通过设置 Timer 控件的属性来说明如何设置属性。

- (1) 单击选中的 Timer 控件，或在【属性】窗口的对象框中单击三角按钮，在对象列表中选择 Timer1。
- (2) 在属性列表中选择 Interval 属性，在其属性值框中输入 500，设置 Timer 控件的计时事件间的毫秒数。
- (3) 在属性列表中选择 Enabled 属性，该属性用来确定一个窗体或控件是否能够对用户产生的事件作出反应。该属性值的设置通过在其对应属性值框出现下拉列表中选择，这里将

Enabled 属性设为 False。

依照以上方法，将 Command1、Command2、Label1 的属性分别设置如表 1-1 所示。

(三) 编写应用程序的代码

代码编辑窗口是编写应用程序 Visual Basic 代码的地方。代码由语句、常量和声明部分组成。在工程管理窗口中选择窗体或类名，然后单击【查看代码】按钮，则显示程序代码编辑窗口，如图 1-5。

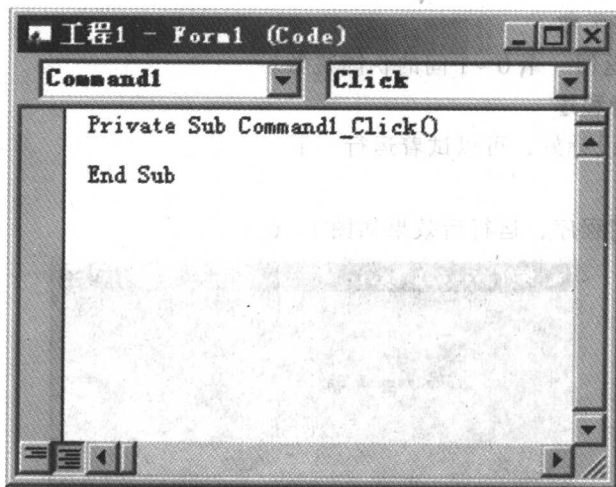


图 1-5 编写代码

还可以通过其他方法打开代码编辑窗口。例如直接双击窗体或控件。

切换到代码窗口后，在对象列表框中选择 Command1，这时在代码编辑窗口中出现如下代码行，并将指针放在这两行之间：

```
Private Sub Command1_Click ()
End Sub
```

这里，Command1 是控件名，Click 是事件名，在每次单击 Command1 按钮时，都将激发该事件。在本程序中，希望用户单击【问候】按钮时，标签控件显示“你好，Visual Basic”，并且字体颜色也随时间改变，所以在 Click 事件中添加如下代码：

```
Label1.Caption = “你好，Visual Basic”
```

```
Timer1.Enabled = True
```

用来 Label 控件的属性 Caption 显示文本。将 Timer 控件的 Enabled 属性设为 True，使其成为可以触发的事件。

在 Command2_Click 事件程序中加入下列代码：

```
Private Sub Command2_Click ()
End
End Sub
```

End 语句的功能是结束程序的运行。

在 Timer1_Timer 的事件中加入下列代码：

```
Private Sub Timer1_Timer ()
```

```
    Label1.ForeColor = RGB (Int (Rnd * 255), Int (Rnd * 255), Int (Rnd * 255))
```

```
End Sub
```

该程序改变标签中文字的颜色。其中：

- RGB () 函数返回一个表示颜色值的长整数值，其中的 3 个基本参数分别表示 3 个基本颜色（红、绿、蓝）的值，取值范围 0~255 的整数值。

- Int () 是取整函数。

- Rnd () 函数返回一个 0~1 间的伪随机数。

(四) 调试和运行程序

第一个程序已经设计好，可以试着运行一下。

1. 运行应用程序

按 F5 键运行应用程序，运行后效果如图 1-6。



图 1-6 运行效果图

2. 调试应用程序

如果在运行时，程序不能正常工作，系统会弹出一个对话框提示程序出错。这时可以通过执行工具栏中的结束按钮结束程序的执行。然后检查各个控件的属性设置以及程序代码是否有误，直到程序可以正常运行。

(五) 保存工程和生成可执行文件

1. 保存工程

(1) 打开【文件】菜单，选择【保存工程】命令，将文件按自己起好的文件名保存。（这里我们用“hello.vbp”来表示）

(2) 存储了窗体文件之后，集成环境会要求存储工程文件，按照上一步操作，将新建的工程保存为名为“XXXX”的工程文件。

(3) 要注意存盘，以防突然断电等意外因素带来数据丢失。

2. 为工程创建可执行文件

程序在编写和调试成功后，该为工程创建可执行文件了。这样就可以在没有 Visual Basic 编译环境下，运行该应用程序。

(1) 选择【文件】菜单下【生成工程·EXE】命令，出现【生成工程】对话框（见图 1-7）。

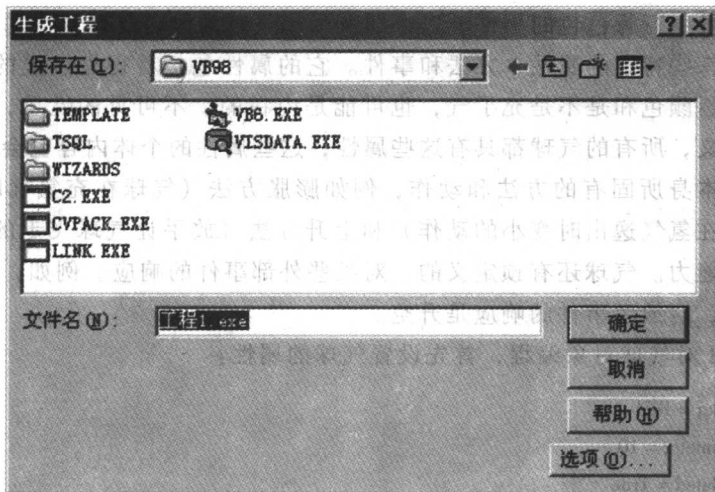


图 1-7 【生成工程】对话框

- (2) 在【文件名】文本框中输入文件名。
- (3) 单击【确定】按钮，即完成。

在确定前，看到有一个【选项】，这里就可以填入版本号等其他选项。

第2节

对象、属性、事件和方法

在学习了程序的创立和基本的编程之后，我们看看有关对象、属性、事件和方法的学习。

一、学习有关对象的概念

世界上存在的任何具体事物都可以称之为对象。从天体太阳、月亮、地球，到城市上海、北京，再到常见的物品报纸、衣服等都可以看作对象。对象是具体存在的，每个对象都是惟一的，有着独特的个性。类则是具有相同性质的对象的集合。对象和类的区别在于前者都是具体的，而后者则是抽象的。比如人可以称之为类，平常谈起的人类，绝不是指某一具体的人，而每一个具体的人，只能看作是对象。

计算机内的零配件可以看作是对象，可以任意选用，把不同类型的主板、显示器、显示卡、声卡、硬盘等组装在一起，形成各不相同的计算机，可以满足不同用户的需要。相类似的，VB6.0 中的对象也都是现成的，利用它们可以设计，并可构造出各种各样满足不同用户需求的应用程序。

二、对象的属性、事件和方法

日常生活中常见的对象，具有各种各样的属性，会做各式各样的动作(方法)，也会因

外界事物的影响发生形形色色的事件。以小朋友爱玩气球为例，气球对小朋友而言是一个具体的对象，具有一些特定的属性、方法和事件。它的属性包括看得见、摸的着的一些性质，比如说它的直径、颜色和是不是充了气，也可能是内在的、不可见的性质，例如它的寿命、归属等。通过定义，所有的气球都具有这些属性；这些属性的个体内容也会因气球不同而不同。气球还具有本身所固有的方法和动作，例如膨胀方法（气球在充氢气时变大的动作）、缩小方法（气球在氢气逸出时变小的动作）和上升方法（放手让气球飞走的动作）。所有的气球都具备这些能力。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。

现在利用 VB 对气球对象编程，首先设置气球的属性：

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

上述代码的语法是：首先是对象（Balloon）接着是属性（Color），然后是赋值（Red）。换一个不同的值就可以改变气球的颜色。对象属性也可以在程序设计阶段【属性】窗口中设置。

调用气球对象的方法是：

```
Balloon.Inflated
Balloon.Deflate
Balloon.Rise 5Inflated
```

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。一些方法有一个或多个参数，可对执行的动作做进一步的描述。

下面是气球对象对被刺了的事件的响应：

```
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，气球对象和方法：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，所以把属性设置为 False，并给 Diameter 属性设置为一个新值。

上述这种编程的方式是典型的 VB 编程方式，对窗体对象或控件对象的编程方式也与此相同。程序员决定应更改对象的哪些属性、调用哪些方法、对哪些事件做出响应，进而得到希望的外观和行为。

类似于日常生活中的对象，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，把事件作对象的响应。

窗体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 VB 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和调用程序和处理数据，就像这些应用程序是自己的代码一样。

三、事件驱动

Windows 的应用程序同 DOS 程序的工作机制有本质的不同，所谓“事件驱动”就是说只有在事件发生时，程序才会运行；在没有事件发生时，整个程序处于停滞状态的，这就是和以前的 DOS 程序的极大区别。

四、窗口、事件与消息

为了简单地了解 Windows 的工作机制，需要理解 3 个关键的概念，即窗口、事件和消息。

可以简单地将窗口看作带有边界的矩形区域。如果计算机中安装的操作系统是 Windows 操作系统，就已经了解几种不同类型的窗口。例如 Windows2000 中的【资源管理器】窗口、文字处理程序 Word 中的文档窗口或者弹出有提示信息的对话框。

Microsoft Windows 操作系统通过给每一个窗口指定一个惟一的标识符（窗口句柄）来管理窗口。操作系统连续地监视每一个窗口的活动或事件信号。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。

每发生一次事件，将引发一条消息发送到操作系统。操作系统处理该消息并广播给其他窗口。然后每一个窗口才能根据自身处理该条消息的指令而采取适当的操作，例如，当窗口解除了其他窗口的覆盖时，重显自身窗口。

可以想象，处理各种窗口、事件和消息的所有可能的组合将是惊人的工作量。幸运的是，VB 使用户摆脱了所有的低程序消息处理。许多消息由 VB 自动处理了，其他消息作为事件过程由编者自行处理。这样可以快速创建功能强大的应用程序，而不必涉及不必要的细节。

五、事件驱动模型

在传统的面向过程应用程序设计中，应用程序控制了执行哪一部分代码以及按何种顺序执行代码。从第一行代码执行程序，并按应用程序中难以实现的路径执行，必要时调用过程。在事件驱动的应用程序设计中，代码不是按照难以实现的路径执行，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，也可以由操作系统或其他应用程序的消息触发，甚至可以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可能都是不同的。

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。当作出某些假设时（例如，假设在运行处理某一输入的过程之前，该输入必须包含确定的值），应该设计好应用程序的结构，以确保该假设始终有效（例如，在输入的过程禁止使用启动该处理过程的命令按钮）。

在执行中，代码也可以触发事件。例如，在程序中改变文本框的文本，将引发文本框的