



高职高专“十一五”规划教材

计算机类

Visual Basic 程序设计

刘战鹏 主编



冶金工业出版社

www.cnmp.com.cn

高职高专“十一五”规划教材·计算机类

Visual Basic 程序设计

主 编 刘战鹏
副主编 靳广斌

北 京
冶金工业出版社
2008

内 容 简 介

本书是作者针对高职高专的特点,根据多年从事计算机职业教育的经验编写而成的。本书基于 Visual Basic 6.0 介绍面向对象编程语言的集成开发环境,兼顾理论基础及应用实例讲解。全书共分为 6 章,包括 Visual Basic 基础知识、控件的固有属性和结构程序设计、循环控制结构、对象及其应用、Visual Basic 数据库编程和 ActiveX 控件等。编写中特别强调培养学生的程序设计能力,各章均附有实训操作及大量习题,以突出高职教育的特点。

本书适合供高职高专院校学生学习面向对象编程语言的开发和程序设计,并作为教学教材和实验指导教材,也适合广大计算机爱好者和 Visual Basic 高级程序语言的初学者使用。

图书在版编目(CIP)数据

Visual Basic 程序设计/刘战鹏主编. —北京:冶金工业出版社,
2008.6
ISBN 978-7-5024-4682-6

I. V… II. 刘… III. BASIC 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2008)第 103510 号

出 版 人 曹胜利

地 址 北京北河沿大街嵩祝院北巷 39 号,邮编 100009

电 话 (010)64027926 电子信箱 postmaster@cnmip.com.cn

责任编辑 马文欢

ISBN 978-7-5024-4682-6

北京天正元印务有限公司印刷;冶金工业出版社发行;各地新华书店经销

2008 年 6 月第 1 版,2008 年 6 月第 1 次印刷

787mm×1092mm 1/16;12 张;309 千字;184 页;1-3000 册

29.00 元

冶金工业出版社发行部 电话:(010)64044283 传真:(010)64027893

冶金书店 地址:北京东四西大街 46 号(100711) 电话:(010)65289081

(本书如有印装质量问题,本社发行部负责退换)

前 言

美国 Microsoft 公司开发的 Visual Basic 源于 BASIC 编程语言，用于编写 Windows 平台上使用的应用程序。它拥有图形用户界面和快速应用程序开发系统，可以轻松连接数据库或创建 ActiveX 控件，是一种支持面向对象、采用事件驱动的程序设计语言，近年来在国内外得到迅速的推广使用。

本书作者在多年教学经验的基础上，根据学生的认知规律精心组织了 Visual Basic 6.0 教材内容，并通过大量具有现实意义的例题，深入浅出地介绍了结构化的程序设计语言、面向对象的程序设计方法和事件驱动机制的程序设计思想和编程技巧。

本书注重操作技能，实用性强，条理清晰，内容完备，知识系统，通过通俗易懂的语言描述，并配有大量习题，供读者边学边练，立竿见影地达到学习效果，是一本不可多得的高等院校计算机教材。本书适合作为高职高专各专业学生学习面向对象程序语言开发和程序设计的教学教材和实验指导教材，同时也适合广大计算机爱好者和 Visual Basic 高级程序语言的初学者使用。

本书由刘战鹏任主编，靳广斌任副主编，霍福华参加编写。

由于编者水平所限，书中如有不足之处敬请使用本书的师生与读者批评指正，以便修订时改进。如读者在使用本书的过程中有其他意见或建议，恳请向编者(bjzhangxf@126.com)踊跃提出宝贵意见。

编 者

《Visual Basic 程序设计》

编委会

主 编

顾沈明 叶其宏 郑 芸

副主编

任守华 纪玉玲 于晓强

编委会成员(按姓氏笔画为序)

文仕江 卢 畅 冯相忠 刘 军 毕振波

吴远红 李 东 张建科 张艳艳 杨永华

陈荣品 陈洪涛 范银琛 姚笑秋 徐妙君

徐光明 袁 燕 章毓凤 黄海峰 康 萍

谭小球 管林挺

参编学校

浙江海洋学院

黑龙江八一农垦大学

大连工业大学

编 者

2007年11月

目 录

第 1 章 Visual Basic 基础 1

1.1 VB 开发环境及帮助文档	1
1.1.1 VB6 IDE	1
1.1.2 熟悉 VB6 IDE	2
1.1.3 使用 MSDN	2
1.1.4 文件的保存	3
1.2 OOP	5
1.2.1 VB6 固有控件	5
1.2.2 OOP 简介	6
1.2.3 对象的属性	6
1.2.4 对象的事件	8
1.3 基本语法	9
1.3.1 变量与常量	9
1.3.2 过程	12
1.3.3 变量的作用范围	14
1.3.4 系统函数和颜色常数	15
1.3.5 输入/输出对话框函数	17
1.4 实训	18
习题	22

第 2 章 控件的固有属性和结构

程序设计 29

2.1 固有控件属性的应用	29
2.1.1 标签控件	29
2.1.2 文本框控件	31
2.1.3 命令按钮控件	34
2.1.4 复选框	36
2.1.5 单选按钮	37
2.1.6 列表框控件	39
2.1.7 组合框控件	41
2.2 选择结构程序设计	43
2.2.1 单行 If 语句	43
2.2.2 If 语句	45
2.2.3 块 If 语句的嵌套	47

2.2.4 Else If 语句	48
------------------------	----

2.2.5 Select Case 语句	49
----------------------------	----

2.3 实训	51
--------------	----

习题	57
----------	----

第 3 章 循环控制结构 65

3.1 滚动条控件	65
3.1.1 基本功能	65
3.1.2 实例	65
3.1.3 说明	67
3.2 计时器控件	68
3.2.1 基本功能	68
3.2.2 实例	68
3.2.3 说明	70
3.3 过程	70
3.3.1 参数传递	70
3.3.2 过程定义	71
3.4 Do while...Loop 循环控制	72
3.5 数组	73
3.5.1 一维数组	73
3.5.2 动态数组	75
3.5.3 多维数组	75
3.5.4 数组应用	75
3.6 For...Next 循环控制	78
3.7 实训	81
习题	88

第 4 章 对象及其应用 94

4.1 CommonDialog 控件	94
4.1.1 打开对话框	95
4.1.2 保存对话框	96
4.1.3 颜色对话框	96
4.1.4 字体对话框	97
4.2 控件数组	98

4.3 FSO 对象	100	5.4 系统开发实例：学生信息 管理系统.....	146
4.3.1 创建 FileSystemObject 对象	100	5.4.1 系统功能	147
4.3.2 Drive 对象	101	5.4.2 系统实现过程	147
4.3.3 Folder 对象.....	102	习题.....	153
4.3.4 File 对象.....	102	第 6 章 ActiveX 控件	160
4.3.5 TextStream 对象.....	103	6.1 MSHFlexGrid 控件	160
4.3.6 读写顺序文件	103	6.1.1 MSHFlexGrid 控件的 常用属性	160
4.4 Word 对象、Excel 对象	104	6.1.2 MSHFlexGrid 控件的 重要方法	161
4.4.1 Word 对象	104	6.1.3 MSHFlexGrid 控件的 主要事件	162
4.4.2 Excel 对象	108	6.1.4 MSHFlexGrid 控件示例	162
4.5 Menu 控件	116	6.2 TreeView 控件.....	163
4.5.1 标准下拉菜单	116	6.2.1 TreeView 控件的常用属性 ...	164
4.5.2 弹出式菜单	117	6.2.2 TreeView 控件的重要方法 ...	164
4.6 实训.....	119	6.2.3 TreeView 控件的主要事件 ...	165
习题.....	123	6.2.4 TreeView 控件示例	165
第 5 章 Visual Basic 数据库编程	127	6.3 ListView 控件.....	170
5.1 使用 ADO 控件创建数据库应用程序	127	6.3.1 ListView 控件的常用属性 ...	170
5.1.1 ADO 数据控件.....	127	6.3.2 ListView 控件的重要方法 ...	171
5.1.2 数据绑定控件	131	6.3.3 ListView 控件的主要事件 ...	171
5.2 基本 SQL 语法	132	6.3.4 ListView 控件示例	171
5.2.1 Select 语句	132	6.4 API	174
5.2.2 Insert、Update 和 Delete	137	6.5 实训.....	179
5.2.3 Delete 语句.....	138	习题.....	182
5.3 ADO 对象模型	139	参考文献	184
5.3.1 Connection 对象和 RecordSet 对象	139		
5.3.2 Command 对象.....	146		

习题九	173
第 10 章 界面设计	175
10.1 通用对话框	175
10.2 菜 单	179
10.3 多文档界面	183
10.4 工具栏和状态栏	186
10.5 应用程序向导	188
习题十	191
第 11 章 图形操作	193
11.1 坐标系统	193
11.2 图形显示与绘图控件	195
11.3 颜色与绘图方法	198
11.4 其他与绘图相关的属性、事件与方法	207
11.5 应用举例	211
习题十一	222
第 12 章 数据库与数据访问	224
12.1 数据库概念	224
12.2 数据库管理器	226
12.3 数据控件	229
12.4 ADO 数据控件	235
12.5 结构化查询语言	238
12.6 报表制作	241
12.7 数据库应用案例	245
习题十二	251
课后习题参考答案	253

第 1 章 Visual Basic 基础

1.1 VB 开发环境及帮助文档

Visual Basic(以下简称 VB)的工作环境是指集成开发环境(IDE)，这是因为它在一个公共环境里集成了许多不同的功能，例如，设计、编辑、编译和调试。

1.1.1 VB6 IDE

进入 Visual Basic 6.0 (VB6) IDE 有如下几种方式：

- 执行 VB6.exe 程序，打开如图 1-1 所示的“新建工程”对话框；单击“打开”按钮后，进入 VB6 IDE 开发环境，如图 1-2 所示。

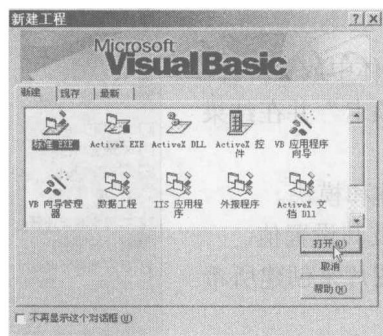


图 1-1 “新建工程”对话框

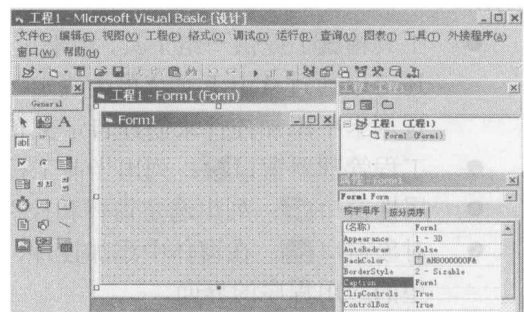


图 1-2 VB6 IDE 开发环境

- 执行 VB6.exe 程序，打开“新建工程”对话框时单击“最新”标签，打开“最新”选项卡，如图 1-3 所示；选中 VB 工程名，单击“打开”按钮后，进入 VB6 IDE 开发环境，如图 1-4 所示。

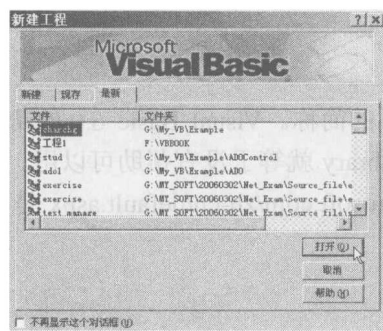


图 1-3 “最新”选项卡

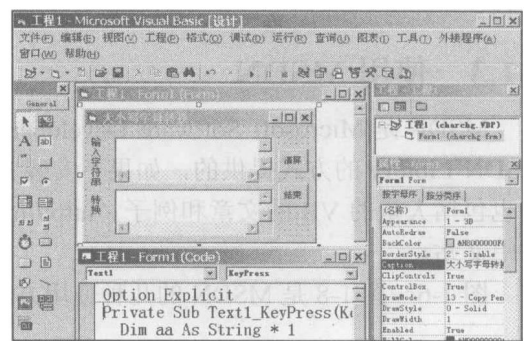


图 1-4 Visual Basic 集成开发环境

- 双击某个 VB 工程文件(.vbp)。

1.1.2 熟悉 VB6 IDE

在开发应用程序时,要使用工程来管理构成应用程序的所有不同的文件。一个工程包括如下内容:

- 跟踪所有部件的工程文件(.vbp)。
- 每个窗体的文件(.frm)。
- 每个窗体的二进制数据文件(.frx),它含有窗体上控件的属性数据。
- 每个类模块的文件(.cls),该文件是可选项。
- 每个标准模块的文件(.bas),该文件是可选项。
- 一个或多个包含 ActiveX 控件的文件(.ocx),该文件是可选项。
- 单个资源文件(.res),该文件是可选项。

工程文件就是与该工程有关的全部文件和对象的清单,每次保存工程时,这些信息都要被更新。所有这些文件和对象也可供其他工程共享。

Visual Basic 集成开发环境(IDE)主要由以下元素组成(如图 1-4 所示):

- 工具箱:提供一组工具,用于设计时在窗体中放置控件。还可以通过从上下文菜单中选定“添加选项卡”并在结果选项卡中添加控件来创建自定义布局。
- 工程管理器窗口:列出当前工程中的窗体和模块。
- 属性窗口:列出选定窗体和控件的属性及其设置值。
- 对象窗口:在窗体中添加控件、图形和图片来创建所希望的应用程序的界面。
- 立即窗口、本地窗口和监视窗口:这些附加窗口是为调试应用程序提供的。

以上窗口,除了通过“工具栏”打开外,也可通过打开“视图”菜单打开(如图 1-5 所示)。

应用程序的每个窗体或标准模块都有一个单独的代码编辑器窗口。

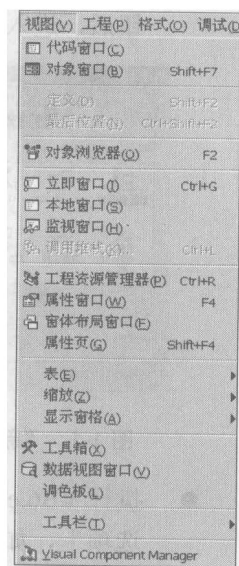


图 1-5 “视图”菜单

1.1.3 使用 MSDN

MSDN 是 Microsoft Software Developer Network 的简称。Visual Basic 6.0 的帮助是以 MSDN Library 的方式提供的,如果不安装 MSDN Library 就等于没有帮助可以用。MSDN 中也包括大量的 VB 的文章和例子。<http://msdn2.microsoft.com/zh-cn/default.aspx> 是 MSDN 的中文站点。

图 1-6~图 1-8 是 MSDN 的几种使用方法。

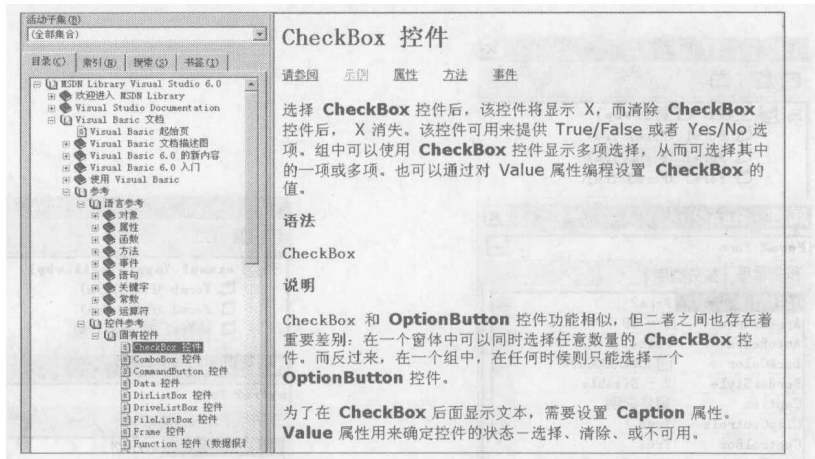


图 1-6 MSDN 的目录方式

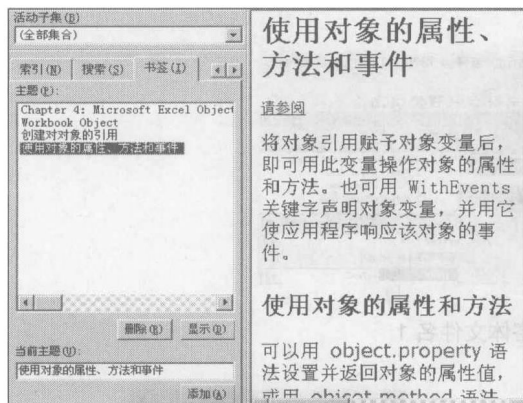


图 1-7 MSDN 的书签方式

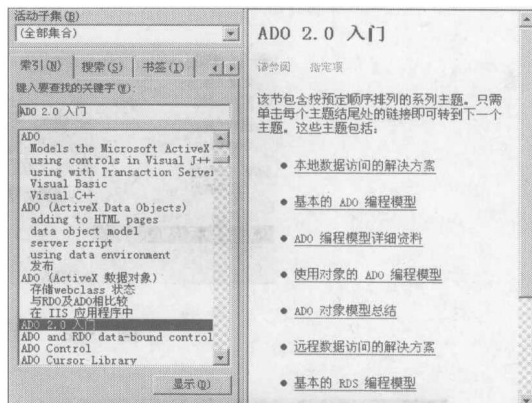


图 1-8 MSDN 的索引方式

1.1.4 文件的保存

我们首先要区分对象名与文件名，如图 1-9 所示。实现步骤如图 1-10~图 1-15 所示。

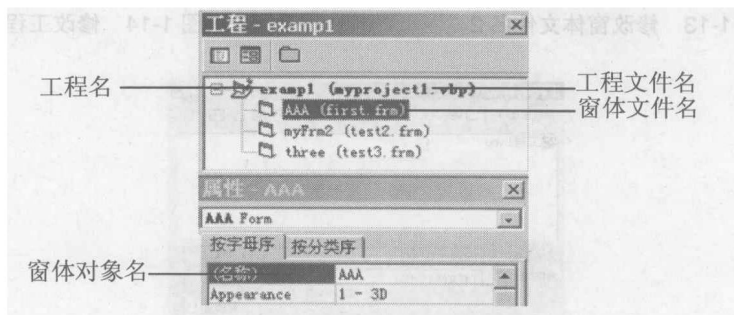


图 1-9 工程窗口对象



图 1-10 系统初始默认

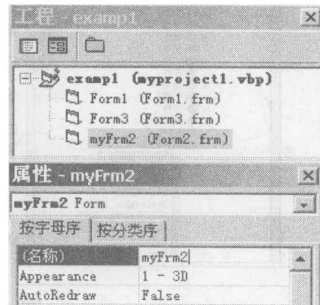


图 1-11 修改窗体名

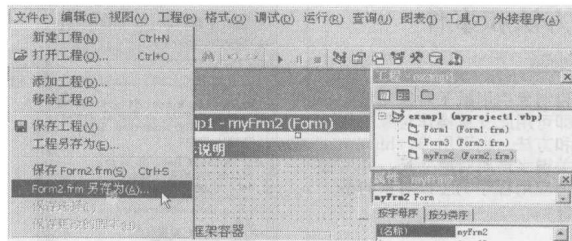


图 1-12 修改窗体文件名 1

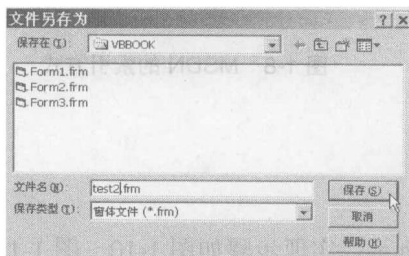


图 1-13 修改窗体文件名 2

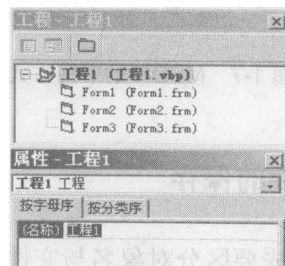


图 1-14 修改工程名

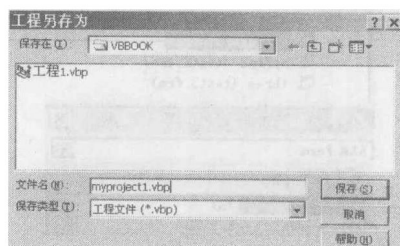


图 1-15 修改工程文件名

1.2 OOP

1.2.1 VB6 固有控件

常用固有控件如下：

- 标签(Label 控件)：控件图标 。
- 文本框(TextBox 控件)：控件图标 。
- 框架(Frame 控件)：控件图标 。
- 命令按钮(CommandButton 控件)：控件图标 。
- 复选框(CheckBox 控件)：控件图标 。
- 单选框(OptionButton 控件)：控件图标 。
- 列表框(ListBox 控件)：控件图标 。
- 组合框(ComboBox 控件)：控件图标 。
- 水平滚动条(HScrollBar 控件)：控件图标 。
- 垂直滚动条(VScrollBar 控件)：控件图标 。
- 定时器(Timer 控件)：控件图标 。
- 线条(Line 控件)：控件图标 。

固有控件如图 1-16 所示。

ComboBox 控件

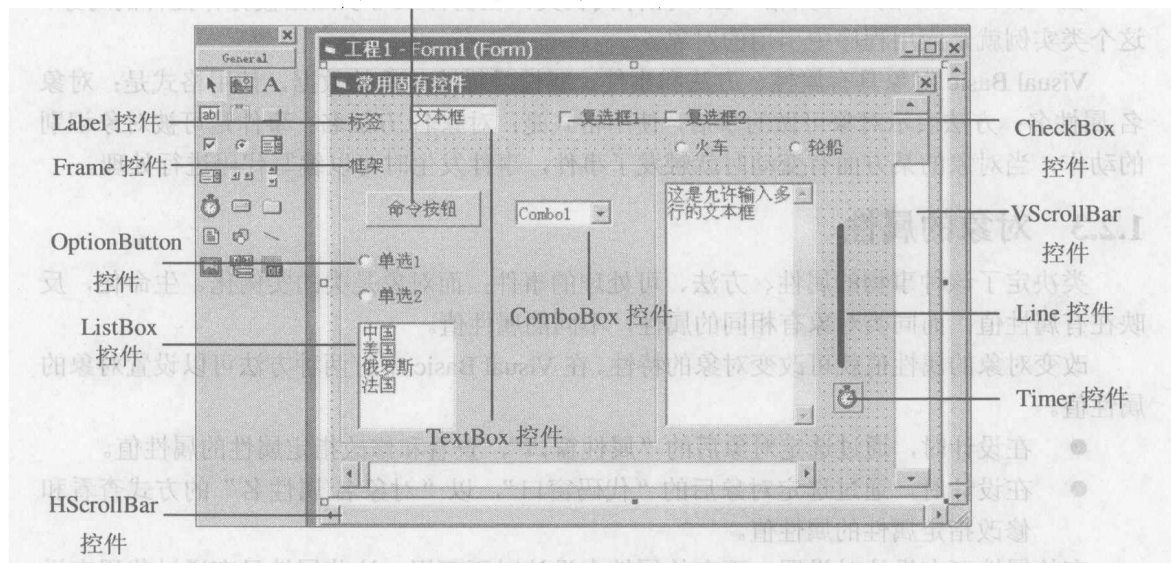


图 1-16 设计时固有控件

常用固有控件在运行时的状态如图 1-17 所示。

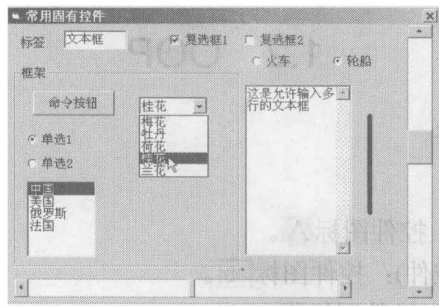


图 1-17 运行时固有控件

1.2.2 OOP 简介

OOP(Object Oriented Programming)即面向对象程序设计。类(class)定义了对象特征及对象外观和行为的模板。对象是代码和数据的组合，可以作为一个单位来处理。

在 Visual Basic 中，对象是由类创建的，因此对象被说成是类的一个实例。对象和它的类之间的关系如同饼干模子和饼干之间的关系：饼干模子是类，它确定了每块饼干的特征，比如大小和形状；用类创建对象，对象就是饼干。又如人和“张三”：人有身高、体重、性别，而“张三”也有身高、体重、性别，但是人的身高无具体值，而“张三”的身高有具体值；我们说人是一种抽象、是类，而“张三”是这种抽象的具体化、实例化，有该种类所确定的属性的具体值，因此称之为该类的一个对象。

在 Visual Basic 的“工具箱”上，控件代表类。在窗体上可以画出被称作控件的对象，这个类实例就是应用程序中引用的对象。

Visual Basic 对象具有属性、方法和事件。属性是描述对象的数据，使用格式是：对象名.属性名；方法表示对象可做的事情，使用格式是：对象名.方法名；事件是可被对象识别的动作，当对象的某方面有变动时就触发了事件，事件发生时可以编写代码进行处理。

1.2.3 对象的属性

类决定了该种事物的属性、方法、可处理的事件；而对象是类的实例化、生命化，反映在有属性值；而同类对象有相同的属性，不同的属性值。

改变对象的属性值就可改变对象的特性。在 Visual Basic 中有两种方法可以设置对象的属性值。

- 在设计时，通过选定对象后的“属性窗口”，查看和修改指定属性的属性值。
- 在设计时，通过选定对象后的“代码窗口”，以“对象名.属性名”的方式查看和修改指定属性的属性值。

有的属性可在设计时设置，而有的属性在设计时不可用，这些属性只有通过代码在运行时设置。

- 在运行时可以设置并可获得值的属性叫做读写属性。
- 在运行时只能读取的属性叫做只读属性。

如表 1-1 中的 name 属性就是只读属性，表中符号 ✓ 代表控件拥有该种属性。

表 1-1 控件的常见属性

控 件 属 性	窗 体	标 签	文 本 框	框 架	命 令 按 钮	复 选 框	单 选 框	列 表 框	组 合 框	水 平 滚 动 条	垂 直 滚 动 条	定 时 器	线 条
Name	√	√	√	√	√	√	√	√	√	√	√	√	√
Top	√	√	√	√	√	√	√	√	√	√	√	√	
Left	√	√	√	√	√	√	√	√	√	√	√	√	
Width	√	√	√	√	√	√	√	√	√	√	√		
Height	√	√	√	√	√	√	√	√	√	√	√		
Enabled	√	√	√	√	√	√	√	√	√	√	√	√	
Visible	√	√	√	√	√	√	√	√	√	√	√		√
Caption	√	√		√	√	√	√						
BackColor	√	√	√	√		√	√	√	√				
ForeColor	√	√	√	√		√	√	√	√				
FontName	√	√	√	√	√	√	√	√	√				
FontSize	√	√	√	√	√	√	√	√	√				
FontBold	√	√	√	√	√	√	√	√	√				
FontItalic	√	√	√	√	√	√	√	√	√				
FontStrikethru	√	√	√	√	√	√	√	√	√				
FontUnderline	√	√	√	√	√	√	√	√	√				
TabIndex		√	√	√	√	√	√	√	√	√	√		
ToolTiptext		√	√	√	√	√	√	√	√				

表 1-2 属性含义

属 性	属 性 含 义
Name	对象的名字
Top、Left	对象的内部上边界(左边)与其容器的上边界(左边)之间的距离
Width、Height	对象的宽度与高度
Enabled	其值决定运行时对象是否能够对用户产生的事件作出反应 如 myopt.Enabled=false, 则对象 myopt 不可用
Visible	其值决定运行时对象为可见或隐藏 如 myopt.Enabled=false, 则运行时对象 myopt 不可见
Caption	对象的标题栏或图标文本
ForeColor 、BackColor	对象文本的前景色(背景色)
FontName	字体
FontSize	字号
FontBold	是否为粗体
FontItalic	是否为斜体
FontStrikethru	是否带删除线
FontUnderline	是否带下划线
TabIndex	对象的 Tab 键次序
Tooltiptext	运行时当鼠标指针在对象上暂停时, 出现的描述对象的词或短语

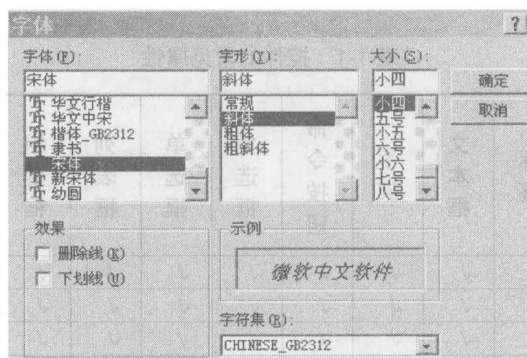


图 1-18 设计时“字体属性”对话框

1.2.4 对象的事件

事件(event)是由对象识别的一个动作,我们可以编写相应的代码对此动作进行响应。事件可以通过诸如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。

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。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。

Visual Basic 的每一个窗体和控件都有一个预定义的事件过程集。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 Visual Basic 调用该代码。代码部分(即事件过程)与每个事件对应。想让控件响应事件时,就把代码写入这个事件的事件过程之中。例如,在程序中改变文本框 ppt 中的文本将引发文本框的 Change 事件,如果 ppt_Change() 事件过程中包含有代码,则将导致该代码的执行。

表 1-3 事件

事件名	事件被触发的动作	标签	文本框	窗体
Click	在一个对象上按下,然后释放一个鼠标按钮时发生	√	√	√ 在单击一个空白区或一个无效控件时发生
Change	控件的内容已经改变	√ 改变 Label 的 Caption 属性	√	×
Dblclick	一个对象上按下和释放鼠标按钮并再次按下和释放鼠标按钮时,该事件发生	√	√	√ 当双击窗体上被禁用的控件或窗体的空白区域时发生
MouseDown	当按下鼠标按钮时发生	×	√	√
MouseUp	当释放鼠标按钮时发生	×	√	√
MouseMove	在移动鼠标时发生	×	√	√
KeyDown	当一个对象具有焦点时,按下一个键时发生	×	√	×

续表

事件名	事件被触发的动作	标签	文本框	窗体
KeyUp	当一个对象具有焦点时, 松开一个键时发生	×	√	×
KeyPress	当一个对象具有焦点时, 用户按下和松开一个 ANSI 键时发生	×	√	×
GotFocus	当对象获得焦点时发生	×	√	√
LostFocus	在一个对象失去焦点时发生	×	√	√
Load	在一个窗体被装载时发生	×	×	√
Unload	当窗体从屏幕上删除时发生	×	×	√
Resize	当一个对象第一次显示或当一个对象的窗口状态改变时发生	×	×	√
Activate	当一个窗口对象成为活动窗口时发生	×	×	√
Deactivate	当一个窗口对象不再是活动窗口时发生	×	×	√

焦点是对象在任何时间接收鼠标单击或键盘输入的能力。在 Microsoft Windows 环境中, 在同一时间只有一个窗口、窗体或控件具有这种能力。“具有焦点”的对象通常会以突出显示标题或标题栏来表示。获得焦点可以通过诸如 Tab 切换, 或在运行时选择对象的用户动作, 或在代码中用 SetFocus 方法改变焦点来实现。只有当对象的 Enabled 和 Visible 属性为 True 时, 它才能接收焦点。当对象得到或失去焦点时, 会产生 GotFocus 或 LostFocus 事件。

1.3 基本语法

Visual Basic 的代码存储在模块中。模块有 3 种: 窗体模块、标准模块(文件扩展名为 .BAS)和类模块(文件扩展名为 .CLS)。简单的应用程序可以只有一个窗体, 应用程序的所有代码都驻留在窗体模块中。而当应用程序庞大复杂时, 就要另加窗体, 最终可能会发现在几个窗体中都有要执行的公共代码。因为不希望在两个窗体中重复代码, 所以要创建一个独立模块, 它包含实现公共代码的过程。独立模块应为标准模块。用类模块可以创建对象, 这些对象可被应用程序内的过程调用。

1.3.1 变量与常量

Visual Basic 使用变量来存储值。变量有名字和数据类型(确定变量能够存储的数据的种类)。Visual Basic 中常用数据类型见表 1-4。常量也存储数值, 顾名思义, 在整个应用程序执行过程中, 这些值都保持不变。常量有两种: 系统定义的常量, 如 vbBlack、vbRed、vbCrLf、vbTab、vbKeySpace 等; 用户定义的常量(用 Const 语句来声明的), 如 Const conPai = 3.141 592 65、Public Const conVer = "07.10.A"。

类型声明字符(见表 1-4)是附加到变量名上的字符, 指出变量的数据类型。如:

```
Dim MyStr$
MyStr = "Because"
```