

第 3 章 Visual Basic 应用程序界面设计

Visual Basic 6.0 应用程序界面设计就是在工程中添加窗体,在窗体中添加菜单栏、工具栏以及设计各种控件对象;在工程中设计各种对话框。本章主要介绍窗体设计以及一些常用标准控件。

3.1 Visual Basic 对象的基本属性

对象的属性特征标识了对象的物理性质,对象的行为特征描述了对象可执行的行为动作。对象的每个属性,都是与其他对象加以区别的特性,都具有一定的含义、具有一定的值。在 Visual Basic 6.0 中,有些属性是大多数对象共有的,有些属性是某些对象特有的。大多数对象共有的属性称为对象的基本属性。

3.1.1 属性值的设置方法

对象的属性值的设置方法有两种:一是通过属性窗口设置属性值;二是在应用程序运行时,通过程序代码改变属性值。

通过属性窗口设置属性值,必须先选定对象,再设置属性值。当选定属性时,或出现插入点,此时可以直接编辑修改属性值;或出现下拉列表按钮,此时在下拉列表选定属性值;或出现对话框按钮,此时单击该按钮,打开对话框,进行属性值的设置。例如,选定对象(Form1)的属性(Font)的属性窗口如图 3-1 所示。在属性窗口改变对象的外观属性值,可以立即预览对象的设置效果。

对象的有些属性没有在属性窗口中列出,有些属性值需要在运行模式中更改。这些属性值只能通过程序代码进行设置。一般采用赋值语句设置属性值,赋值语句格式如下。

[对象名.] 属性名=表达式

其中,对象名是对象在属性窗口设置的名称(Name)属性值;省略对象名表示当前对象;对象名为 Me 表示当前窗体。对象名与属性名之间由成员运算符“.”连接。该语句表示将赋值号“=”右边表达式的值赋值给左边对象的属性。

3.1.2 对象的基本属性

无论是窗体还是控件都具有表 3-1 所示的基本属性。



图 3-1 选定属性 Font

表 3-1 对象的基本属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	对象的名称	Left	对象到容器左边框的距离
ForeColor	对象的文本或图形的前景色	Top	对象到容器上边框的距离
BackColor	对象的文本或图形的背景色	Width	对象的宽度
BackStyle	对象的背景样式	Height	对象的高度
BorderStyle	对象的边框样式	Enabled	确定对象是否响应事件
Font	对象中文本的字体格式	Visible	对象是否可见

(1) 名称(Name)属性。该属性主要用来识别和访问不同的对象。在设计模式可以设置该属性值;在运行模式只能使用该属性值,不能修改该属性值。在同一工程中,窗体对象不能同名;在同一窗体中,控件对象不能同名。

(2) ForeColor 属性和 BackColor 属性。这组属性分别用来设置对象的文本或图形的前景色和背景色。在属性窗口设置时,弹出系统默认颜色和调色板,如图 3-2 所示。

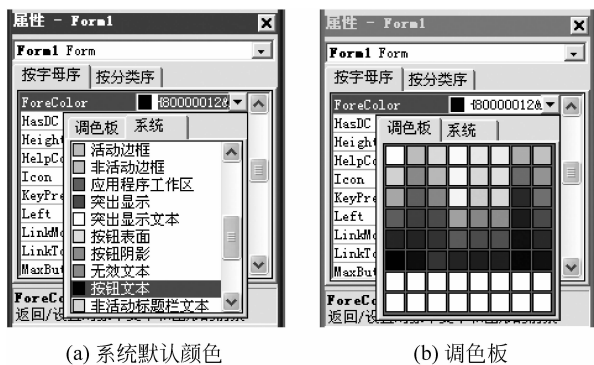


图 3-2 ForeColor 属性设置

(3) BackStyle 属性和 BorderStyle 属性。BackStyle 属性用于设置对象的背景样式,有如下两种取值。

① 当属性值为 0 时,表示透明显示;

② 当属性值为 1(默认值)时,表示不透明显示。只有背景不透明时,BackColor 属性才能有效设置。

BorderStyle 属性用于设置对象的边框样式,主要有如下两种取值。

① 当属性值为 0 时,表示对象无边框;

② 当属性值为 1 时,表示对象有单线边框。

对于不同的对象,该属性可能还有其他取值。

(4) Font 属性。该属性用于设置对象中文本的字体格式。在属性窗口设置时,弹出“字体”对话框,如图 3-3 所示。在程序代码中设置时,该对话框中各设置项分别对应以下

名称。

- ① FontName 属性：表示字体类型。
- ② FontSize 属性：表示字体大小。
- ③ FontBold 属性：表示字体为粗体。
- ④ FontItalic 属性：表示字体为斜体。
- ⑤ FontUnderLine 属性：表示加下划线。
- ⑥ FontStrikethru 属性：表示加删除线。



图 3-3 设置 Font 属性的字体对话框

(5) Left 属性、Top 属性、Width 属性和 Height 属性。在容器中创建对象时，每个对象都有位置坐标属性和大小属性。Left 属性和 Top 属性分别表示对象左上角在容器中的横坐标和纵坐标，即对象到容器的左边距和上边距。Width 属性和 Height 属性分别表示对象的宽度和高度。这组属性值的单位都是 twip，1twip=1/20 磅。

注意：能装入其他对象的对象称为容器；不能装入其他对象的对象称为控件。坐标原点(0,0)在容器的左上角，横向向右为 x 轴正方向；纵向向下为 y 轴正方向。

(6) Enabled 属性和 Visible 属性。Enabled 属性用于确定对象是否响应用户或系统产生的事件，有以下两种取值。

- ① 当属性值为 True(默认值)时，能有效响应。
- ② 当属性值为 False 时，禁止响应。

Visible 属性用于确定对象在程序运行时是否可见，有以下两种取值。

- ① 当属性值为 True(默认值)时，表示对象可见。
- ② 当属性值为 False 时，表示对象隐藏。

3.2 窗 体

窗体用于创建 Visual Basic 应用程序的界面或对话框。每个 Visual Basic 工程至少有一个窗体，窗体是各种控件的载体。

3.2.1 窗体的常用属性

窗体的常用属性如表 3-2 所示。

表 3-2 窗体的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	窗体的名称	Font	窗体中文本的字体格式
Caption	窗体标题	Picture	窗体中的背景图片
MinButton	窗体是否显示最小化按钮	WindowState	窗体运行时的显示状态
MaxButton	窗体是否显示最大化按钮	CurrentX	窗体当前位置的横坐标
BorderStyle	窗体的边框样式	CurrentY	窗体当前位置的纵坐标

(1) Caption 属性。该属性用于设置窗体标题栏的文本内容。默认值为窗体名称(Name)属性的默认值。

(2) MinButton 属性和 MaxButton 属性。这两个属性都有以下两种取值。

- ①当属性值为 True(默认值)时,表示窗体右上角有最小/最大化按钮。
- ②当属性值为 False 时,表示窗体右上角没有最小/最大化按钮。

(3) Picture 属性。该属性用于设置窗体的背景图片,使窗体界面更加美观。若通过属性窗口设置,则打开如图 3-4 所示的“加载图片”对话框,确定图片文件,打击“打开”按钮。若通过程序代码实现,则要使用图片加载函数 LoadPicture(),赋值语句格式如下。

[对象名.] Picture=LoadPicture("图片文件名")

若使用默认的“图片文件名”,则表示清空窗体的背景图片。

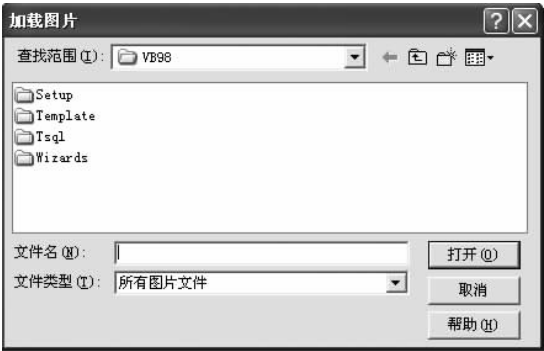


图 3-4 “加载图片”对话框

(4) WindowState 属性。该属性用于设置窗体运行时的尺寸状态,有以下 3 种取值。

- ① 当属性值为 0(默认值)时,表示窗体以有边界窗口的正常状态显示。
- ② 当属性值为 1 时,表示窗体以最小化的图标方式显示。
- ③ 当属性值为 2 时,表示窗体以无边框充满整个屏幕的最大化状态显示。

(5) CurrentX 属性和 CurrentY 属性。该组属性值只能在程序代码中引用或更改。

一般与输出方法 Print 结合使用,可以获得窗体的当前坐标值,也可以用来设置下一个输出位置坐标。该属性值以 twip 为单位。

3.2.2 窗体的常用方法

1. 方法的调用

方法是附属于对象的行为和动作。创建对象后,就可以在应用程序中调用方法。在程序代码中调用方法的一般格式如下:

[对象名.]方法名 [参数列表]

其中,[参数列表]是可选项,有些方法有参数列表;有些方法没有参数列表。

2. 窗体的常用方法

窗体的常用方法如表 3-3 所示。

表 3-3 窗体的常用方法

方法名	说 明	方法名	说 明
Print	输出	Show	显示窗体
Cls	清屏	Hide	隐藏窗体
Move	移动窗体	Reresh	刷新窗体

(1) Print 方法。该方法是窗体的一个最重要、最常用的方法,它可以将数据文本在窗体上输出。Print 方法的调用格式如下。

[对象名.] Print [输出项列表]

几点说明:

- ① 省略对象名,表示在当前窗体输出数据文本。
- ② 省略输出项列表,表示输出一个空行。
- ③ 对于具有多个输出项,若输出项之间以分号分隔,则以紧凑格式输出,即后一输出项紧接着前一输出项输出;若输出项之间以逗号分隔,则以标准格式输出,即每个输出项一般占用一个标准输出区,一个标准输出区有 14 列。
- ④ 在输出项列表中经常使用定位函数 Spc(*n*)和 Tab(*n*)。
 - 函数 Spc(*n*): 表示输出 *n* 个空格。
 - 函数 Tab(*n*): 表示将其后的输出项在第 *n* 列输出;若函数无参数 *n*,则表示将其后的输出项定位于下一个标准输出区。

使用 Print 方法输出数据文本时,数据文本总是输出在当前坐标 (CurrentX, CurrentY)处。在首次使用 Print 方法时,CurrentX 与 CurrentY 的默认属性值均为 0,即输出的数据文本显示在窗体的左上角。通过给定 CurrentX 与 CurrentY 的属性值,可以确定 Print 方法的输出位置;Print 方法的调用,会自动调整 CurrentX 与 CurrentY 的属性值。

例 3-1 Print 方法格式输出示例。

给工程添加一个窗体,窗体的标题(Caption)属性值设置为“Print 方法格式输出示例”,窗体单击事件(Click)的过程代码如下。应用程序运行结果如图 3-5 所示。

```
Private Sub Form_Click()  
    Print "紧凑格式"  
    Print "Visual"; "Basic"  
    Print  
    Print "标准格式"  
    Print "Visual", "Basic"  
End Sub
```



图 3-5 例 3-1 程序运行结果

(2) Cls 方法。该方法用于清除程序运行时在窗体或图片框中用 Print 方法或其他方法显示的数据文本或图形。调用该方法后,CurrentX 与 CurrentY 的属性值复原为 0。

(3) Move 方法。该方法用于移动窗体对象,同时还可以改变对象的大小。窗体对象的移动位置和窗体大小的改变由参数列表中对应的 Left、Top、Width、Height 的属性值确定。参数 Left 为必选参数;其他参数为可选参数。Move 方法的调用格式如下:

```
[对象名.] Move Left [,Top][,Width][,Height]
```

3.2.3 窗体的常用事件

1. 事件分类

窗体和控件对象的事件可分为以下 3 类。

- (1) 程序事件。诸如 Visual Basic 程序的装载、卸载以及打开和关闭窗体时触发的事件。
- (2) 鼠标事件。操作鼠标时触发的事件。
- (3) 键盘事件。按下键盘上的按键时触发的事件。

2. 窗体事件过程的框架结构

当输入编辑窗体事件代码时,打开代码窗口,系统自动给出如下的窗体事件过程的框架结构。

```
Private Sub Form_事件名 ([参数列表])           '过程开始  
    [过程体(事件过程代码)]  
End Sub                                           '过程结束
```

几点说明:

- (1) 每个事件过程的开始部分都有保留字 Private 表示该过程是模块级;保留字 Sub 表示这是一个子程序过程。
- (2) 无论窗体的名称(Name)属性值是什么,窗体事件过程名都是“Form_事件名([参数列表])”。控件事件过程名与控件的名称(Name)属性值有关,即控件事件过程名为“控件名_事件名([参数列表])”。
- (3) 事件过程中的[参数列表]由 Visual Basic 系统根据具体事件自动提供,用户无需

关心。

3. 窗体的常用事件

Visual Basic 应用程序是事件驱动,要学好 Visual Basic 编程,必须知道对象有什么事件,事件何时发生,事件发生的次序如何。窗体的常用事件如表 3-4 所示。

表 3-4 窗体的常用事件

事 件 名	说 明	事 件 名	说 明
Initialize	初始化事件	Click	单击事件
Load	装载事件	DblClick	双击事件
QueryUnload	卸载前触发事件	GotFocus	获得焦点事件
Unload	卸载时触发事件	LostFocus	失去焦点事件

(1) Initialize 事件和 Load 事件。当窗体运行时,窗体的 Initialize 事件比 Load 事件先触发。Initialize 事件是在窗体进行初始化时触发;Load 事件是在窗体读入内存时触发。

(2) QueryUnload 事件和 Unload 事件。在关闭窗体或卸载应用程序时,QueryUnload 事件比 Unload 事件先触发。

(3) Click 事件和 DblClick 事件。若鼠标单击或双击窗体的空白处,则触发窗体的该组事件。若鼠标单击或双击窗体上的控件,则触发相应控件的单击事件(Click)或双击事件(DblClick)。

注意: 鼠标双击实际上依次触发单击事件(Click)和双击事件(DblClick)两个事件。

(4) GotFocus 事件和 LostFocus 事件。当窗体获得焦点时,触发 GotFocus 事件;当窗体失去焦点时,触发 LostFocus 事件。能够接收焦点的对象都具有这组事件。

例 3-2 焦点事件示例(焦点的概念见第 3.8 节)。

对于焦点事件的应用,向窗体添加两个文本框,文本框获得焦点表现为光标在文本框内闪烁。具体创建 Visual Basic 应用程序和上机操作步骤如下。

步骤 1,应用程序界面设计和属性设置。启动 Visual Basic 6.0,自动创建一个窗体,可以利用鼠标移动窗体位置或改变窗体的大小。通过工具箱向窗体添加两个文本框和一个命令按钮,利用鼠标调整它们的位置和大小。

打开属性窗口,选定对象的相应属性进行属性设置,各对象主要属性设置如表 3-5 所示。界面设计如图 3-6 所示。

表 3-5 各对象主要属性设置

对 象	属 性 名	属 性 值	对 象	属 性 名	属 性 值
窗体	名称(Name)	Form	文本框 1	名称(Name)	Text1
	Caption	焦点事件示例		Text	(清空)
命令按钮	名称(Name)	Command1	文本框 2	名称(Name)	Text2
	Caption	退出		Text	(清空)

步骤 2,编写相关事件代码。打开代码窗口,依次选定对象命令按钮 Command1 的单击事件 Click、对象文本框 1 的获得焦点事件 GotFocus 和失去焦点事件 LostFocus、对象文本框 2 的获得焦点事件 GotFocus 和失去焦点事件 LostFocus。在相应事件过程框架中输入编辑过程代码。如图 3-7 所示。



图 3-6 焦点事件示例中的界面设计



图 3-7 焦点事件示例中的事件过程代码

步骤 3,调试、运行应用程序。Visual Basic 应用程序设计编辑完成后,按 F5 键或选择“运行”|“启动”菜单命令或单击“启动”图标 ,应用程序开始运行。文本框 1 获得焦点,此时,文本框 1 的内容是“文本框 1 获得焦点!”并出现闪烁的光标,文本框 2 的内容为空;按 Tab 键,使文本框 2 获得焦点,文本框 1 失去焦点,此时,文本框 1 的内容是“文本框 1 失去焦点!”,文本框 2 的内容是“文本框 2 获得焦点!”并出现闪烁的光标;再按一次 Tab 键,使文本框 1 获得焦点,文本框 2 失去焦点,此时,文本框 1 的内容是“文本框 1 获得焦点!”并出现闪烁的光标,文本框 2 的内容是“文本框 2 失去焦点!”。应用程序运行结果界面如图 3-8 所示。单击“退出”按钮,结束应用程序的运行,返回 Visual Basic 集成开发环境的设计模式。

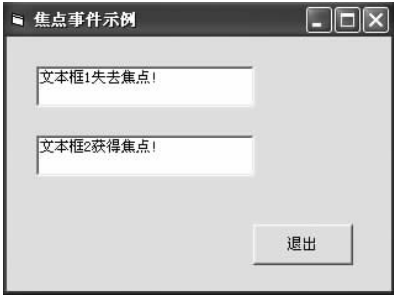


图 3-8 焦点事件示例中的应用程序运行界面

3.3 文本控件

在 Visual Basic 中可以使用的控件分为标准控件和 ActiveX 控件两类。Visual Basic 工具箱中的 20 个控件属于标准控件,其中标签控件和文本框控件统称为文本控件。图 3-9 所示为标签和文本框的外观。

3.3.1 标签

标签控件  (Label)主要用于在窗体上标注和显示提示信息。标签在程序运行时不



图 3-9 标签和文本框的外观

能接收焦点,即不能输入数据文本;但可以通过赋值语句更改标签标题(Caption)的属性值。标签的默认名称为 Label1、Label2、……

1. 常用属性

标签的常用属性如表 3-6 所示。

表 3-6 标签的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	标签的名称	Alignment	标题内容的对齐方式
Caption	标签标题	AutoSize	根据标题内容自动调节大小
BackStyle	标签的背景样式	WordWrap	标题内容的显示方式
BorderStyle	标签的边框样式	ForeColor	标签的文本前景色
Font	标签中文本的字体格式	BackColor	标签的文本背景色

(1) Caption 属性。该属性是标签的重要属性,用于显示文本信息。

(2) Alignment 属性。该属性用于指定在标签上标题内容显示的对齐方式,有以下 3 种取值。

- ① 当属性值为 0(默认值)时,表示标题内容左对齐显示。
- ② 当属性值为 1 时,表示标题内容右对齐显示。
- ③ 当属性值为 2 时,表示标题内容居居中显示。

(3) AutoSize 属性。该属性用于设置标签框的自动调节大小功能,有以下两种取值。

- ① 当属性值为 True 时,表示标签框根据标题内容自动调节大小。
- ② 当属性值为 False(默认值)时,表示标题内容超出标签框部分不显示。

(4) WordWrap 属性。该属性用于设置标签框垂直方向自动调节大小功能,有以下两种取值。

① 当属性值为 True 时,表示标签框在垂直方向根据标题内容自动调节大小;水平方向的大小不改变。

② 当属性值为 False(默认值)时,表示标签框不会改变垂直方向的大小以适应标题内容的显示需要;水平方向的大小变化取决于 AutoSize 属性值的设置。

注意:要使 WordWrap 属性的设置生效,必须将 AutoSize 属性值设置为 True。设置标签的字体属性(Font)和颜色属性(ForeColor/ BackColor),可以使显示的文本内容更清晰、更美观。

2. 方法和事件

标签有刷新(Refresh)、移动(Move)等方法;有单击(Click)、双击(DblClick)等事件。标签的方法和事件不经常使用。

3.3.2 文本框

文本框控件  (TextBox) 又称为编辑框,是最常用的基本控件之一。它可以输入编辑文本,常常作为程序的输入数据控件;它可以容纳大量文本,常常作为程序输出数据控件。文本框事实上是一个小型的全屏幕文本编辑器,它可以有许多变化以适应不同任务的需要。文本框的默认名称为 Text1、Text2、……

1. 常用属性

文本框的常用属性如表 3-7 所示。

表 3-7 文本框的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	文本框的名称	SelStart	选定的文本起始位置
Text	文本框中显示的文本	SelLength	选定的文本字符长度
Locked	文本框是否锁定	BackStyle	文本框的背景样式
PasswordChar	密码显示	BorderStyle	文本框的边框样式
MaxLength	文本框中可输入的字符数	Font	文本框中文本的字体格式
MultiLine	文本框是否多行显示	ForeColor	文本框的文本前景色
ScrollBars	文本框是否有滚动条	BackColor	文本框的文本背景色
SelText	选定的文本内容		

(1) Text 属性。该属性是文本框的最重要属性之一。程序运行时,在文本框中输入、编辑和显示的文本内容都保存在 Text 属性中。通过该属性可以方便地实现数据的输入和数据的输出。

(2) Locked 属性。该属性用于指定文本框中的文本是否可以被编辑,有以下两种取值。

- ①当属性值为 True 时,表示文本框中的文本是只读的,不能被修改。
- ②当属性值为 False(默认值)时,表示文本框中的文本可读可写,可以被编辑。

(3) PasswordChar 属性。该属性用于设置密码显示。属性默认值为空串。若将该

属性设置为某个字符,比如星号(*),则程序运行时,在文本框输入的每个字符均以该字符显示,以隐藏输入的密码或口令。

(4) ScrollBars 属性。该属性用于设置文本框中是否显示滚动条。有以下 4 种取值。

①当属性值为 0(默认值)时,表示无滚动条。

②当属性值为 1 时,表示有水平滚动条。

③当属性值为 2 时,表示有垂直滚动条。

④当属性值为 3 时,表示既有水平滚动条又有垂直滚动条。

注意: 利用 ScrollBars 属性设置滚动条,只有当 MultiLine 属性值为 True 时才有效。

(5) SelText 属性、SelStart 属性和 SelLength 属性。SelText 属性用于获得在文本框中选定的文本内容。SelStart 属性用于获得和设置在文本框中选定的文本的第 1 个字符位置;若没有文本被选定,则给出焦点位置。SelLength 属性用于获得和设置在文本框中选定的文本长度。

在 Visual Basic 运行模式下,通过拖动鼠标可以在文本框中选定一段文本,选定的文本呈反相显示,该组属性就返回所选文本的相关信息,以便操作处理。这组属性在属性窗口中没有列出,只能在程序代码中设置和使用。

2. 方法和事件

文本框最常用的方法是 SetFocus 方法,它使文本框获得焦点,表现为光标在文本框内闪烁,以便进行数据输入。Refresh 方法用于进行文本框刷新。

文本框最重要的事件是 Change 事件,每当文本框的 Text 属性值发生变化时,都将触发 Change 事件。在 Change 事件过程中常用于检验输入的内容。在 Change 事件过程中不要做改变 Text 属性值的事,否则又将触发该事件,以至于导致死循环。

例 3-3 模拟剪贴板功能。

在窗体中添加两个文本框,文本框 1 作为数据源,文本框 2 作为目标处;添加 3 个命令按钮,命令按钮 1 作为复制按钮,命令按钮 2 作为剪切按钮,命令按钮 3 作为粘贴按钮。要求应用程序实现以下功能。

(1) 在文本框 1 中选定文本后,单击“复制”或“剪切”按钮,使“粘贴”按钮从无效状态变为有效状态。

(2) 当文本框 2 获得焦点后,单击“粘贴”按钮,实现从文本框 1 到文本框 2 的选定文本的复制或剪切操作。

问题分析: 实现题目要求,需要使用文本框的 SelText 属性、SelStart 属性和 SelLength 属性。即实现复制或剪切操作,首先利用这 3 个属性记录文本框 1 中选定文本的相关内容。若进行复制操作,则只需将选定的文本直接在文本框 2 中显示;若进行剪切操作,则除了将选定的文本在文本框 2 中显示外,还要将文本框 1 中选定的文本删除。另外,由于在不同的事件过程中要进行数据的传递,因此需要定义一个模块级变量 temptext。Visual Basic 应用程序设计与上机操作步骤如下。

步骤 1,应用程序界面设计和属性设置。启动 Visual Basic 6.0,自动创建一个窗体,利用鼠标移动窗体位置或改变窗体大小。通过工具箱向窗体添加两个文本框和 3 个命令按钮,利用鼠标调整它们的位置和大小。

打开属性窗口,选定对象的相应属性进行属性设置,各对象主要属性设置如表 3-8 所示。

表 3-8 各对象主要属性设置

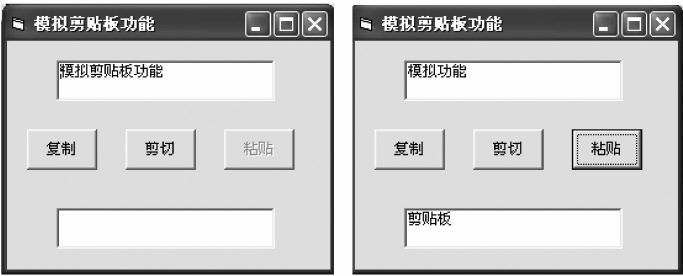
对 象	属 性 名	属 性 值	对 象	属 性 名	属 性 值
窗体	名称(Name)	Form	命令按钮 1	名称(Name)	CmdCopy
	Caption	模拟剪贴板功能		Caption	复制
文本框 1	名称(Name)	Text1	命令按钮 2	名称(Name)	CmdCut
	Text	模拟剪贴板功能		Caption	剪切
文本框 2	名称(Name)	Text2	命令按钮 3	名称(Name)	CmdPaste
	Text	(清空)		Caption	粘贴
				Enabled	False

步骤 2,编写相关事件代码。打开代码窗口,首先定义模块级变量;然后依次选定对象命令按钮 CmdCopy、CmdCut 和 CmdPaste 的单击事件 Click,在相应的事件过程框架结构中输入编辑过程代码,如图 3-10 所示。



图 3-10 模拟剪贴板功能中的事件过程代码

步骤 3,调试、运行应用程序。Visual Basic 应用程序设计编辑完成后,按 F5 键或选择“运行”|“启动”菜单命令或单击“启动”图标 ,应用程序开始运行。程序运行初始界面如图 3-11(a)所示。选定文本框 1 中的文本“剪贴板”,单击“剪切”按钮,按 Tab 键,使文本框 2 获得焦点,单击“粘贴”按钮,得到如图 3-11(b)所示的应用程序运行结果。单击“关闭”按钮 ,结束应用程序的运行,返回 Visual Basic 集成开发环境的设计模式。



(a) 程序运行初始界面 (b) 对选定文本剪切后的界面

图 3-11 模拟剪贴板功能的程序运行界面

3.4 命令按钮

命令按钮(CommandButton)  是窗体上最常见的控件之一。Visual Basic 应用程序界面中基本上都有命令按钮,它是用户与程序进行交互的最常用控件。命令按钮的默认名称为 Command1、Command2、……

1. 常用属性

命令按钮的常用属性如表 3-9 所示。

表 3-9 命令按钮的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	命令按钮的名称	Picture	设置图形按钮的图片
Caption	命令按钮中显示的文本	Enabled	命令按钮是否有效
Style	设置命令按钮样式	BackColor	命令按钮的背景色
ToolTipText	设置命令按钮的提示文本	Font	设置字体
Default	设置默认命令按钮	Visible	命令按钮是否可见
Cancel	设置取消命令按钮		

(1) Caption 属性。该属性用于设置命令按钮上的显示标题。一般 Caption 属性值设置为能反映该命令按钮功能的简要文本说明。

在 Caption 属性值设置中可以创建命令按钮的快捷键。所谓快捷键是指命令标题中某字符带有下划线,如“文件(F)”,只要按 Alt+F 键即可执行该命令。在 Visual Basic 6.0 中,设置快捷键的方法比较简单,只要在 Caption 属性值设置中的相应字符前加上符号“&”即可。例如,在窗体添加命令按钮,将命令按钮的 Caption 属性设置为“退出(&Q)”,这样,就为“退出”命令设置好了快捷键 Alt+Q,窗体运行效果如图 3-12 所示。



图 3-12 为命令按钮设置快捷键

(2) Style 属性。该属性用于设置命令按钮的样式,有以下两种取值。

① 当属性值为 0(默认值)时,表示命令按钮为标准样式,即命令按钮中只显示文本内容。

② 当属性值为 1 时,表示命令按钮为图形样式,命令按钮中不仅可以显示文本,还可以通过 Picture 属性设置显示图片。

(3) ToolTipText 属性。该属性用于设置当鼠标在命令按钮上停留时显示的文本,起提示作用。

(4) Default 属性。该属性用于设置使窗体中的某一命令按钮为默认按钮。有以下两种取值。

① 当属性值为 True 时,表示将该命令按钮设置为默认按钮,窗体上其他命令按钮的 Default 属性值自动设置为 False。程序运行时,只要拥有焦点的控件不是任何一个命令按钮,按 Enter 键,就触发该命令按钮的单击事件(Click)。

② 属性值 False 为命令按钮的默认值。

(5) Cancel 属性。该属性用于设置使窗体中的某一命令按钮为取消按钮。有以下两种取值。

① 当属性值为 True 时,表示将该命令按钮设置为取消按钮,窗体上其他命令按钮的 Cancel 属性值自动设置为 False。程序运行时,只要按 Esc 键,就触发该命令按钮的单击事件(Click)。

② 属性值 False 为命令按钮的默认值。

2. 方法和事件

调用 SetFocus 方法可以使对象获得焦点,命令按钮获得焦点的表现是按钮表面有一个虚线框。

单击事件(Click)是命令按钮最基本、最常用的事件之一。在应用程序运行的窗口界面中,常常通过激活命令按钮触发单击事件来完成相应的操作。

在应用程序运行时,有多种方法可以激活命令按钮。

(1) 用鼠标直接单击命令按钮。

(2) 用 Tab 键或调用 SetFocus 方法使命令按钮获得焦点,按 Enter 键。

(3) 当命令按钮具有快捷键功能时,使用快捷键(Alt+带有下划线的字母)。

(4) 当命令按钮的 Default 属性值为 True 时,按 Enter 键;或当命令按钮的 Cancel 属性值为 True 时,按 Esc 键。

(5) 在事件过程代码中设置命令按钮的 Value 属性值为 True。

3.5 单选按钮、复选框和框架

在许多应用程序中,为了更好地体现交互性和便利性,经常对一些受限的内容进行选择操作。Visual Basic 6.0 提供了单选按钮、复选框和框架,以便程序员组织友好的窗口界面。单选按钮、复选框和框架的外观如图 3-13 所示。

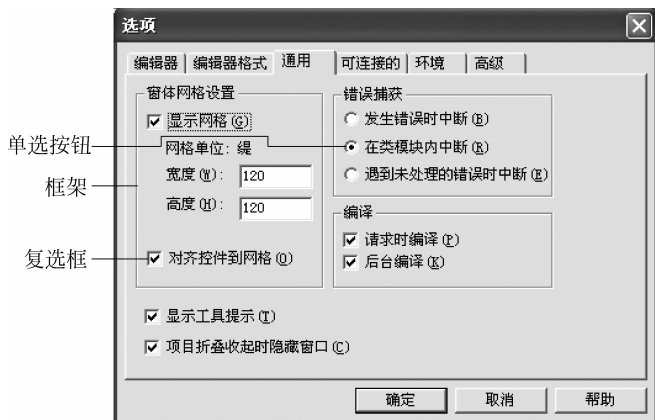


图 3-13 单选按钮、复选框和框架的外观

3.5.1 单选按钮

单选按钮(OptionButton)  用于在一组选项中选且仅选一个选项。单选按钮一般是成组出现,每次只能在这组选项中选定一个。单选按钮的默认名称为 Option1、Option2、……

1. 常用属性

单选按钮的常用属性如表 3-10 所示。

表 3-10 单选按钮的常用属性

属性名	说明	属性名	说明
名称(Name)	单选按钮的名称	Style	设置单选按钮样式
Caption	单选按钮的标题	Enabled	单选按钮是否有效
Alignment	标题对齐方式	Visible	单选按钮是否可见
Value	单选按钮的被选状态	Font	设置字体

(1) Alignment 属性。该属性用于设置单选按钮标题的对齐方式,如图 3-14 所示,有以下两种取值。

① 当属性值为 0(默认值)时,表示左对齐,标题在控件对象的右侧。

② 当属性值为 1 时,表示右对齐,标题在控件对象的左侧。



图 3-14 单击按钮标题对齐方式

(2) Value 属性。该属性用于设置单选按钮的被选状态,有以下两种取值。

① 当属性值为 True 时,表示该按钮被选定,按钮状态为 .

② 当属性值为 False(默认值)时,表示该按钮没被选定,按钮状态为 .

在一组单选按钮中,只有一个单选按钮的 Value 属性值为 True。应用程序运行时,可以通过用鼠标单击来改变按钮的状态。

2. 方法和事件

调用 SetFocus 方法可以使单选按钮被选定,使 Value 属性值为 True。
单选按钮支持的事件不多,应用较多的基本事件是 Click 事件。

3.5.2 复选框

复选框(CheckBox)用于在一组选项中可以选定多个选项。复选框的默认名称为 Check1、Check2、……

1. 常用属性

复选框的常用属性如表 3-11 所示。

表 3-11 复选框的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	复选框的名称	Style	设置复选框样式
Caption	复选框的标题	Enabled	复选框是否有效
Alignment	标题对齐方式	Visible	复选框是否可见
Value	复选框的状态	Font	设置字体

Value 属性是复选框最基本的属性之一,用于设置复选框的状态,有以下 3 种取值。

- ① 当属性值为 0(默认值)时,表示取消该复选框,复选框状态为 。
- ② 当属性值为 1 时,表示选定该复选框,复选框状态为 。
- ③ 当属性值为 2 时,表示该复选框呈灰色不可选状态,复选框状态为 。

2. 方法和事件

调用 SetFocus 方法可以使复选框获得焦点,但不能改变复选框的 Value 属性值。
复选框支持的事件不多,应用较多的基本事件是 Click 事件。

3.5.3 框架

框架 (Frame)是个容器对象,常用于将其他控件按功能分组,既实现了窗口界面的功能分组,又保证了窗口界面的整齐美观。框架的默认名称为 Frame 1、Frame 2、……

注意：在窗口界面利用框架对控件进行功能分组时,一定要先在窗体中添加框架,然后在框架内添加控件。这样框架及其内部的控件就成为一个整体,随框架容器一起移动、显示和隐藏等。

1. 常用属性

框架的常用属性如表 3-12 所示。

(1) Caption 属性。该属性用于设置框架的标题,标题文本位于框架的左上角。若标题内容为空,则框架为封闭的矩形框。

(2) Enabled 属性。该属性用于确定框架容器是否有效,有以下两种取值。

- ① 当属性值为 True(默认值)时,表示该框架及其内部控件都有效。

表 3-12 框架的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	框架的名称	Enabled	框架是否有效
Caption	框架的标题	Visible	框架是否可见

② 当属性值为 False 时,表示框架容器无效,此时,框架标题变为灰色,内部控件全被屏蔽。

(3) Visible 属性。该属性用于确定框架容器是否可见,有以下两种取值。

① 当属性值为 True(默认值)时,表示该框架及其内部控件都可见。

② 当属性值为 False 时,表示框架容器不可见,即程序运行时,框架及其内部控件全不可见。

例 3-4 框架、单选按钮和复选框应用示例。

在窗体中添加 1 个标签;1 个文本框;1 个命令按钮;2 个框架,其中,一个框架中添加两个单选按钮,另一个框架中添加 4 个复选框。各对象属性设置如表 3-13 所示。

表 3-13 各对象主要属性设置

对 象	属 性 名	属 性 值	对 象	属 性 名	属 性 值
窗体	名称(Name)	Form	单选按钮 1	名称(Name)	Option1
	Caption	框架、单选按钮和复选框应用示例		Caption	本科生
标签	名称(Name)	Label1	单选按钮 2	名称(Name)	Option2
	Text	选择你的学历和所学课程		Caption	研究生
	Font	小四楷体-GB2312	复选框 1	名称(Name)	Check1
文本框	名称(Name)	Text1		Caption	管理学
	Text	(清空)	复选框 2	名称(Name)	Check2
	Multiline	True		Caption	经济学
框架 1	名称(Name)	Frame 1	复选框 3	名称(Name)	Check3
	Caption	学历		Caption	程序设计基础
框架 2	名称(Name)	Frame 2	复选框 4	名称(Name)	Check4
	Caption	课程		Caption	数据库基础
命令按钮	名称(Name)	Command1	命令按钮	名称(Name)	Command1
	Caption	显示		Caption	显示

对命令按钮(Command1)的单击事件(Click)编写过程代码如图 3-15 所示。运行应用程序得到如图 3-16 所示窗口界面。

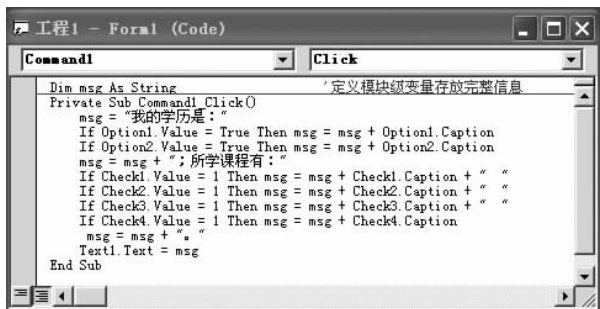


图 3-15 Click 事件过程代码



图 3-16 应用程序运行的窗口界面

3.6 列表框和组合框

列表框和组合框都可以提供一个已知选项的列表清单,供用户选择,两者之间有许多相似的地方。

3.6.1 列表框

列表框(ListBox)  用于显示数据项列表,用户可以通过鼠标单击或按空格键选择所需要的数据项(列表项)。列表框可以单列显示数据项,也可以多列显示数据项。当数据项超出列表框时,会自动添加滚动条。列表框的默认名称为 List1、List2、……

1. 常用属性

列表框的常用属性如表 3-14 所示。

表 3-14 列表框的常用属性

属 性 名	说 明	属 性 名	说 明
名称(Name)	列表框的名称	Style	设置列表框样式
List	列表框数据项内容	Columns	列表框列数
ListCount	列表框数据项数目	MultiSelect	设置列表框的单选或多选方式
ListIndex	被选定数据项索引号	SelCount	多选的数目
Text	被选定数据项内容	Selected	数据项是否被选定

(1) List 属性。该属性是列表框的最重要属性之一,用于记录列表框中的所有数据项,它是一个字符串数组。

在设计模式下,可以通过属性窗口直接输入数据项,每输入一项数据,按 Ctrl+Enter 键实现换行,接着输入下一项数据,当输完最后一项数据,按 Enter 键关闭编辑区,如图 3-17 所示。

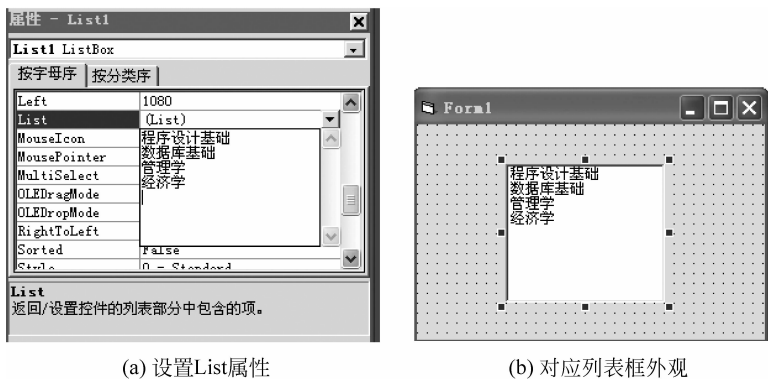


图 3-17 设置 List 属性及对应列表框外观

在过程代码中,可以使用下面赋值语句将列表项值赋给变量(比如 strv1);也可以通过变量(比如 strv2)设置列表项值。

```
strv1=[对象名.] List(下标)           '将列表项值赋给变量
[对象名.] List(下标)= strv2          '通过变量值设置列表项值
```

其中,下标是列表项所处的位置编号,即索引号,索引号从 0 开始。例如图 3-17 中列表框的列表项[对象名.] List(0)的值为“程序设计基础”、[对象名.]List(3)的值为“经济学”。

(2) ListCount 属性。该属性是运行模式下的属性,用于返回当前列表框中列表项的总数目。

(3) ListIndex 属性和 Text 属性。这两个属性是运行模式下的属性。

ListIndex 属性用于返回最后一次被选定的列表项的索引号。显然,List(ListIndex)为所选定的列表项。若未选定任何列表项,则 ListIndex 属性值为-1。

Text 属性用于返回最后一次被选定的列表项。显然,

```
[对象名.] Text=[对象名.] List(ListIndex)
```

(4) Style 属性。该属性用于设置列表框的外观,有以下两种取值。

① 当属性值为 0(默认值)时,表示列表框为标准形式。

② 当属性值为 1 时,表示列表框为复选框形式。如图 3-18 所示。



图 3-18 列表框的多列形式、标准形式和复选框形式

(5) Columns 属性。该属性用于设置列表框的列数,只能在属性窗口设置。当属性值为 0(默认值)时,表示列表框呈单列显示;当属性值为 1 或大于 1 时,表示列表框呈多列显示。如图 3-18 所示。

(6) MultiSelect 属性。该属性用于设置列表框的单选或多选方式,只能在属性窗口设置。有以下 3 种取值。

- ① 当属性值为 0(默认值)时,表示只能进行单项选择。
- ② 当属性值为 1 时,表示可以通过鼠标单击或按空格键来选定或取消多个列表项。
- ③ 当属性值为 2 时,表示可以利用 Shift 键或 Ctrl 键与鼠标单击或按空格键配合进行多个列表项选定。

(7) SelCount 属性。该属性是运行模式下的属性,用于返回在多项选择时所选列表项的数目。

(8) Selected 属性。该属性是运行模式下的属性,用于判断列表项是否被选定。对于确定的索引号 i,若 List(i)被选定,则 Selected(i)的值为 True,否则 Selected(i)的值为 False。

例 3-5 将列表框中选定的列表项在窗体上输出。

在窗体中添加 1 个列表框和 1 个命令按钮。各对象属性设置如表 3-15 所示。命令按钮单击事件过程代码如图 3-19 所示。应用程序运行界面如图 3-20 所示。

表 3-15 各对象主要属性设置

对 象	属 性 名	属 性 值	对 象	属 性 名	属 性 值
窗体	名称(Name)	Form	列表框	名称(Name)	List1
命令按钮	名称(Name)	Command1		List(0)-List(3)	(略)
	Caption	输出		MultiSelect	1-Simple

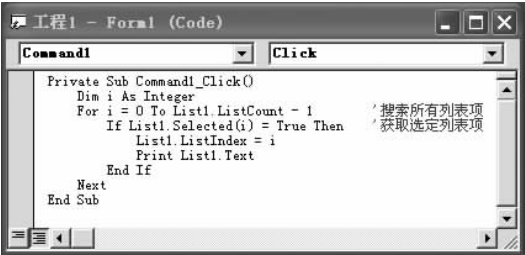


图 3-19 命令按钮单击事件过程代码



图 3-20 应用程序运行界面

2. 方法和事件

列表框支持的常用方法和事件有 Additem 方法、Removeitem 方法、Clear 方法和 Click 事件。

(1) Additem 方法。该方法是列表框最重要的方法之一,用于向列表框中添加列表项。调用格式如下：