

任务 3.5 创建输入框和消息框

在本任务中，将创建一个应用程序。当运行程序时，首先弹出一个输入框提示输入用户名，如图 3.9 所示。在输入用户名并单击“确定”按钮后，在窗体上显示对用户的欢迎信息，如图 3.10 所示。当关闭窗口时将弹出一个消息框，如果单击“确定”按钮，则退出程序；如果单击“取消”按钮，则返回窗口，如图 3.11 所示。

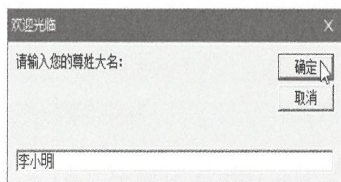


图 3.9 通过输入框输入用户名

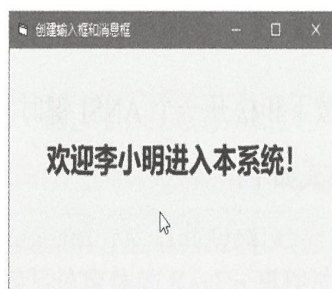


图 3.10 在窗体上显示欢迎信息

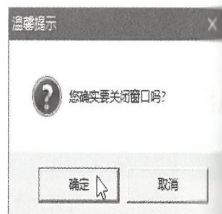


图 3.11 弹出消息框

任务目标

- 掌握使用 InputBox 函数创建输入框的方法。
- 掌握使用 MsgBox 函数创建消息框的方法。
- 掌握窗体的 Activate 事件的应用。

任务分析

如果想要实现这个任务的目标，则需要在程序窗体被激活以前通过 Load 事件过程调用 InputBox 函数来获取用户输入的信息；在得到用户信息后，可以使用窗体的 Activate 事件在窗体激活时对该用户显示欢迎信息。

任务实施

- (1) 在 Visual Basic 6.0 集成开发环境中创建一个标准 EXE 工程。
- (2) 把窗体 Form1 调整到所需大小, 将其 Caption 属性值设置为“创建输入框和消息框”。
- (3) 在窗体 Form1 的代码窗口中声明一个窗体级变量, 代码如下:

```
Private Username As String
```

- (4) 在窗体 Form1 的代码窗口中编写该窗体的 Activate 事件过程, 代码如下:

```
'当激活窗口时执行以下事件过程
Private Sub Form_Activate()
    Print: Print
    Print Tab(4); "欢迎" & Username & "进入本系统!"
End Sub
```

- (5) 在窗体 Form1 的代码窗口中编写该窗体的 Load 事件过程, 代码如下:

```
'当加载窗体时执行以下事件过程
Private Sub Form_Load()
    Me.FontSize = 20
    '将 InputBox 函数的返回值赋给变量 Username
    Username = InputBox("请输入您的尊姓大名: ", "欢迎光临")
End Sub
```

- (6) 在窗体 Form1 的代码窗口中编写该窗体的 Unload 事件过程, 代码如下:

```
'当卸载窗体时执行以下事件过程
Private Sub Form_Unload(Cancel As Integer)
    Dim Choice As Integer
    Choice = MsgBox("您确实要关闭窗口吗?", vbQuestion + vbOKCancel, "温馨提示")
    If Choice = vbCancel Then
        Cancel = True    '不卸载窗体
    End If
End Sub
```

- (7) 将窗体文件保存为 Form03-05.frm, 工程文件保存为工程 03-05.vbp。

程序测试

- (1) 按下 F5 键, 以运行程序。
- (2) 当出现消息框提示“请输入您的尊姓大名:”时, 输入用户名并单击“确定”按钮。
- (3) 此时出现程序主画面, 并在窗体上显示“欢迎 XXX 进入本系统!”。
- (4) 当单击窗口右上角的关闭按钮时, 将弹出一个消息框, 如果单击“确定”按钮则结束程序, 如果单击“取消”按钮则继续程序的运行。

相关知识

1. 窗体的 Activate 事件

当一个窗体成为活动窗体时发生窗体的 Activate 事件，语法格式如下：

```
Private 窗体名_Activate()
```

一个对象通过用户单击，或使用代码中的 Show 和 SetFocus 方法等操作而变成活动的。Activate 事件仅当一个对象可见时才发生。

窗体的 Load 事件和 Activate 事件的区别。当程序载入 (Load) 一个窗体的 Visible 属性值为 False 的窗体时不产生 Activate 事件，只有当使用 Show 方法将该窗体的 Visible 属性值设置为 True 时才产生 Activate 事件。Load 事件在 Activate 事件之前发生，Load 事件在看不到窗体时就已经发生了，一般是对窗体进行初始化；而 Activate 事件是看到窗体时（当前窗体）才发生。

2. 用户对话框

在设计基于 Windows 的应用程序时，对话框用来提示用户应用程序继续运行所需的数据或向用户显示信息。对话框是一种特殊类型的窗体对象。创建对话框有以下 3 种方法。

- 使用 MsgBox 或 InputBox 函数的代码可以创建预定义对话框。
- 使用标准窗体或自定义已经存在的对话框创建自定义对话框。
- 使用 CommonDialog 控件可以创建标准对话框，如“打印”对话框和“打开文件”对话框等。

由于大多数对话框需要用户的交互作用，因此通常显示为模态对话框。在继续使用应用程序的其他部分以前，必须关闭（隐藏或卸载）模态对话框。例如，若在切换到其他窗体或其他对话框前必须单击“确定”或“取消”按钮，则这个对话框就是模态对话框。

非模态对话框不需要关闭就可以使焦点在该对话框和其他窗体之间移动。当对话框显示时，可以在当前应用程序的其他地方继续工作。非模态对话框很少；通常因为应用程序继续前需要响应才显示对话框。Visual Basic 中“编辑”下拉菜单中的“查找”对话框就是一个非模态对话框。使用非模态对话框显示常用的命令或信息。在任务 3.5 中介绍的是创建对话框的第一种方式。

3. MsgBox 函数

MsgBox 函数用来在对话框中显示消息，等待用户单击按钮，并返回一个整数表明用户单击了哪一个按钮，语法格式如下：

```
MsgBox(prompt[, buttons][, title][, helpfile, context])
```

其中，参数 prompt 是必需的，它是一个字符串表达式，用于指定显示在对话框中的消息。该参数的最大长度为 1024 个字符，由所用字符的宽度决定。如果该参数的内容超过一行，则可以在各行之间使用回车符 (Chr(13))、换行符 (Chr(10)) 或回车符与换行符的组合 (Chr(13))

及 Chr(10)) 将各行分隔开。

参数 buttons 是可选的, 该参数为数值表达式, 是一些数值的总和, 用于指定显示的按钮的数目及形式、使用的图标样式、默认按钮及消息框的强制回应等。如果省略参数 buttons, 则其默认值为 0。参数 buttons 的设置值如表 3.3 所示。

表 3.3 参数 buttons 的设置值

符号常量	数 值	描 述
vbOKOnly	0	只显示“确定”按钮
vbOKCancel	1	显示“确定”和“取消”按钮
vbAbortRetryIgnore	2	显示“终止”、“重试”和“忽略”按钮
vbYesNoCancel	3	显示“是”、“否”和“取消”按钮
vbYesNo	4	显示“是”和“否”按钮
vbRetryCancel	5	显示“重试”和“取消”按钮
vbCritical	16	显示 Critical Message 图标
vbQuestion	32	显示 Warning Query 图标
vbExclamation	48	显示 Warning Message 图标
vbInformation	64	显示 Information Message 图标
vbDefaultButton1	0	第一个按钮是默认按钮
vbDefaultButton2	256	第二个按钮是默认按钮
vbDefaultButton3	512	第三个按钮是默认按钮

参数 title 是可选的, 用于指定在对话框标题栏中显示的字符串表达式。如果省略参数 title, 则将应用程序名放入对话框标题栏中。

参数 helpfile 是可选的, 该参数为字符串表达式, 用来识别向对话框提供上下文相关帮助的帮助文件。如果提供了参数 helpfile, 则必须提供参数 context。

参数 context 是可选的, 该参数为数值表达式, 由帮助文件的作者指定给某个帮助主题的帮助上下文编号。如果提供了参数 context, 则必须提供参数 helpfile。

当提供参数 helpfile 与参数 context 时, 用户可以按下 F1 键来查看与参数 context 相应的帮助主题, 此时会在对话框中添加一个“帮助”按钮。

MsgBox 函数的返回值如表 3.4 所示。

表 3.4 MsgBox 函数的返回值

常 数	数 值	返回的按钮
vbOK	1	OK (“确定”)
vbCancel	2	Cancel (“取消”)
vbAbort	3	Abort (“放弃”)
vbRetry	4	Retry (“重试”)
vbIgnore	5	Ignore (“忽略”)
vbYes	6	Yes (“是”)
vbNo	7	No (“否”)

如果对话框显示“取消”按钮, 则按下 Esc 键的效果与单击“取消”按钮的效果相同。如

果对话框中有“帮助”按钮，则对话框中提供与上下文相关的帮助。但是，直到其他按钮中有一个被单击之前，都不会返回任何值。

如果还要指定第一个命名参数以外的参数，则必须在表达式中使用 `MsgBox` 函数。为了省略某些位置参数，必须加入相应的逗号分隔符。

4. `InputBox` 函数

`InputBox` 函数用来在一个对话框中显示提示，等待用户输入文本或按下按钮，并返回包含文本框内容的字符串，语法格式如下：

```
InputBox(prompt[,title][,default][,xpos][,ypos][,helpfile,context])
```

其中，参数 `prompt` 是必需的，用于指定作为对话框消息出现的字符串表达式。参数 `prompt` 的最大长度是 1024 个字符，由所用字符的宽度决定。如果参数 `prompt` 包含多个行，则可以在各行之间使用回车符(`Chr(13)`)、换行符(`Chr(10)`)或回车符与换行符的组合(`Chr(13) & Chr(10)`)将各行分隔开。

参数 `title` 是可选的，用于指定显示在对话框标题栏中的字符串表达式。如果省略参数 `title`，则将应用程序名放入对话框标题栏中。

参数 `default` 是可选的，用于指定显示文本框中的字符串表达式，在没有其他输入时作为默认值。如果省略参数 `default`，则文本框为空。

参数 `xpos` 是可选的，该参数为数值表达式，与参数 `ypos` 一起成对出现，用于指定对话框的左边与屏幕左边的水平距离。如果省略参数 `xpos`，则对话框会在水平方向居中。

参数 `ypos` 是可选的，该参数为数值表达式，与参数 `xpos` 一起成对出现，用于指定对话框的上边与屏幕上边的垂直距离。如果省略参数 `ypos`，则对话框被放置在屏幕垂直方向距下边大约三分之一处。

参数 `helpfile` 是可选的，该参数为字符串表达式，用来识别为对话框提供上下文相关帮助的帮助文件。如果已经提供参数 `helpfile`，则必须提供参数 `context`。

参数 `context` 是可选的，该参数为数值表达式，由帮助文件的作者指定给某个帮助主题的帮助上下文编号。如果已经提供了参数 `context`，则必须提供参数 `helpfile`。

如果用户单击“确定”按钮或按下 `Enter` 键，则 `InputBox` 函数返回文本框中的内容。如果用户单击“取消”按钮，则此函数返回一个长度为零的字符串(“”)。

如果还要指定第一个命名参数以外的参数，则必须在表达式中使用 `InputBox` 函数。如果想要省略某些位置参数，则必须加入相应的逗号分隔符。