

# 可视化编程应用 Visual Basic

李涛 熊布庭 主编



北京理工大学出版社

BEIJING UNIVERSITY OF TECHNOLOGY PRESS

# 可视化编程应用

## ——Visual Basic

主 编 李 涛 熊布庭

副主编 郭旺飞

## 内容提要

本书通过大量实例,深入浅出地介绍了 Visual Basic 中文版的可视化编程环境,基本内部控件,常用 ActiveX 控件的使用以及基本程序设计及调试方法。除了介绍 Visual Basic 自身特有的功能外,作为一门计算机语言的学习,本书更注重程序设计基本知识的讲授和对编程能力的培养,将结构化程序设计的三种基本结构、数组、过程等程序设计基本概念的学习与 Visual Basic 控件的学习紧密地结合起来。

版权专有 侵权必究

---

### 图书在版编目(CIP)数据

可视化编程应用: Visual Basic / 李涛, 熊布庭主编. —北京: 北京理工大学出版社, 2016.3

ISBN 978-7-5682-1949-5

I. ①可… II. ①李… ②熊… III. ① BASIC 语言-程序设计 IV. ① TP312

中国版本图书馆 CIP 数据核字(2016)第 043703 号

---

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

82562903 (教材售后服务热线)

68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京通县华龙印刷厂

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 13.75

字 数 / 353 千字

版 次 / 2016 年 3 月第 1 版第 1 次印刷

定 价 / 44.00 元

责任校对 / 陈玉梅

责任印制 / 边心超

---

图书出现印装质量问题, 请拨打售后服务热线, 本社负责调换

随着计算机技术的不断发展，软件程序设计方法已经与早期的方式有了巨大的差别。Visual Basic 就是以面向对象程序设计的思想和方法为基础的一种常用、简单的 Windows 应用程序开发工具。因此《可视化编程应用—Visual Basic》是非计算机应用专业的一门基础课，也是一门实践性较强的课程。

通过本书的学习目的是使学生掌握一种简单、实用的编程工具，初步掌握面向对象的基本概念与高级语言的基本结构，掌握面向对象程序设计的思想和方法，提高学生动手编程能力及上机操作能力，增强学生学习程序设计的信心和兴趣，为今后进一步学习计算机程序设计打下基础。

本书以 Visual Basic 中文版为语言背景，通过大量实例，深入浅出地介绍了 Visual Basic 中文版的可视化编程环境，基本内部控件，常用 ActiveX 控件的使用以及基本程序设计及调试方法。除了介绍 Visual Basic 自身特有的功能外，作为一门计算机语言的学习，本书更注重程序设计基本知识的讲授和对编程能力的培养，将结构化程序设计的三种基本结构、数组、过程等程序设计基本概念的学习与 Visual Basic 控件的学习紧密地结合起来。

本书的讲解通俗易懂，同时与教育部考试中心制定的全国计算机等级考试二级 Visual Basic 考试大纲接轨，包含大纲中所有知识点。本书适用于具有 Windows 基本知识的计算机爱好者以及欲参加全国计算机等级考试二级 Visual Basic 的应考者使用。

限于编者的知识和经验，书中难免有错误和不当之处，请读者不吝指正。



# 目 录

模块1 可视化编程与 VB 开发环境 .....	1
任务1 可视化编程的概念 .....	1
1.1.1 面向对象的程序设计 .....	1
1.1.2 面向对象程序设计的概念 .....	2
任务2 Visual Basic 程序设计的开发平台 .....	4
1.2.1 Visual Basic 的产生和发展 .....	4
1.2.2 Visual Basic 开发平台简介 .....	5
1.2.3 Visual Basic 的基本特点 .....	5
任务3 Visual Basic 6.0 的集成开发环境 .....	6
任务4 Visual Basic 程序设计的基本步骤 .....	9
1.4.1 界面设计 .....	10
1.4.2 属性设置 .....	12
1.4.3 代码编写 .....	15
1.4.4 调试运行 .....	17
任务5 第一个 Visual Basic 应用程序 .....	18
模块2 VB 程序设计语言基础 .....	20
任务1 VB 中的数据与运算 .....	20
2.1.1 基本数据类型 .....	20
2.1.2 自定义数据类型 .....	23
2.1.3 常量与变量 .....	23
2.1.4 运算符和表达式 .....	26
2.1.5 数组 .....	29
任务2 VB 函数 .....	34
2.2.1 数学函数 .....	34
2.2.2 字符串函数 .....	35
2.2.3 转换函数 .....	36
2.2.4 判断函数 .....	37



2.2.5 其他函数 .....	38
任务3 输入与输出 .....	38
2.3.1 数据的输入 .....	38
2.3.2 数据的输出 .....	41
任务4 程序的流程控制 .....	42
2.4.1 顺序结构 .....	42
2.4.2 分支结构 .....	43
2.4.3 循环结构 .....	48
<b>模块3 Visual Basic 的过程</b> .....	<b>58</b>
任务1 过程的定义与调用 .....	58
3.1.1 过程的定义 .....	58
3.1.2 过程的调用 .....	59
任务2 参数传送 .....	60
3.2.1 形式参数和实际参数 .....	60
3.2.2 按值传递参数 .....	60
3.2.3 按地址传递参数 .....	61
3.2.4 传递一个数组 .....	62
<b>模块4 Visual Basic 的窗体与常用控件</b> .....	<b>64</b>
任务1 常用标准控件 .....	64
4.1.1 命令按钮与文本框控件 .....	64
4.1.2 复选框和单选按钮 .....	68
4.1.3 列表框和组合框 .....	73
任务2 焦点与 Tab 顺序 .....	83
4.2.1 设置焦点 .....	83
4.2.2 Tab 顺序 .....	84
任务3 其他常用控件 .....	86
4.3.1 滚动条 .....	86
4.3.2 计时器 .....	89
任务4 窗体 .....	93
4.4.1 窗体的结构 .....	93
4.4.2 窗体的属性 .....	93
4.4.3 窗体的事件与方法 .....	95



模块 5 鼠标与键盘的控制 .....	97
任务 1 鼠标的控制 .....	97
5.1.1 鼠标事件 .....	97
5.1.2 鼠标光标形状 .....	98
任务 2 键盘的控制 .....	98
5.2.1 Keypress 事件 .....	98
5.2.2 Keydown 和 KeyUp 事件 .....	98
模块 6 菜单、工具栏与状态栏 .....	111
任务 1 多文档界面 MDI .....	111
6.1.1 多窗体 .....	111
6.1.2 多文档界面 .....	115
任务 2 菜单的规划与设计 .....	121
6.2.1 下拉菜单设计 .....	122
6.2.2 弹出式菜单设计 .....	126
6.2.3 菜单项的修改 .....	128
任务 3 工具栏 .....	128
6.3.1 手工创建工具栏 .....	129
6.3.2 使用工具栏控件制作工具栏 .....	130
模块 7 文件的管理 .....	135
任务 1 文件的结构及访问类型 .....	135
7.1.1 文件结构 .....	135
7.1.2 文件分类 .....	136
任务 2 文件的处理 .....	137
7.2.1 顺序文件的处理 .....	137
7.2.2 随机文件 .....	144
7.2.3 二进制文件 .....	145
任务 3 文件处理常用命令、函数及语句 .....	146
7.3.1 文件的建立与打开 .....	146
7.3.2 文件的关闭 .....	148
7.3.3 文件指针 .....	149
7.3.4 常用的文件操作函数 .....	149
7.3.5 文件基本操作语句 .....	151



任务4 文件管理控件 .....	152
<b>模块8 图形处理与多媒体应用 .....</b>	<b>155</b>
任务1 图形处理 .....	155
8.1.1 坐标系 .....	155
8.1.2 自定义坐标系统 .....	156
8.1.3 颜色的使用 .....	158
8.1.4 图形控件的使用 .....	159
8.1.5 常用的绘图方法 .....	161
任务2 调用 Windows API 函数 .....	164
8.2.1 使用 API Viewer 查看 API .....	165
8.2.2 API 函数声明 .....	166
8.2.3 使用 API 函数实例 .....	166
任务3 多媒体应用 .....	168
8.3.1 Multimedia MCI 控件 .....	168
8.3.2 MCI 命令 .....	172
8.3.3 用多媒体控件编程 .....	172
8.3.4 使用 API 多媒体函数 .....	177
<b>模块9 数据库应用的开发 .....</b>	<b>180</b>
任务1 数据库基础 .....	180
9.1.1 数据库的基础知识 .....	180
9.1.2 建立数据库 .....	181
9.1.3 建立并维护数据库表 .....	181
任务2 用 Data 控件访问数据库 .....	187
9.2.1 Data 常用属性 .....	188
9.2.2 Data 数据控件的事件 .....	190
9.2.3 Data 数据控件的方法 .....	191
9.2.4 Data 数据控件上的对象 .....	191
9.2.5 实现数据库记录的操作方法 .....	192
任务3 使用 ADO 控件访问数据库 .....	193
9.3.1 ADO 的数据库访问技术 .....	193
9.3.2 ADO Data 控件 .....	194
9.3.3 创建数据连接 .....	194





模块 10 实 验 .....	196
实验 1 VB 环境的使用和简单应用程序设计 .....	196
实验 2 过程的使用 .....	199
实验 3 常用控件的应用 .....	200
实验 4 菜单 .....	202
实验 5 文件 .....	205



# 模块1 可视化编程与VB开发环境

随着计算机技术的飞速发展,软件编程概念发生了根本性的转变,应运而生的可视化编程技术获得了广泛应用。Visual Basic 6.0 即是可视化编程的有力工具之一。该软件是 Microsoft 推出的 Visual Studio 的重要组成部分,是集创建、编辑、链接、运行、调试为一体的软件集成开发环境。

如何更加有效地使用 Visual Basic 6.0 提供的具有强大功能的集成环境呢?

## 任务1 可视化编程的概念

随着图形用户界面(Graphical User Interface, GUI)操作系统 Windows 的广泛使用,用户不再需要记忆各种命令,而是通过鼠标对操作对象的单击、双击或拖放等即可轻松地完成相应的操作。

可视化(Visual)编程技术使得编程人员在编写应用程序时,既不需要编写大量的代码来描述组成用户界面的各种对象的外观和位置,也不需要通过反复调试运行来调整它们,而只要把由开发工具提供的各种对象拖放到屏幕上并直接调整位置、大小、颜色和字体等即可。程序运行时所见到的结果就是设计时调整的结果,也就是常说的“所见即所得(What You See Is What You Get, WYSIWYG)”。

由此可见,可视化编程犹如搭积木式的游戏,极大提高了应用程序的开发效率,降低了程序员的劳动强度。许多高级语言都相继推出了其可视化版本,这类开发工具一经推出便深受编程人员的喜爱。本书中的 Visual Basic 就是一种典型的可视化编程开发工具,除此之外,像 FoxPro、C 和 Java 等也都有它们的可视化版本。

### 1.1.1 面向对象的程序设计

面向对象的程序设计方法与编程技术不同于标准的过程化程序设计。程序设计人员在进行面向对象的程序设计时,不再是单纯地从代码的第一行编到最后一行,而是通过考虑如何创建对象,利用对象来简化程序设计,提供代码的可重用性。

对象可以是应用程序中一个自包含组件,一方面具有私有的功能,供自己使用;另一方面又提供公用的功能,供其他用户使用。对象包含数据及其代码,这比传统编写代码的方法更容易维护。

在 Visual Basic 中,应用面向对象的程序设计方法(OOP),把程序代码和数据封装起

来视为一个对象,大部分对象都是可视的。程序员在设计程序时只需用现有工具根据界面设计的要求,直接在屏幕上“画”出窗口、菜单、按钮和滚动条等不同类型的对象,并依次为这些对象设置属性。程序员编程工作的重点仅限于编写相关对象要完成的功能代码,因而程序设计的效率可极大提高。

## 1.1.2 面向对象程序设计的概念

类、对象、属性、方法、事件、事件过程、事件驱动是 Visual Basic 程序设计中最基本的概念,即面向对象程序设计的基本概念,本节将深入地一一进行讨论。

### 1. 类

类(Class)是一组用于定义对象的相关数据和方法的集合。简单地说,类是创建对象的模型,对象则是按模型生产出来的成品,是类在应用程序中的具体实例。

在 Visual Basic 中,工具箱中的每一个控件,如文本框、标签、命令按钮等,都代表一个类。当将这些控件添加到窗体上时就创建了相应的对象。由同一个类创建的对象(如文本框控件 Text1、Text2、Text3 等)具有由该类定义的公共属性、方法和事件。

### 2. 对象

对象(Object)是代码和数据的集合,可将它看作用户界面上的一个单元。例如,控件、窗体和应用程序的部件等。对象是窗体、控件和部件的统称。它具有属性、方法和事件的特征。VB 是一种可视化的面向对象的程序设计语言。

### 3. 属性

属性(Property)就是指一个对象所具有的性质、特征。这些特征可能是外在的,也可能是内在的。在 VB 中,对象的数据就保存在属性中。属性值是用来描述和反映对象特征参数。例如,控件的“名称”(Name)、“颜色”(Color)、“是否有效”(Enabled)、“位置”(Top、Left)等属性决定了对象展现给用户的界面具有什么样的外观和功能。同类的对象具有相同的属性定义,但属性值可能不同。例如,两个命令按钮,它们的标题显示分别为“确定”和“取消”,这是因为这两个命令按钮都定义有属性 Caption,它们分别取了不同的值“确定”和“取消”。不同类的对象具有不同的属性定义,例如,命令按钮定义有属性 Caption,但没有属性 Text;而文本框定义有属性 Text,但没有属性 Caption。在设计应用程序时,通过改变对象的属性值来改变对象的外观和特性。对象属性的设置可以通过以下两种办法来实现。

(1)应用程序界面设计阶段,通过选中对象,利用属性窗口直接对对象属性进行设置。

(2)在程序代码窗口,通过编写赋值语句对象的属性进行赋值,一般格式为:对象名.属性名 = 属性值。

例如,给一个命令按钮 Command1 标题显示文字改为“确定”,其程序代码为:



Command1. Caption = “确定”。

运行程序中,当执行到该语句时,对象的属性就会按程序要求发生改变。

值得注意的是,有的属性只能在属性窗口中修改,例如,Name 属性。有的属性在属性窗口中不存在,这种属性只能在代码中使用,例如,文本框的 SelText 属性。大多数属性都可以用上述两种办法设置。

#### 4. 方法

方法(Method)是对象能够主动完成的操作,每种对象能做的操作在定义类时已经确定了。方法只能在程序代码中使用,其调用格式如下:[对象名.]方法名([参数])。有的方法需要提供参数,有的方法是不带参数的。例如:

```
Form1. Cls '清除窗体 Form1 上显示的内容
```

#### 5. 事件

事件(Event)是由 Visual Basic 系统预先设置好的、能够被对象识别的动作,即对象被动接受的动作,每种对象能接受的动作也是在定义类时确定的。例如,单击(Click)事件、双击(Db1Click)事件等。每一种对象能识别的事件在设计阶段可以从代码窗口中该对象的过程框的下拉列表框中看到。对象的事件可以由用户触发,如单击、按某个键等;也可以由系统或应用程序触发,如装载窗体、卸载窗体等。

#### 6. 事件过程

对象响应事件后就会执行一段程序代码,这样的代码称为事件过程或事件驱动程序。一个对象可以识别一个或多个事件,因此可以使用一个或多个事件过程对相应的事件做出响应。事件过程的一般格式如下:

```
Private Sub 对象名—事件名([参数表])
```

处理事件的程序代码 End Sub 例如:

```
Private Sub Form_Click()
```

```
Print"欢迎您来到 Visual Basic 世界!"
```

```
End Sub
```

虽然对象拥有许多事件过程,但程序设计者并不需要去为每个事件都编写事件过程,只需要编写自己想要触发的那些事件过程。

#### 7. 事件驱动

Visual Basic 应用程序运行时,先装载和显示一个窗体,然后等待下一个事件(由用户操作来引发或程序运行触发)的发生。当某一事件发生时,程序就会执行此事件的事件过程,完成一个事件过程后,程序又会进入等待状态,直到下一事件发生为止。如此周而复始地执行,直到程序结束。也就是说,事件过程要经过事件的触发才能被执行,这种工作模式称为事件驱动方式。



Visual Basic 程序采用事件驱动的运行机制,是通过响应不同的事件执行不同的事件过程的程序代码段。响应的事件顺序不同,执行的程序代码段的顺序也不同,即事件发生的顺序决定了整个程序的执行流程。由于事件可以由用户触发,也可以由系统或应用程序触发,所以程序每次执行的流程都可以不同。因此,设计 Visual Basic 应用程序时,用户一定要明确哪个对象的哪个事件发生时需要机器完成哪些工作,进而编写一些必要的事件过程。

## 任务 2 Visual Basic 程序设计的开发平台

### 1.2.1 Visual Basic 的产生和发展

BASIC 是 Beginner's All-purpose Symbolic Instruction Code(初学者通用符号指令代码)的缩写(常写作 Basic),是国际上广泛使用的一种计算机高级语言。Basic 简单、易学,目前仍是计算机入门学习的主要语言之一。Basic 语言的产生和发展经历了以下 4 个阶段。

- (1)1946 年至 20 世纪 70 年代初:1964 年 Basic 语言问世。
- (2)1975 年至 20 世纪 80 年代中:微机上固化的 Basic 语言。
- (3)20 世纪 80 年代中至 20 世纪 90 年代初:结构化 Basic 语言。
- (4)1991 年以来:Visual Basic 程序设计语言。

20 世纪 90 年代初,由于 Windows 操作平台的出现,PC 的操作方式开始由命令方式向图形方式转变。1991 年,微软推出了 Visual Basic 1.0 版。当时,它是第一个“可视”的编程软件。这使得程序员都尝试在 Visual Basic 的平台上进行软件创作。随着 Windows 操作平台的不断成熟,Visual Basic 产品由 1.0 版本升级到 3.0 版,利用 Visual Basic 3.0 产品可以快速地创建各种应用程序,如多媒体应用程序,各种图形操作界面等。并且从 Visual Basic 3.0 开始,微软将 Access 的数据库驱动集成到了 Visual Basic 中,极大提高了 Visual Basic 的数据库编程能力。在 Visual Basic 4.0 版本中,提供了创建自定义类模块、属性及过程功能,利用这些功能,用户可以创建自定义类和对象;还提供了强大的数据库管理能力,通过 ActiveX 数据对象(ADO)、数据访问对象(DAO)、远程数据对象(RDO)和 ODBC,用户可以访问任何一种类型的数据库。

随着互联网的出现和迅速发展,微软公司推出了 ActiveX 技术,并且将这一技术引入到了 Visual Basic 6.0 版本中,Visual Basic 6.0 对面向对象编程的技术也做了许多扩展,用户可以自定义对象所处理的事件,还可以定义构造函数和析构函数。另外,还提供了用组件编程的概念,实际上是对象编程的扩展。如今,Visual Basic 已经成为一种真正专业化的开发语言。用户不仅可以用 Visual Basic 快速创建 Windows 程序,还可以编写企业水平的客户/服务器程序及强大的数据库应用程序。



## 1.2.2 Visual Basic 开发平台简介

Visual Basic 6.0 中文版包括3种版本,分别为 Visual Basic 学习版(Learning), Visual Basic 专业版(Professional)和 Visual Basic 企业版(Enterprise),这些版本是在相同的基础上建立起来的。

Visual Basic 学习版不要求用户具有编程经验。这个版本是为学生、业余爱好者和其他想更多地了解基于 Windows 的应用程序是如何开发的人而设计的。利用它可以轻松地开发 Windows 的应用程序。这个版本包括 Visual Basic 内部控件、网格控件、表格控件和数据库控件。

专业版是为需要创建客户/服务器应用程序或能访问 Internet 的应用程序的个体专业人员或公司开发人员设计的。它包括了一整套完备的开发工具。该版本包括学习版的全部功能以及 ActiveX 控件,还包括 Internet 控件和 Crystal Report Writer。

企业版是为要创建分布式、高性能的客户/服务器应用程序或 Internet 及 Internet 上的应用程序的开发组设计的。该版本包括专业版的全部功能连同自动化管理器、部件管理器、数据库管理工具及 Microsoft Visual SourceSafe 面向工程版的控制系统等。

## 1.2.3 Visual Basic 的基本特点

“Visual”指的是可视化、图形化的应用程序开发方法,它不需要编写大量代码去描述界面元素的外观和位置,而只要把预先建立的对象拖放到屏幕上即可。这是 Visual Basic 最显著的特点。Visual Basic 是一个强大的在 Windows 平台上使用的开发工具,从开发个人或小组使用的小工具,到大型企业应用系统,甚至通过 Internet 遍及全球的分布式应用软件,都可以使用 Visual Basic 语言进行开发。Visual Basic 之所以有这么广泛的用途是因为它具有以下特点。

(1)真正的面向对象编程,使开发人员在维护系统运行时只需要修改很少的代码,同时也就加快了系统开发的速度。

(2)可视化的编程方法,以及向导的功能,使开发人员几乎不用输入太多的代码就可以开发 Windows 程序。

(3)数据访问特性允许对包括 Microsoft SQL Server 和其他企业数据库在内的大部分数据库格式建立数据库和前端应用程序。

(4)通过 ActiveX 技术可以使用其他应用程序提供的功能,例如,Microsoft Word 字处理软件,Microsoft Excel 电子表格及其他 Windows 应用程序,甚至可以直接使用 Visual Basic 创建应用程序和对象。

(5)支持 Internet,很容易通过 Internet 访问文档和应用程序。

(6)已经完成的应用程序是真正的 \*.exe 文件,并提供运行时的可自由发布的动态链接库(DLL)。

## 任务 3 Visual Basic 6.0 的集成开发环境

Visual Basic 6.0 的集成开发环境如图 1-1 所示,下面把这个集成开发环境窗口分解开来,逐一介绍。

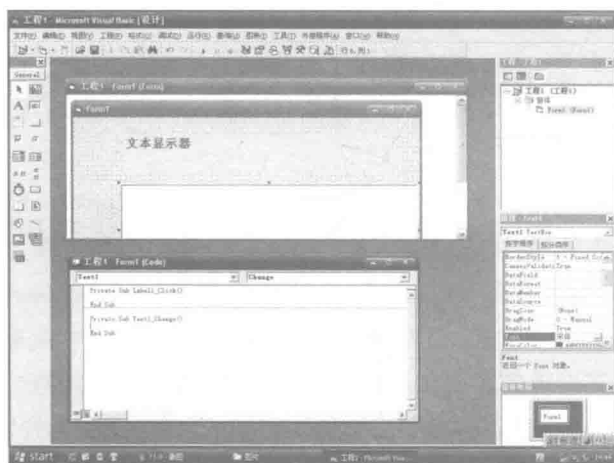


图 1-1 Visual Basic 6.0 开发环境

### 1. 标题栏

如图 1-2 所示,窗口标题栏显示当前正在做什么。标题栏为蓝灰色,表示当前处于活动状态。



图 1-2 标题栏

### 2. 菜单栏

菜单栏如图 1-3 所示。其中显示了所有的 Visual Basic 菜单命令。



图 1-3 菜单栏

在 VB 6.0 中,菜单栏上的菜单是下拉式的。菜单栏上的菜单还可以用按 Alt + 菜单名称后带下画线的字符键的方式激活。下拉菜单中如果菜单命令选项右边带一个三角符号,则表示它还有下一级子菜单;带省略号,则表示选择该菜单命令后会弹出一个对话框。按 ↑、↓ 键移动下拉菜单中的蓝条到所需的选项上,按 Enter 键,就可以执行相应的菜单命令。

### 3. 工具栏

工具栏如图 1-4 所示,它提供了常用命令的快速访问方式。单击工具栏上的按钮,

就会执行按钮所代表的操作。按照默认规定,启动VB之后即显示标准工具栏。附加编辑,窗体设计和调试的工具栏可以从“视图”菜单上的“工具栏”命令中显示或隐藏。



图1-4 工具栏

#### 4. 窗体设计窗口

窗体设计窗口用于设计应用程序界面,如图1-5所示。在该窗口中,可以添加控件、图形和图像来创建所需要的各种应用程序的外观。应用程序的每个窗体都拥有自己的窗体设计窗口。

窗体的上方是标题栏,系统初始化后默认的窗体称为“Form1”,其中带网格点的窗体称为窗体设计器。一个应用程序可以有一个窗体,也可以有多个窗体。每一个窗体都有自己的窗体设计窗口。

#### 5. 工程资源管理器

在窗体右边是“工程”窗口、“属性”窗口和“窗体布局”窗口。默认状态下,这3个窗口排在同一列上。“工程”窗口也称工程资源管理器窗口(Project Explorer)。在该窗口中,可以看到装入的工程以及工程中的项目,如图1-6所示。“工程”窗口是一个活动的窗口,可以单击其标题栏,然后按住鼠标左键,任意移动。单击“工程”窗口标题栏上的“关闭”按钮,关闭该窗口。需要查看时,可单击“视图”菜单,选择“工程窗口”命令。

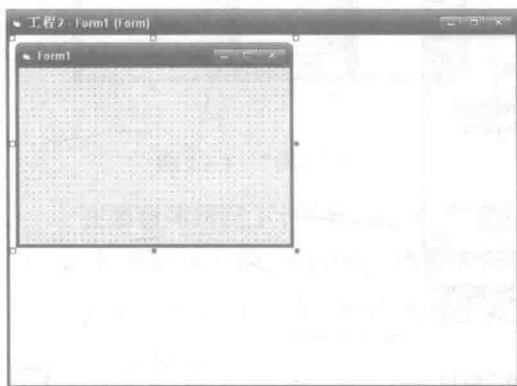


图1-5 窗体设计窗口

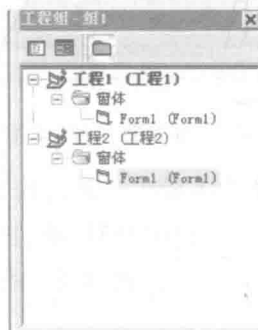


图1-6 工程窗口

工程窗口中列出了已经装入的工程以及工程中的项目。工程中的项目可分为如表1-1所示的9类。



表 1-1 工程所包含的项目

项目名称	说明
工程	工程及其包含的项目
窗体	所有与此工程有关的 .frm 文件
标准模块	工程中所有的 .bas 模块
类模块	工程中所有的 .cls 模块
用户控件	工程中所有的用户控件
用户文档	工程中所有的 ActiveX 文档, 即 .doc 文件
属性页	工程中所有的属性页, 即 .pag 文件
相关文档	列出所有需要的文档(在此存放的是文档的路径而不是文档本身)
资源	列出工程中所有的资源

## 6. 属性窗口

VB 6.0 中,窗体及窗体上的每个控件都用不同的属性描述。每个对象的属性可以通过“属性”窗口中的属性项改变或设置,也可以在程序代码中进行设置。在初始化时,每个控件都有一组默认的值,称作默认值。“属性”窗口如图 1-7 所示。



图 1-7 “属性”窗口

对象框显示当前的对象名,并附上所属的控件类。对象框右边有一个下拉按钮,单击该按钮,VB 6.0 在其下拉列表中列出本窗体上所有控件的名称及所属的类。

属性列表框是“属性”窗口的主体。属性列表框上有两个选项卡,一个是按字母顺序排列的属性,另一个是按逻辑(诸如与外观、字体或位置有关)分类的层次结构视图。由于属性较多,可以用滚动条进行翻页查看。可根据习惯选用“按字母序”选项卡或“按分类序”选项卡,属性设置结果是相同的。属性列表框中左列显示所选对象的全部属性名,右列是可编辑和查看的设置属性的值。