第 10 章　巧妙应用 Excel 的宏与 VBA

10.1 宏的录制、创建与命名

如何录制宏

具体操作步骤如下。

1 将宏病毒安全级设置为“ 中 ”或“ 低 ”。

2 在菜单栏上选择“ 工具｜宏｜录制新宏 ”命令，弹出“ 录制新宏 ”对话框，如图 10.1 所示。



图 10.1 “ 录制新宏 ”对话框

3 在“ 宏名 ”文本框中，输入需设定的宏的名称。

4 在“ 保存在 ”下拉列表框中，选择需要存放宏的地址。

5 如果要添加有关宏的说明，可以在“ 说明 ”文本框中输入该宏的详细说明。

6 单击“ 确定 ”按钮，弹出“ 停止录制 ”工具栏，如图 10.2 所示。



图 10.2 “ 停止录制 ”工具栏

7 在工作簿中执行需要录制的操作，完毕后，在“ 停止录制 ”工具栏上，单击“ 停止录制 ”按钮即可完成。

如何将已经录制的宏指定给工作簿中添加的窗体按钮

具体操作步骤如下。

1 右击菜单栏，弹出快捷菜单，选择“ 窗体 ”选项，打开“ 窗体 ”工具栏，如图 10.3 所示。



图 10.3 “ 窗体 ”工具栏

2 在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，然后在工作表上绘制一个按钮，此时会弹出“ 指定宏 ”对话框，如图 10.4 所示。



图 10.4 “ 指定宏 ”对话框

3 在列表框中选择以前录制的宏“ Macrol ”，然后单击“ 确定 ”按钮，如图 10.5 所示。

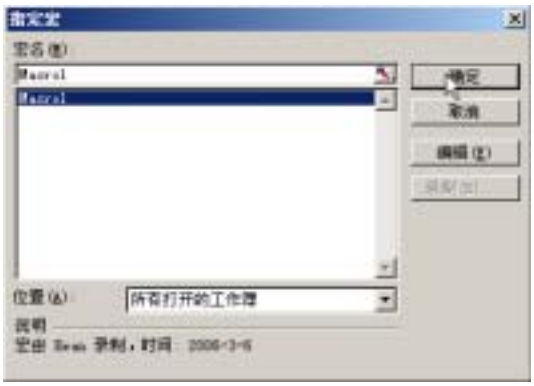


图 10.5 选择宏

4 返回工作表中，如图 10.6 所示，单击此按钮，便可执行选定宏所录制的操作。

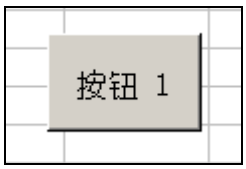


图 10.6 添加的按钮

如何使用 Microsoft Visual Basic 编辑器创建宏

具体操作步骤如下。

1 在 Excel 的菜单栏上选择“ 工具｜宏｜Visual Basic 编辑器 ”命令，弹出 Visual Basic 编辑器窗口，如图 10.7 所示。

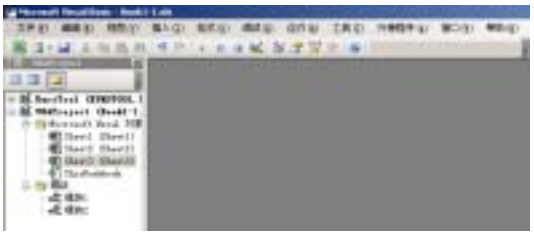


图 10.7 Visual Basic 编辑器窗口

2 在菜单栏上选择 “ 插入｜模块 ” 命令，如图 10.8 所示。

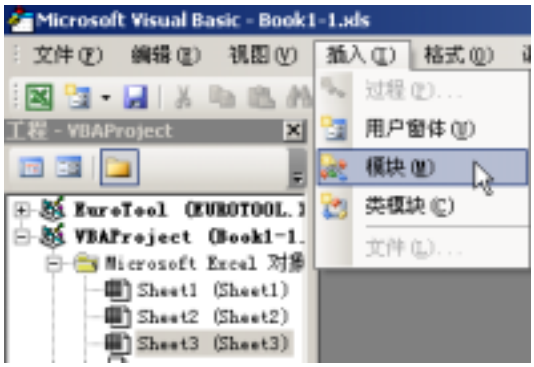


图 10.8 选择 “ 模块 ” 命令

3 将需要设置的代码输入或复制到模块的代码窗口中。

4 如果要模块窗口中运行宏，可以直接按 F5 键。

5 编写完宏后，在菜单栏上选择 “ 文件｜关闭并返回到 Microsoft Excel ” 命令即可关闭此窗口，如图 10.9 所示。

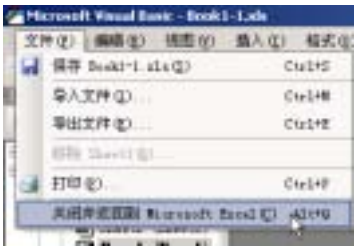


图 10.9 选择 “ 关闭并返回到 Microsoft Excel ” 命令

如何宏命令转化成 VBA 代码

具体操作步骤如下。

1 在菜单栏上选择 “ 工具｜宏｜宏 ” 命令。

2 弹出“ 宏 ”对话框，在列表框中选择“ Macro2 ” 选项，然后单击 “ 编辑 ” 按钮，如图 10.10 所示。



图 10.10 设置 “ 宏 ” 对话框

3 弹出 Visual Basic 编辑器窗口，在代码编辑区内显示的代码便是宏命令的程序代码，如图 10.11 所示。



图 10.11 代码

录制宏的过程中如何设置快捷键

在录制新宏的过程中，会弹出“ 录制新宏 ”对话框，在“ 快捷键 ”文本框中可以输入一个字母，如图 10.12 所示。



图 10.12 输入快捷键

这样，可以直接按 Ctrl+V 键，便可执行宏。

宏录制后如何设置快捷键

具体操作步骤如下。

1 在菜单栏上选择“ 工具｜宏｜宏 ”命令，弹出“ 宏 ”对话框，单击“ 选项 ”按钮，如图 10.13 所示。



图 10.13 单击“ 选项 ”按钮

2 弹出“ 宏选项 ”对话框，在“ 快捷键 ”文本框中，输入需要设定的快捷键，如图 10.14 所示。



图 10.14 “ 宏选项 ”对话框

3 单击“ 确定 ”按钮，这样，直接按 Ctrl+Shift+S 快捷键即可执行宏。

10.2 VBA 中按钮的使用

如何将录制的宏指定给自定义的工具按钮

具体操作步骤如下。

1 在菜单栏上右击鼠标，弹出快捷菜单栏，选择“ 自定义 ”命令，弹出“ 自定义 ”对话框，单击“ 命令 ”标签，切换至“ 命令 ”选项卡下，在“ 类别 ”列表框中选择“ 宏 ”选项，如图 10.15 所示。

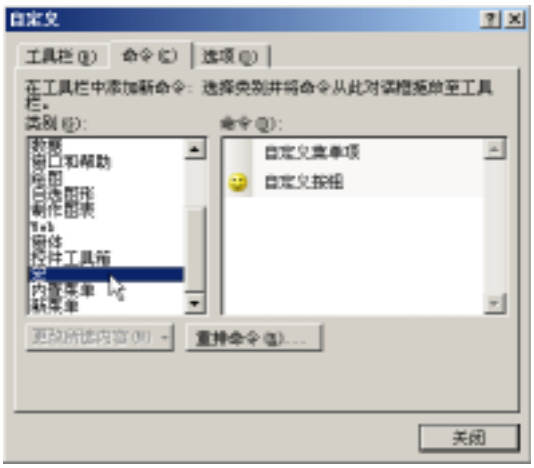


图 10.15 “ 自定义 ” 对话框

2 在右侧的 “ 命令 ” 列表框中，将 “ 自定义按钮 ” 选项拖动到工具栏上，如图 10.16 所示，然后单击 “ 关闭 ” 按钮。

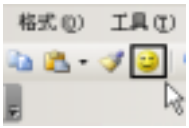


图 10.16 添加按钮

3 单击自定义的按钮，弹出 “ 指定宏 ” 对话框，在列表框中选择需要指定的宏选项，如图 10.17 所示。

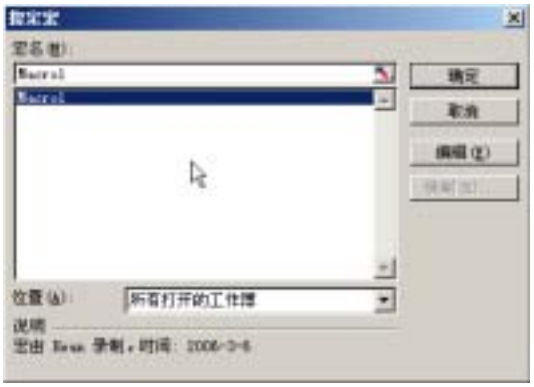


图 10.17 选择宏

5 单击 “ 确定 ” 按钮，这样，以后在菜单栏上单击自定义的按钮即可快速执行宏。

如何更改自定义的工具按钮图标以及名称

具体操作步骤如下。

1 在菜单栏上选择 “ 工具 | 自定义 ” 命令，弹出 “ 自定义 ” 对话框，单击 “ 命令 ” 标签，切换至 “ 命令 ” 选项卡下，在 “ 类别 ” 列表框中选择 “ 宏 ” 选项，然后右击工具栏上的自定义按钮，弹出快捷菜单，选择 “ 更改按钮图像 ” 选项，弹出 “ 图像 ” 列表，选择需要的图标，如图 10.18 所示。

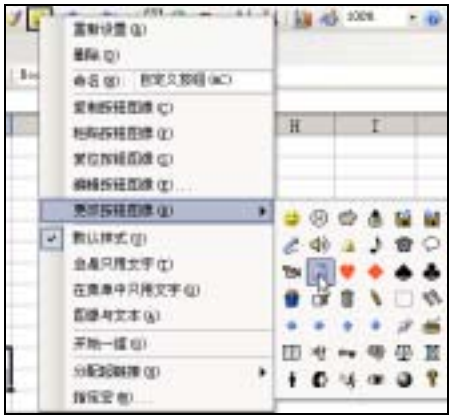


图 10.18 更改图标

2 此时工具栏上的自定义的图标样式已经更换，再右击它，弹出快捷菜单，选择 “ 命名 ” 项，然后在其右侧的文本框中输入 “ 我的按钮 ”，如图 10.19 所示。



图 10.19 输入名称

3 在 “ 自定义 ” 对话框中，单击 “ 关闭 ” 按钮，这样，以后将鼠标指针指向自定义按钮就会出现按钮名称提示，如图 10.20 所示。

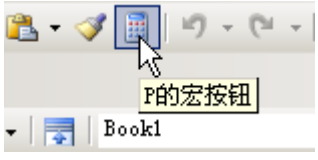


图 10.20 显示名称

如何删除自定义的工具按钮

具体操作步骤如下。

1 在菜单栏上选择“ 工具｜自定义 ”命令，弹出“ 自定义 ”对话框，单击“ 命令 ”标签，切换至“ 命令 ”选项卡下，在“ 类别 ”列表框中选择“ 宏 ”选项。

2 在工具栏上单击自定义的工具按钮，将它拖动至“ 自定义 ”对话框中的“ 命令 ”选项卡的“ 命令 ”列表框中的“ 自定义按钮 ”选项上，如图 10.21 所示。

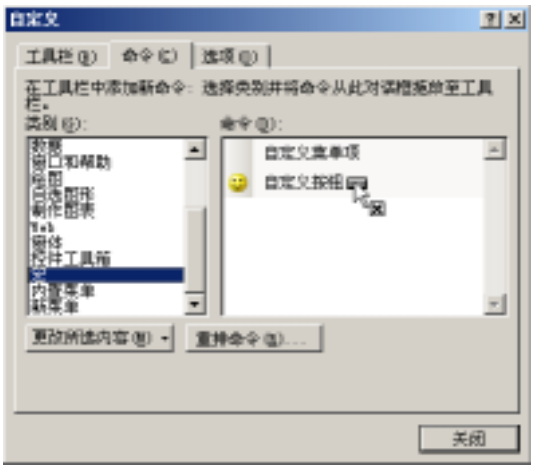


图 10.21 拖动按钮

3 释放鼠标，自定义工具按钮就从工具栏上删除，单击“ 关闭 ”按钮即可完成。

如何复制已创建的宏中的一部分创建另一个宏

具体操作步骤如下。

1 将宏病毒安全级设置为“ 中 ”或“ 低 ”级。

2 打开要复制的宏所在的工作簿文件。

3 在菜单栏上选择“ 工具｜宏｜宏 ”命令。

4 弹出“ 宏 ”对话框，在“ 宏名 ”文本框中输入要复制的宏的名称，然后单击“ 编辑 ”按钮，如图 10.22 所示。

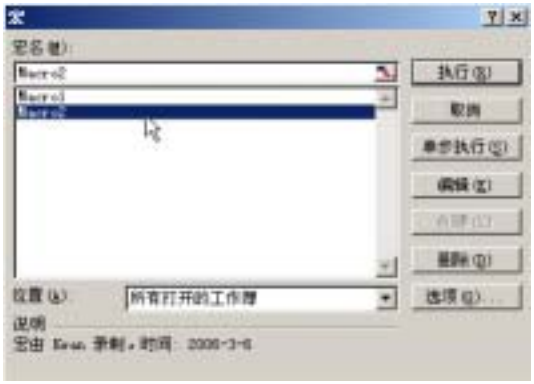


图 10.22 输入宏名

5 弹出 Visual Basic 编辑器，在窗口中的代码编辑区内，选取要复制的程序行，如果要复制整个宏，要在选定区域中包括了“ With ”和“ End With ”行，如图 10.23 所示。



图 10.23 选取代码

6 在 Visual Basic 编辑器窗口中，在菜单栏上选择“ 编辑｜复制 ”命令，如图 10.24 所示。



图 10.24 选择“ 复制 ”命令

7 切换到要放置代码的模块中，然后在工具栏上单击“ 粘贴 ”按钮即可。

操作工作表时如何快捷打开和关闭网格线

具体操作如下。

在菜单栏上选择“ 视图｜工具栏｜窗体 ”命令，弹出“ 窗体 ”工具栏，如图 10.25 所示，“ 切换网格 ”按钮呈打开状态，如果需要关闭风格线，单击此按钮即可完成。



图 10.25 “ 窗体 ”工具栏

如何使用 VBA 进行累加计算

具体操作步骤如下。

1 在工作表中创建如图 10.26 所示的表格。

2 调出“ 窗体 ”工具栏，然后在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，如图 10.27 所示。



图 10.26 创建表格

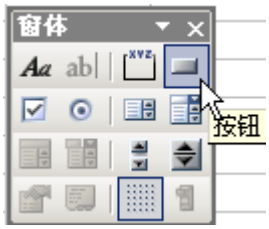


图 10.27 单击“ 按钮 ”按钮

3 在刚创建的表格的下方单击并拖出一个按钮，此时会自动弹出“ 指定宏 ”对话框，单击“ 新建 ”按钮，如图 10.28 所示。

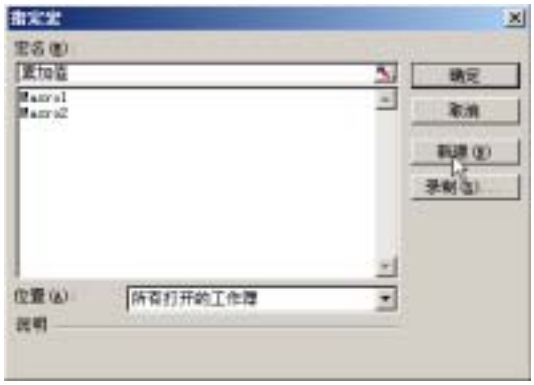


图 10.28 单击“ 新建 ”按钮

4 切换至 Visual Basic 编辑器窗口中，在代码编辑区内输入相应的代码，如图 10.29 所示。

```
Sub 累加值()
    call leijia
End Sub
```

图 10.29 输入代码

其中的代码如下：

```
Sub 按钮 1_单击 ()
    Call leijia
End Sub
```

5 在下方添加一个程序 leijia()，如图 10.30 所示。

```
Sub 累加值()
    Call leijia
End Sub
Sub leijia()
    Dim a, b, c As Long
    a = Cells(1, 2)
    b = c = 0
    While (b < a)
    b = b + 1
    c = c + b
    Wend
    c = c
    Cells(2, 2) = c
End Sub
```

图 10.30 添加程序

其中的代码如下：

```
Sub leijia()
    Dim a,b,c As Long
    a = Cells(1,2)
    b = c = 0
    While (b＜a)
    b = b + 1
    c = c + b
```

```
        Wend
        c = c
        Cells(2,2) = c
    End Sub
```

6 关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮上的文本更改为“ 计算 ”，如图 10.31 所示。



图 10.31 更改按钮名称

7 在单元格 C3 中输入“ 500 ”，单击“ 计算 ”按钮，在单元格 C4 中便显示出累加值“ 125250 ”，如图 10.32 所示。



图 10.32 计算累加值

如何制作行颜色交叉 Excel 表格

具体操作步骤如下。

1 在菜单栏上选择“ 工具｜宏｜录制新宏 ”命令。

2 弹出“ 录制新宏 ”对话框，在“ 宏名 ”文本框中输入宏名“ 行颜色交叉的 Excel 表格 ”，在“ 快捷键 ”文本框中输入“ e ”，如图 10.33 所示。



图 10.33 设置“ 录制新宏 ”对话框

3 单击“ 确定 ”按钮，返回工作表中，在其中建立边框，然后进行多种颜色的格式化，如图 10.34 所示。

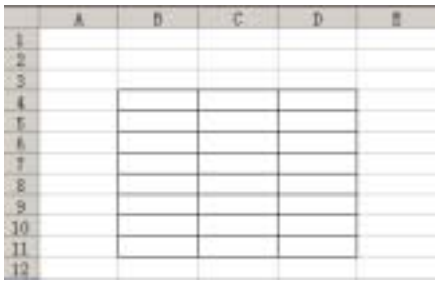


图 10.34 建立表格边框

4 在“ 停止录制 ”工具栏上，单击“ 停止录制 ”按钮，如图 10.35 所示。



图 10.35 单击“ 停止录制 ”按钮

5 返回工作表中并选择一单元格区域，然后按 Ctrl+A 键，在选定的单元格区域内，就会自动建立多色边框表格，如图 10.36 所示。

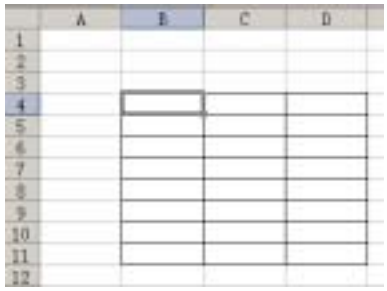


图 10.36 自动建立多色表格

如何使用下拉列表框来控制显示

具体操作步骤如下。

1 调出“ 窗体 ”工具栏，单击“ 窗体 ”工具栏上的“ 组合框 ”按钮，如图 10.37 所示。



图 10.37 单击“ 组合框 ”按钮

2 返回工作表中，单击并拖出一个下拉列表框，如图 10.38 所示。

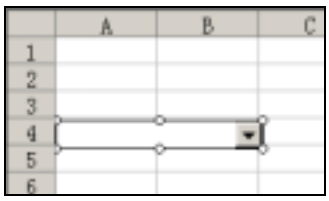


图 10.38 绘制组合框

3 右击下拉列表框，弹出快捷菜单，选择“ 设置控件格式 ”命令。

4 弹出“ 设置控件格式 ”对话框，单击“ 控制 ”标签，切换至“ 控制 ”选项卡下，如图 10.39 所示。



图 10.39 “ 设置控件格式 ”对话框

5 在工作表的其他区域内建立如图 10.40 所示的部门列表。

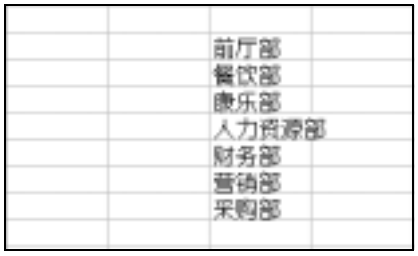


图 10.40 建立部门列表

6 在 “ 设置控件格式 ”对话框中，在“ 控制 ”选项卡下，进行如图 10.41 所示的设置。



图 10.41 设置“ 控制 ”选项卡

7 单击“ 确定 ”按钮，返回工作表中，单击下拉列表按钮，弹出下拉列表，选择“ 财务部 ”选项，如图 10.42 所示。



图 10.42 选择“ 财务部 ”选项

8 此时在单元格 D5 中显示出人力资源部在部门列表中的排序“ 3 ”，如图 10.43 所示。

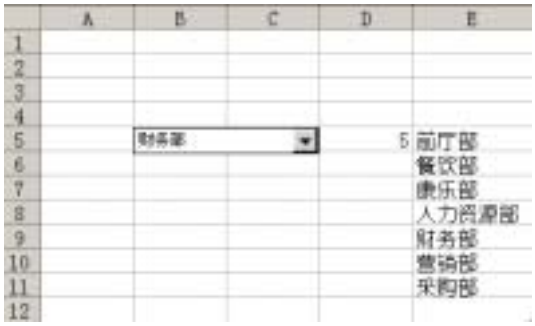


图 10.43 显示排序

如何使用按钮来控制弹出提示对话框的显示

具体操作步骤如下。

1 调出“ 窗体 ”工具栏，单击“ 窗体 ”工具栏

上的“ 按钮 ”，如图 10.44 所示。



图 10.44　单击“ 按钮 ”按钮

2　返回工作表中，单击并拖动出一个按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 弹出提示 ”，然后单击“ 新建 ”按钮，如图 10.45 所示。

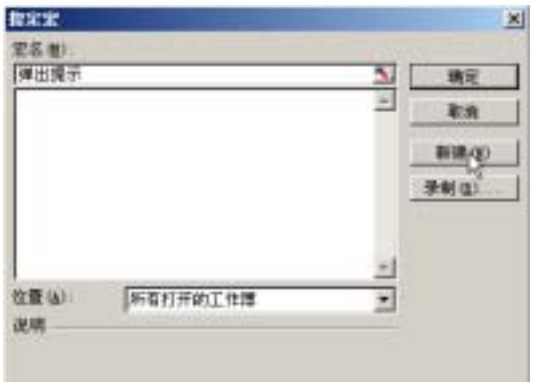


图 10.45　“ 指定宏 ”对话框

3　弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.46 所示。



图 10.46　输入代码

输入的代码如下：

```
Sub 弹出提示()
    ANS = MsgBox ( `请重新输入´ vbYesNo, `输入错误!´ )
    If ANS = vbNo Then Exit Sub
End Sub
```

4　关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮名称更改为“ 输入错误! ”，如下图 10.47 所示。



图 10.47　更改名称

5　单击“ 输入错误! ”按钮，即可弹出对话框。

为何“ 刷新 ”时有消息显示对宏的更改将丢失

在 Visual Basic 编辑器窗口中更改宏时，如果又在 Microsoft 脚本编辑器中更改了工作簿的副本，就会显示此消息。

如果要保存对宏所做的更改，可以执行下列操作。

1　单击“ 否 ”按钮。

2　切换到 Visual Basic 编辑器窗口中。

3　导出已更改的任何模块。

4　在工作表簿的 Microsoft 脚本编辑器窗口中，单击“ 刷新 ”工具栏上的“ 刷新 ”按钮。

5　单击“ 是 ”按钮。

6　导入模块以恢复对宏的更改。

如何使用 InputBox 对话框在工作表中输入信息

具体操作步骤如下。

1　在工作表中建立如图 10.48 所示的表格。

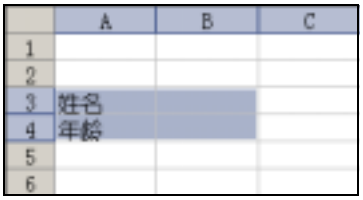


图 10.48　建立表格

2　调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，在建立的表格的右侧单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 开始输入 ”，然后单击“ 新建 ”按钮，如图 10.49 所示。

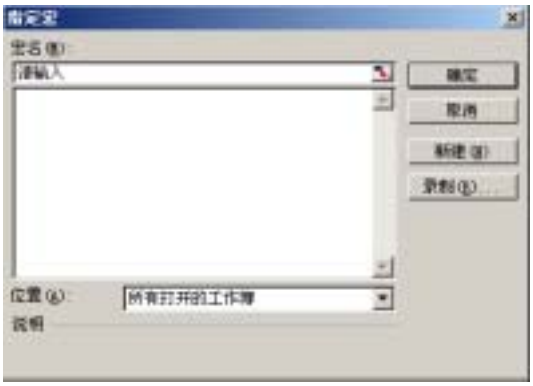


图 10.49 设置“ 指定宏 ”对话框

3 弹出 Visual Basic 编辑窗口，在代码编辑区内输入代码，如图 10.50 所示。



图 10.50 输入代码

输入的代码如下：

```
Sub 请输入( )
 a=InputBox(〝请输入你的姓名：〞)
 Cells(1,2)=a
 If a〈〉 〝〞 Then
 b=InputBox(〝请输入你的年龄：〞)
 Cells(2,2)=b
 End If
End Sub
```

4 关闭 Visual Basic 编辑器窗口，返回工作表中，更改按钮名称为“ 请输入 ”，如图 10.51 所示。



图 10.51 更改“ 按钮 ”名称

5 单击“ 请输入 ”按钮，弹出对话框，提示用户输入姓名，如图 10.52 所示。



图 10.52 输入姓名

6 输入姓名后，单击“ 确定 ”按钮，再次弹出对话框，提示输入年龄，如图 10.53 所示。



图 10.53 输入年龄

7 输入年龄后，单击“ 确定 ”按钮，返回工作表中，此时表格中已经显示出刚才输入的信息，如图 10.54 所示。

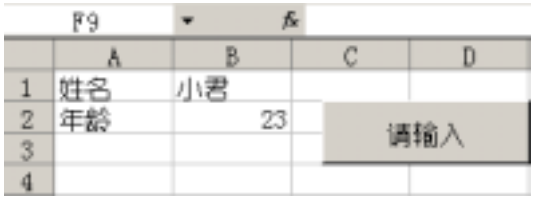


图 10.54 输入信息

如何制作会根据早中晚问候的提示对话框

具体操作步骤如下。

1 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上，单击“ 按钮 ”按钮，在工作表的适当位置单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中，输入“ 温馨提示 ”然后单击“ 新建 ”按钮，如图 10.55 所示。

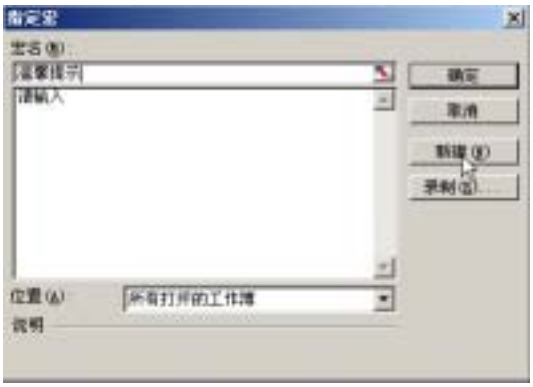


图 10.55　设置“ 指定宏 ”对话框

2　弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.56 所示。



图 10.56　输入代码

输入的代码如下：

```
Sub 温馨提示( )
  Select Case Time
    Case Is＜0.5
      msg=〝小君，早上好！〞
    Case 0.5 To 0.75
      msg=〝小君，下午好！〞
  Case    Else
   msg=〝小君，晚上好！〞
  End Select
  MsgBox msg
  End Sub
```

3　关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮名称更改为“ 温馨提示 ”，如图 10.57 所示。

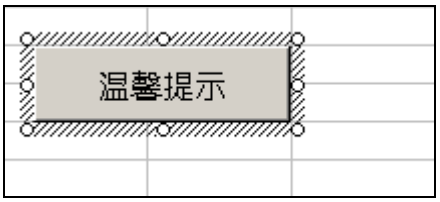


图 10.57　更改名称

4　单击“ 温馨提示 ”按钮，弹出提示对话框，由于当前时间是下午，因此提示信息为“ 小君，下午好！”，如图 10.58 所示。



图 10.58　提示信息

如何显示 Excel 窗口的状态

具体操作步骤如下。

1　调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，在工作表的适当位置单击并拖动出一按钮，在工作表的适当处单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 显示窗口信息 ”，然后单击“ 新建 ”按钮，如图 10.59 所示。

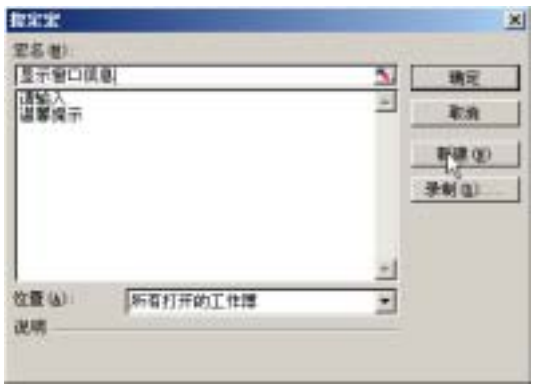


图 10.59　设置“ 指定宏 ”对话框

2　弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.60 所示。



图 10.60 输入代码

其中输入的代码如下：

```
Sub 显示窗口信息()
    Select Case Application.WindowState
        Case xlMaximized: MsgBox "Excel 窗口最大化"
        Case xlMinimized: MsgBox "Excel 窗口工作簿最小化"
        Case xlNormal: MsgBox "Excel 窗口一般大小"
        Select Case ActiveWindow.WindowState
            Case xlMinimized: MsgBox "工作簿最大化"
            Case xlMinimized: MsgBox "工作簿最小化"
            Case xlMinimized: MsgBox "工作簿一般大小"
        End Select
    End Select
End Sub
```

3 关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮名称更改为“ 显示窗口信息 ”，如图 10.61 所示。



图 10.61 更改名称

4 单击“ 显示窗口信息 ”按钮，弹出提示对话框，显示当前 Excel 窗口的信息，如图 10.62 所示。

5 单击“ 确定 ”按钮，弹出工作簿的信息，如图 10.63 所示，单击“ 确定 ”按钮即可完成。



图 10.62 提示 Excel 窗口



图 10.63 提示工作簿窗口

如何快速计算 1~100 的整数的平方根之和

具体操作步骤如下。

1 在工作表中建立如图 10.64 所示的表格。



图 10.64 建立表格

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上，单击“ 按钮 ”按钮，在建立表格的下面单击并拖出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 计算 ”，然后单击“ 新建 ”按钮，如图 10.65 所示。

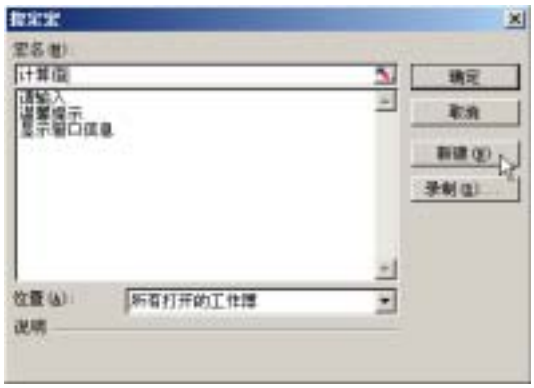


图 10.65 设置“ 指定宏 ”对话框

3 弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.66 所示。



图 10.66 输入代码

输入的代码如下：

```
Sub 计算( )
    Sum = 0
    For Count = 1 To 100
        Sum = Sum + Sqr(Count)
        Next Count
    Cells(3,3) = Sum
    MsgBox Sum
End Sub
```

4 关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮名称更改为“ 计算 ”，然后单击此按钮，弹出计算结果的对话框，如图 10.67 所示。

5 在单元格 C3 中，显示出具体的计算值，如图 10.68 所示。



图 10.67 提示对话框



图 10.68 计算值

如何使用“ 宏 ”对话框来单步执行宏

具体操作步骤如下。

1 在菜单栏上选择“ 工具｜宏 ”命令，弹出“ 宏 ”对话框，在列表框中选择需要进行单步执行的宏名，单击“ 单步执行 ”按钮，如图 10.69 所示。



图 10.69 单击“ 单步执行 ”按钮

2 弹出 Visual Basic 编辑器窗口，在代码编辑区的呈现黄色的行就是正在执行的命令，如图 10.70 所示。



图 10.70 Visual Basic 编辑器窗口

3 单步执行完毕后，可以在菜单栏上选择“ 运行＞运行子过程/用户窗体 ”命令，如图 10.71 所示，来运行程序可行行性。

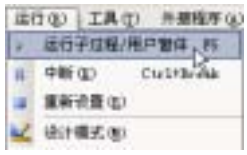


图 10.71 选择“ 运行子程序/用户窗体 ”命令

如何调用其他程序

调用其他程序可以简化代码，代码越简单，就越容易进行修改，同时也容易解码和调试。例如下面的程序代码就调用了“ 温馨提示 ”程序。

```
Sub main( )
        Call 开始输入
        Call 温馨提示
        Call 离开
    End Sub
```

如何添加“ 插入脚本 ”、“ 显示所有脚本 ”等命令

具体操作步骤如下。

1 在菜单栏上选择“ 工具｜自定义 ”命令，弹出“ 自定义 ”对话框，单击“ 命令 ”标签，切换至“ 命令 ”选项卡下，在“ 类别 ”列表框中选择“ 工具 ”选项，如图 10.72 所示。

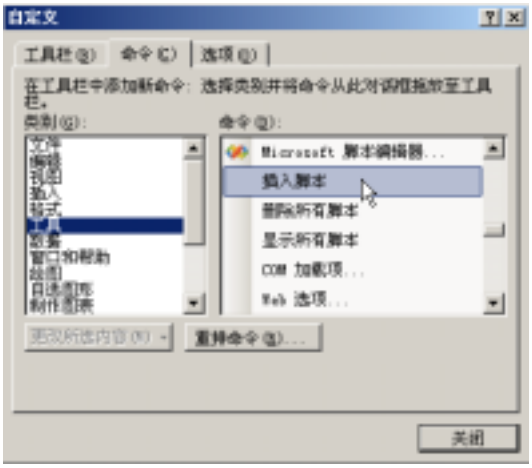


图 10.72 选择“ 工具 ”选项

2 在“ 命令 ”列表框中，将“ 插入脚本 ”选项拖到“ 工具 ”菜单上，当“ 工具 ”菜单显示菜单命令时，将鼠标指针指向“ 宏 ”子菜单，当“ 宏 ”子菜单显示菜单命令时，将鼠标指针指向要放置“ 插入脚本 ”命令的位置，如图 10.73 所示，释放鼠标。



图 10.73 插入选项

3 对“ 显示所有脚本 ”和“ 删除所有脚本 ”命令也采用同样的方法添加。

4 单击“ 自定义 ”对话框中的“ 关闭 ”按钮即可完成。

如何对编写的 VBA 程序进行调试

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中，在菜单栏上选择“ 调试＞逐语句 ”命令。

2 在代码编辑区内程序就会对每个步骤进行调试，如图 10.74 所示。



图 10.74 逐语句调试

如何将宏模块复制到其他工作簿中

具体操作步骤如下。

1 打开包含相应模块的工作簿和需要向其中复制模块的工作簿。

2 在菜单栏上选择“ 工具｜宏|Visual Basic 编辑器 ”命令，弹出 Visual Basic 编辑器窗口。

3 在 Visual Basic 编辑器窗口的菜单栏上，选择“ 视图｜工程资源管理器 ”命令。

4 将需要复制的模块拖到目的工作簿中即可完成。

如何录制一个复制粘贴的宏命令

具体操作步骤如下。

1 在工作表中创建如图 10.75 所示的表格。

	A	B
1	复制区域	
2	粘贴区域	
3		
4		

图 10.75 建立表格

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，然后在建立表格的下侧单击并拖动出一按钮，同时弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 复制 ”，然后单击“ 录制 ”按钮，如图 10.76 所示。

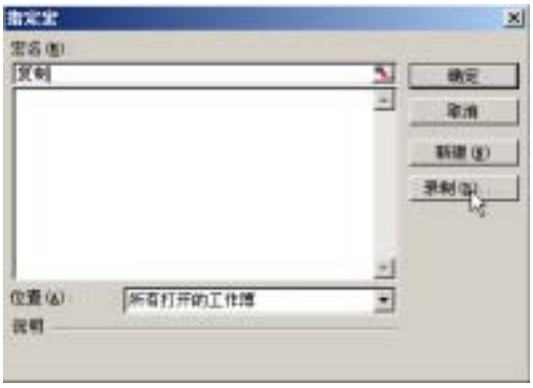


图 10.76 设置“ 指定宏 ”对话框

3 弹出“ 录制新宏 ”对话框，在快捷文本框输入“ E ”，如图 10.77 所示。



图 10.77 设置“ 录制新宏 ”对话框

4 单击“ 确定 ”按钮，返回工作表中，右击单元格 B1，弹出快捷菜单，选择“ 复制 ”命令，右击单元格 B2，弹出快捷菜单，选择“ 粘贴 ”命令，然后单击“ 停止录制 ”工具栏上的“ 停止录制 ”按钮，如图 10.78 所示。



图 10.78 单击“ 停止录制 ”按钮

5 返回工作表中，将按钮名称更改为“ 复制 ”，如图 10.79 所示。



图 10.79 更改按钮名称

6 在复制区域内输入数据“ 天气真好！”，然后单击“ 复制 ”按钮，粘贴区域就会自动显示复制区内的数据，如图 10.80 所示。

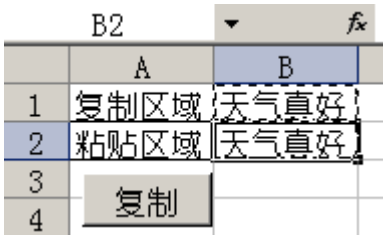


图 10.80 单击“ 复制 ”按钮

如何使用按钮来完成对单元格底纹的填充

具体操作步骤如下。

1 在工作表中建立如图 10.81 所示的表格。

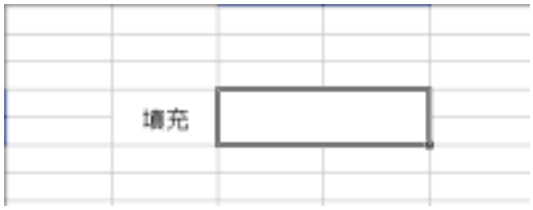


图 10.81 建立表格

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上，单击“ 按钮 ”按钮，然后在建立表格的下侧单击拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 填充 ”，然后单击“ 录制 ”按钮，如图 10.82 所示。

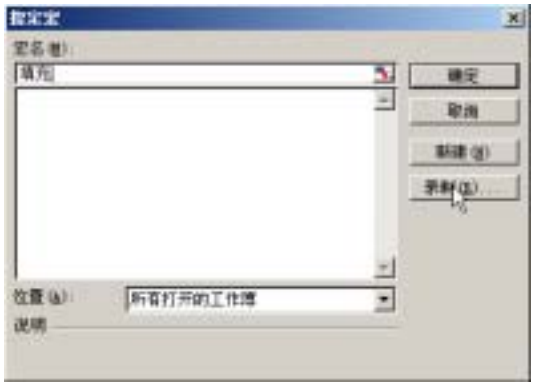


图 10.82 设置“ 指定宏 ”对话框

3 弹出“ 录制新宏 ”对话框，在“ 快捷键 ”文本框中输入“ E ”，如图 10.83 所示。



图 10.83 设置“ 录制新宏 ”对话框

4 单击“ 确定 ”按钮，返回工作表中，对建立的填充表格进行填充设置，单击“ 停止录制 ”工具栏上的“ 停止录制 ”按钮，如图 10.84 所示。



图 10.84 单击“ 停止录制 ”按钮

5 返回工作表中，清除刚才的填充，并将按钮名称更改为“ 填 ”，然后单击“ 填 ”按钮，填充区域便被填充，如图 10.85 所示。

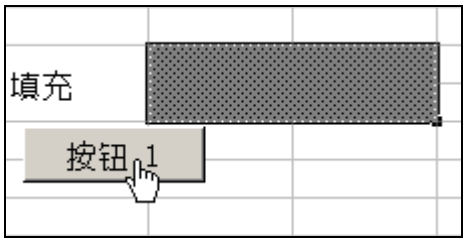


图 10.85 单击“ 填 ”按钮

如何使用按钮来完成清除单元格内容的操作

具体操作步骤如下。

1 在工作表中已经建立的复制表格内进行如图 10.86 所示的设置。

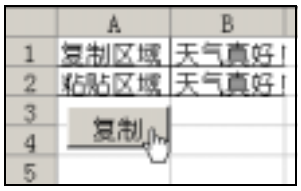


图 10.86 复制区域

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，然后在“ 复制 ”按钮右侧单击并拖动出一按钮，同时弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 清除 ”，然后单击“ 新建 ”按钮，如图 10.87 所示。

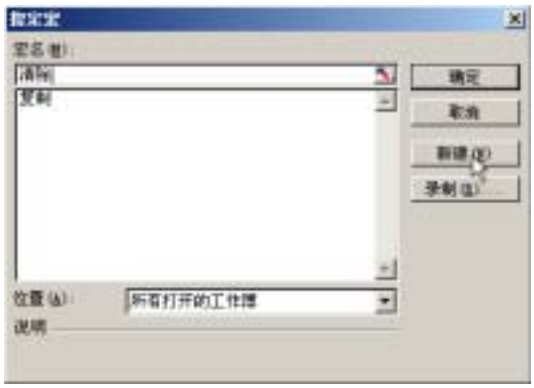


图 10.87 设置“ 指定宏 ”按钮

3 弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.88 所示。

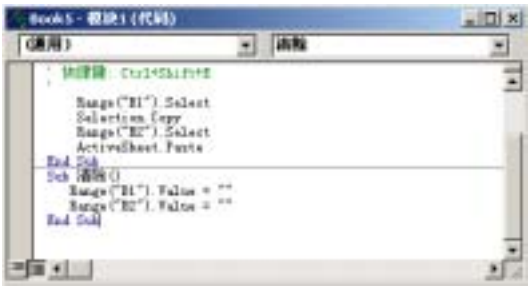


图 10.88 输入代码

输入的代码如下：

```
Sub 清除()
    Range("B1").Value = ""
    Range("B2").Value = ""
End Sub
```

4 关闭“ Visual Basic 编辑器 ”窗口，返回工作表中，将该按钮更名为“ 清除 ”，然后单击此按钮，复制区和粘贴区所有的数据便被清除，如图 10.89、

10.90 所示。

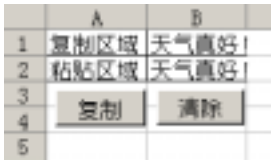


图 10.89 清除按钮



图 10.90 清除数据

如何使用按钮来完成自动求和功能

具体操作步骤如下。

1 在工作表中建立如图 10.91 所示的表格。



图 10.91 建立表格

2 调出“窗体”工具栏，在“窗体”工具栏上单击“按钮”按钮，然后在建立的表格下侧单击并拖动出一按钮，弹出“指定宏”对话框，在“宏名”文本框中输入“求和”，然后单击“新建”按钮，如图 10.92 所示。

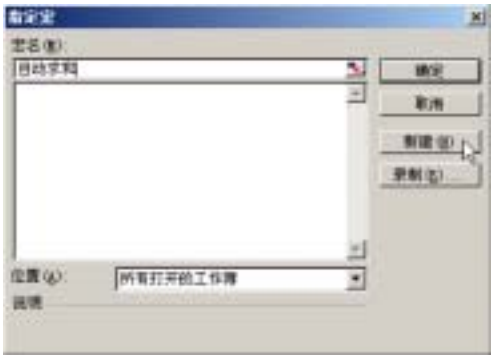


图 10.92 设置“指定宏”对话框

3 弹出“Visual Basic 编辑器”窗口，在代码编辑区内输入代码，如图 10.93 所示。



图 10.93 输入代码

输入的代码如下：

```
Sub 自动求和()
    Dim A As Single
    A = 0
    For i = 3 To 12
    A = A + Cells(i, 3)
    Next
    MsgBox A
End Sub
```

4 关闭 Visual Basic 编辑器窗口，返回工作表中，将该按钮更名为“自动求和”，然后在表格中填入数据，如图 10.94 所示。

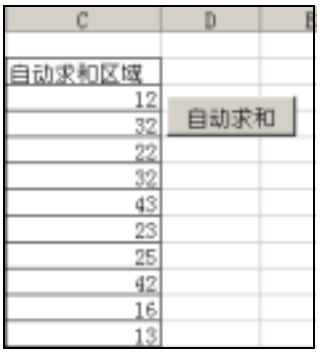


图 10.94 填入数据

5 单击“自动求和”按钮，弹出对话框，提示求和的值，如图 10.95 所示。



图 10.95 提示计算值

如何使用按钮来完成计算平均值功能

具体操作步骤如下。

1 在工作表中创建如图 10.96 所示的表格。

B	C
	数据输入
	12
	32
	22
	32
	43
	23
	25
	42
	16
	13

图 10.96 打开表格

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”按钮，然后在建立的表格下侧单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 计算平均值 ”，然后单击“ 新建 ”按钮，如图 10.97 所示。



图 10.97 单击“ 编辑 ”按钮

3 弹出 Visual Basic 编辑器窗口，在其中的代码编辑区内输入代码，如图 10.98 所示。



图 10.98 输入代码

输入的代码如下：

```
Sub 计算平均值()
    Dim A As Single
    A = 0
    For i = 3 To 12
    A = A + Cells(i, 3)
    Next
    MsgBox A / 10
End Sub
```

4 关闭“ Visual Basic 编辑器 ”窗口，返回工作表中，将该按钮更名为“ 计算 ”，单击此按钮，弹出提示对话框提示计算值，如图 10.99 所示。



图 10.99 提示计算值

如何使用按钮来实现计数功能

具体操作步骤如下。

1 在工作表中某一区域内输入一列数字，如图 10.100 所示。

12
32
22
32
43
23
25
42
16
13

图 10.100 输入数字

2 调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上

单击“ 按钮 ”按钮，然后在输入的数据右侧单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 计数 ”，然后单击“ 新建 ”按钮，如图 10.101 所示。

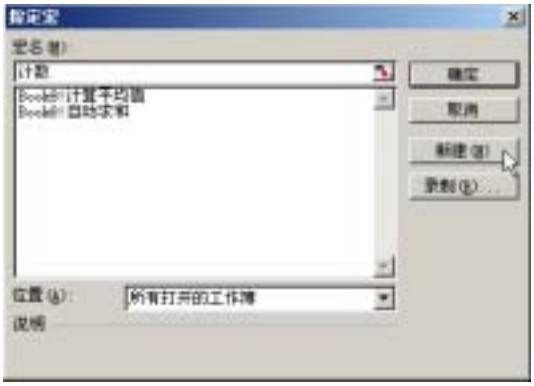


图 10.101　设置“ 指定宏 ”对话框

3　弹出 Visual Basic 编辑器窗口，在其中的代码编辑区内输入代码，如图 10.102 所示。



图 10.102　输入代码

输入的代码如下：

```
Sub 计数()
    Dim a As Single
    a = 0
    For i = 3 To 1000
    If Cells(i, 3) = "" Then
    a = a
    Else
    a = a + 1
    End If
    Next
    MsgBox a
End Sub
```

4　关闭 Visual Basic 编辑器窗口，返回工作表中，将该按钮更名为“ 计数 ”，如图 10.103 所示。

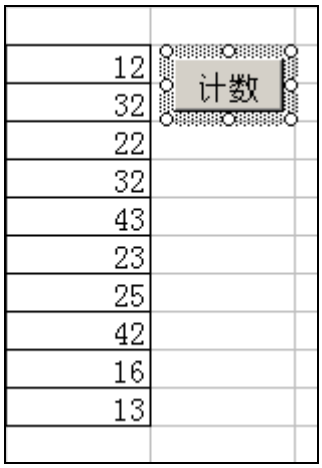


图 10.103　更改按钮名称

5　单击“ 计数 ”按钮，弹出提示对话框，显示出刚输入的数据的个数，如图 10.104 所示。



图 10.104　提示数据个数

如何使用按钮来实现切换工作表功能

具体操作步骤如下。

1　在工作簿中，单击工作表 Sheet1 标签，切换至工作表 Sheet1 中。

2　调出“ 窗体 ”工具栏，在“ 窗体 ”工具栏上单击“ 按钮 ”，在适当处置单击并拖动出一按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 切换工作表 ”，然后单击“ 新建 ”按钮，如图 10.105 所示。

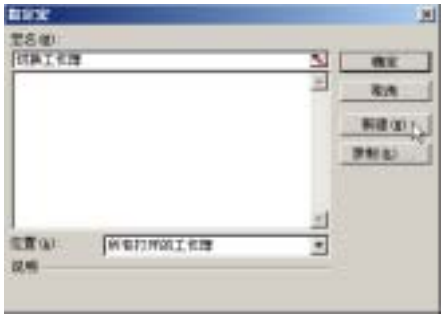


图 10.105　设置“ 指定宏 ”对话框

3 弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.106 所示。



图 10.106 输入代码

输入的代码如下：

```
Sub 切换工作簿()
    Sheet3.Select
End Sub
```

4 关闭 Visual Basic 编辑器窗口，返回工作表中，将该按钮更名为“切换工作表”，单击此按钮，就可以自动切换至工作表 Sheet3 中，如图 10.107 所示。



图 10.107 单击按钮

10.3 VBA 中窗体的使用

如何为 VBA 程序添加窗体

具体操作步骤如下。

1 切换到 Visual Basic 编辑器窗口，如图 10.108 所示。

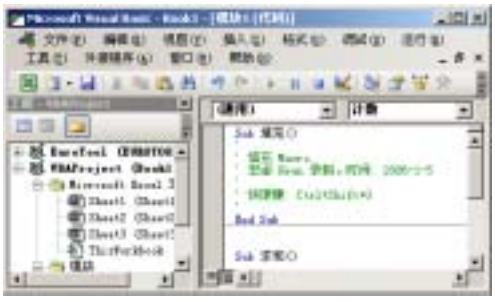


图 10.108 Visual Basic 编辑器窗口

2 在菜单栏上选择“插入｜用户窗体”命令，如图 10.109 所示。



图 10.109 选择“用户窗体”命令

3 在 Visual Basic 编辑器窗口中弹出窗体面板和工具箱，如图 10.110 所示。

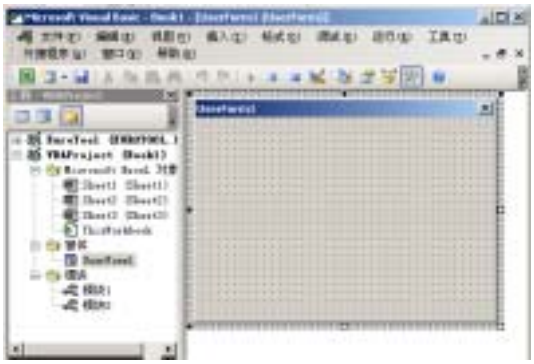


图 10.110 打开面板和工具箱

如何在窗体上添加控件

一般步骤如下。

1 在 Visual Basic 编辑器窗口中，在菜单栏上选择“插入｜用户窗体”命令，显示窗体面板，如图 10.111 所示。

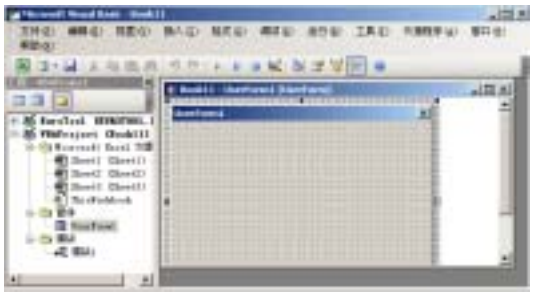


图 10.111 显示窗体面板

2 在菜单栏上选择“视图｜工具箱”命令。调出工具箱，如图 10.112 所示。



图 10.112　工具箱

3 在工具箱上单击“ 标签 ”按钮，返回窗体面板上，在适当位置单击并拖动出标签框，如图 10.113 所示。

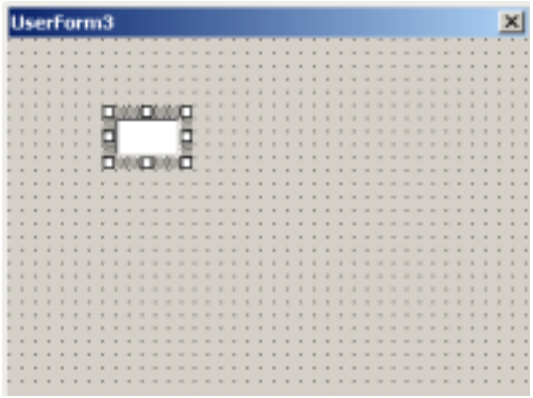


图 10.113　绘制标签框

4 在左侧“ 属性 ”列表框中进行标签的属性设置，如图 10.114 所示。

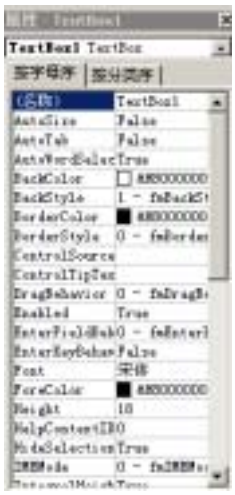


图 10.114　设置属性

如何使用窗体关闭工作簿文件

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中，在菜单栏上选择“ 插入 | 用户窗体 ”命令，显示窗体面板，如图 10.115 所示。

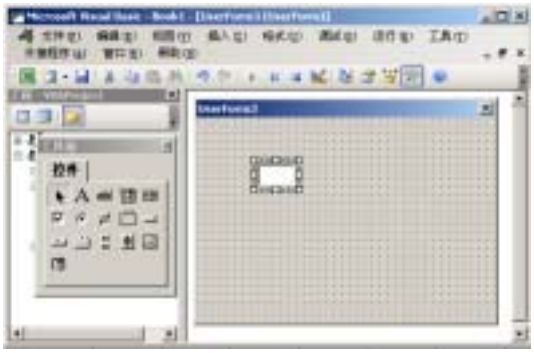


图 10.115　显示窗体

2 在工具箱上单击“ 按钮 ”按钮，然后在窗体中间单击并拖动出一按钮，如图 10.116 所示。

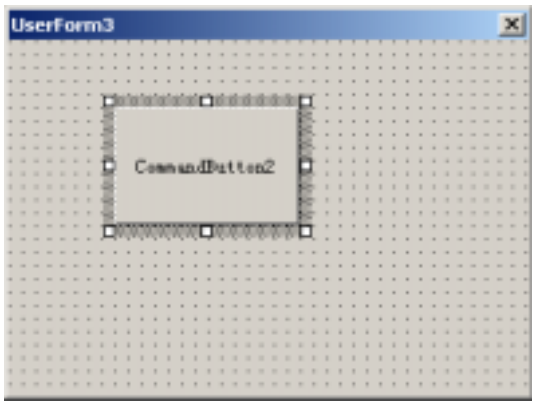


图 10.116　绘制按钮

3 在左侧的“ 属性 ”窗口中，将“ 名称 ”更改为“ 关闭工作簿 ”;将 Caption 更改为“ 关闭工作簿 ”，如图 10.117 所示。



图 10.117 更改属性参数

4 返回窗体上，此时按钮上的名称已经做了相应的更改，然后双击此按钮，弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.118 所示。

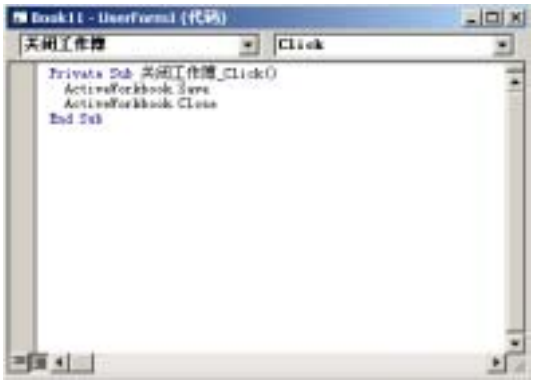


图 10.118 输入代码

输入的代码如下：

```
Private Sub 关闭工作簿_Click( )
    ActiveWorkbook.Save
    ActiveWorkbook.Close
End Sub
```

5 在 Visual Basic 编辑器窗口的菜单栏上，选择“ 运行｜运行子过程/用户窗体 ”命令，如图所示。

6 弹出一个窗体，单击 “ 关闭工作簿 ” 按钮，系统便会保存工作簿文件，然后关闭工作簿，如图 10.119 所示。

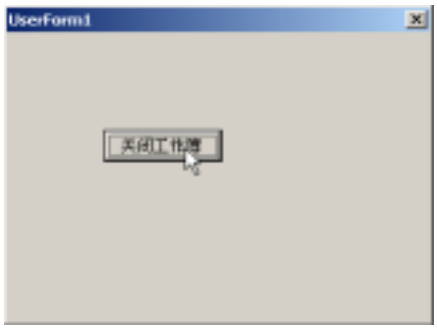


图 10.119 单击按钮

如何使用窗体切换工作簿中的工作表

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择 “ 插入｜用户窗体 ” 命令，显示窗体面板，如图 10.120 所示。

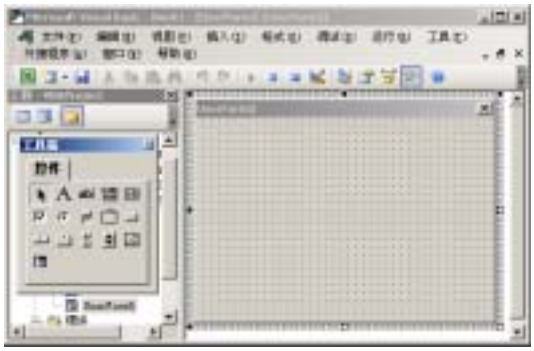


图 10.120 插入窗体

2 在工具箱上单击 “ 按钮 ” 按钮，然后在窗体的适当位置绘制 3 个相同大小的按钮，如图 10.121 所示。

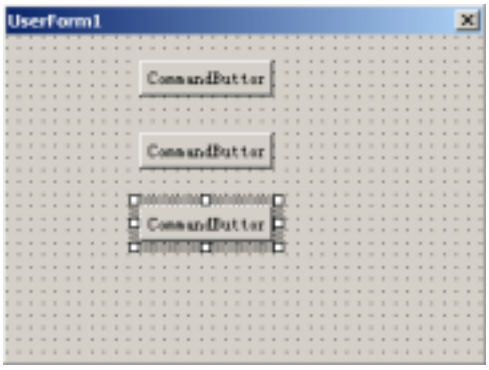


图 10.121 绘制按钮

3 在 “ 属性 ” 列表框中，分别将 3 个按钮的名

称设置为“ 切换至工作表 Sheet1 ”、“ 切换至工作表 Sheet2 ”和“ 切换至工作表 Sheet3 ”；将 3 个按钮的 Cpation 设置为“ 工作表 Sheet1 ”、“ 工作表 Sheet2 ”和“ 工作表 Sheet3 ”。

4 返回窗体上，3 个按钮的名称已经被更改，如图 10.122 所示。

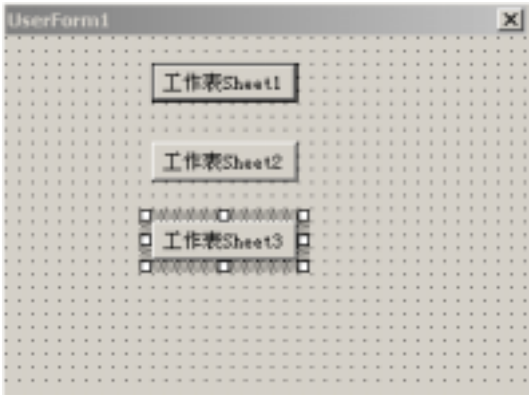


图 10.122 更改名称

5 分别双击 3 个按钮，弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.123 所示。

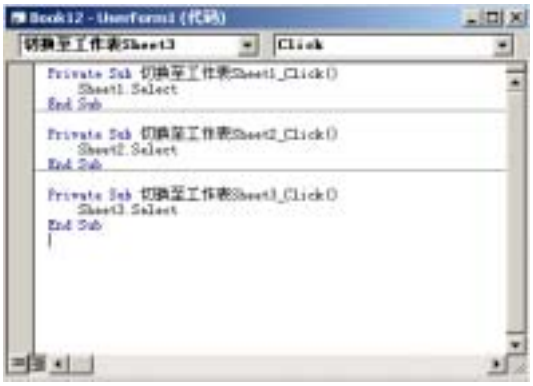


图 10.123 输入代码

输入的代码如下：

```
Private Sub 切换至工作表 Sheet1_Click( )
    Sheet1.Select
End Sub
Private Sub 切换至工作表 Sheet2_ Click( )
    Sheet2.Select
End Sub
Private Sub 切换至工作表 Sheet3_ Click( )
    Sheet3.Select
End Sub
```

6 在工具栏上单击“ 运行子过程/用户窗体 ”按钮，如图 10.124 所示。



图 10.124 单击“ 运行子过程/用户窗体 ”按钮

7 弹出含有 3 个按钮的对话框窗体，单击任意一个按钮，系统就可以切换至相应的工作表中，如图 10.125 所示。

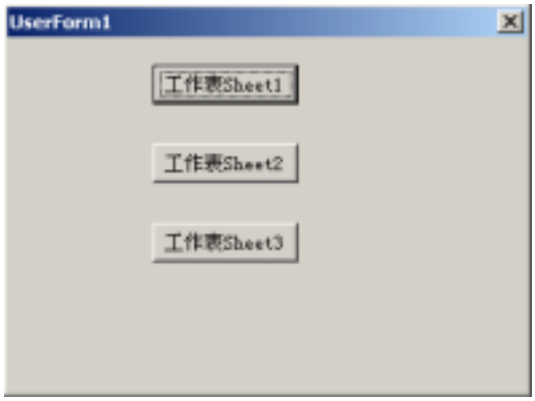


图 10.125 单击按钮

如何在工作表中使用按钮显示窗体

具体操作步骤如下。

1 在工作表中，调出“ 窗体 ”工具栏，单击“ 按钮 ”按钮，如图 10.126 所示。



图 10.126 单击“ 按钮 ”按钮

2 在工作表的适当位置单击并拖动出一个按钮，弹出“ 指定宏 ”对话框，在“ 宏名 ”文本框中输入“ 显示窗体 ”，然后单击“ 新建 ”按钮，如图 10.127 所示。

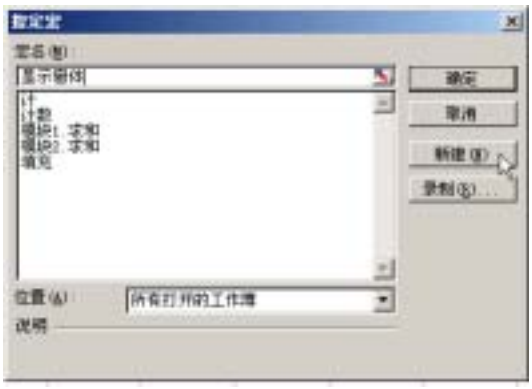


图 10.127 设置“ 指定宏 ”对话框

3 弹出 Visual Basic 编辑器窗口，在代码编辑区内输入代码，如图 10.128 所示。



图 10.128 输入代码

其中输入的代码如下：

```
Sub 显示窗体( )
    UserForm2.Show
End Sub
```

4 关闭 Visual Basic 编辑器窗口，返回工作表中，将按钮名称更改为“ 显示窗体 ”，如图 10.129 所示。



图 10.129 更改名称

5 单击“ 显示窗体 ”按钮，弹出已设定的窗体，如图 10.130 所示。



图 10.130 显示的窗体

如何隐藏不用的窗体

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中打开需要隐藏的窗体，如图 10.131 所示。

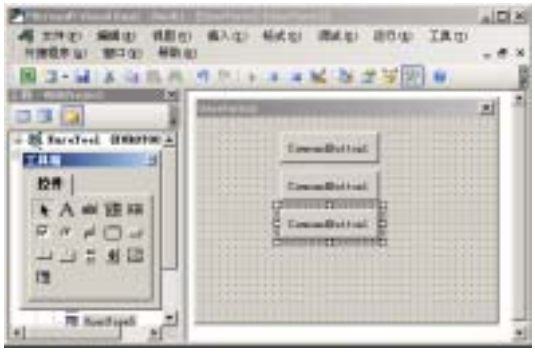


图 10.131 打开窗体

2 分别双击窗体上的 3 个按钮，在切换的代码编辑区内进行代码的修改，如图 10.132 所示。

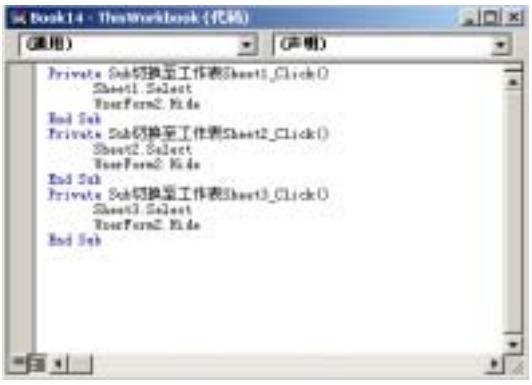


图 10.132 编辑后的代码

输入的代码如下：

```
    Private Sub 切换至工作表 Sheet1_Click()
Sheet1.Select
        UserForm2.Hide
    End Sub
    Private Sub 切换至工作表 Sheet2_Click()
Sheet2.Select
        UserForm2.Hide
    End Sub
    Private Sub 切换至工作表 Sheet3_Click()
        Sheet3.Select
        UserForm2.Hide
    End Sub
```

3 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，如图 10.133 所示。



图 10.133 选择“ 运行|运行子过程/用户窗体 ”命令

4 弹出对话框，单击任一按钮，切换至相应的工作表中后，对话框也立即隐藏起来。

如何使用窗体在工作表中输入数据

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，如图 10.134 所示。

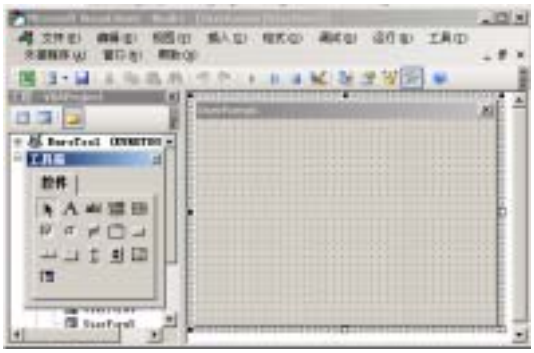


图 10.134 显示插入窗体

2 在工具箱上单击“ 文字框 ”按钮，如图 10.135 所示。



图 10.135 单击“ 文字框 ”按钮

3 在窗体上的适当位置单击并拖动出一文字框区域，如图 10.136 所示。

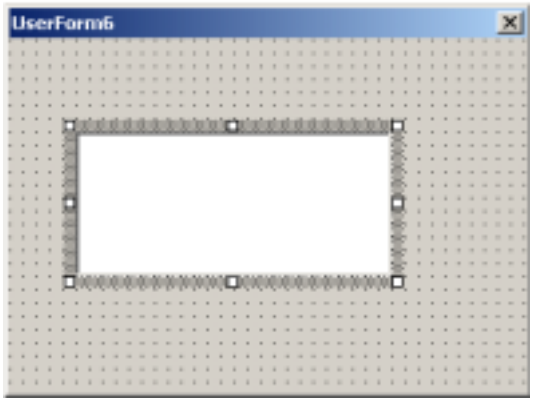


图 10.136 拖动出文字框区域

4 在窗体上添加一按钮，将“ 名称 ”更改为“ 输入 ”，将 Caption 更改为“ 输入 ”，返回窗体上，按钮已经自动更名，如图 10.137 所示。

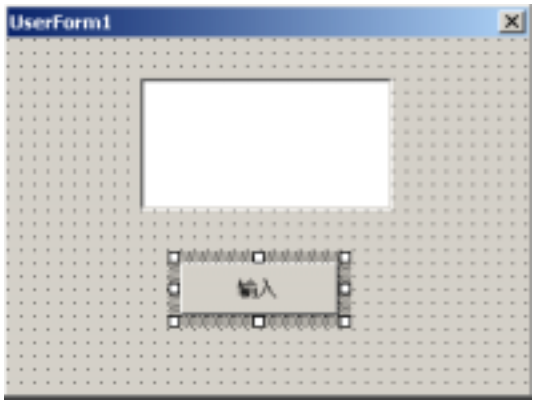


图 10.137 设置属性

5 双击此按钮，切换至代码编辑区域，在其中

输入代码，如图 10.138 所示。

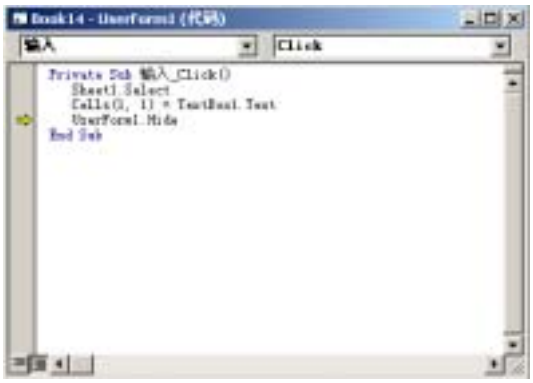


图 10.138 输入代码

其中输入代码如下：

```
Private Sub 输入_Click( )
    Sheet1.Select
    Cells(1,1)=TextBox1.Text
    UserForm1.Hide
End Sub
```

6 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，如图 10.139 所示。

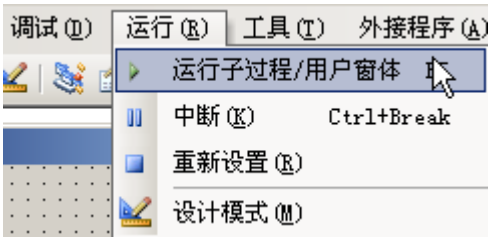


图 10.139 选择“ 运行子过程/用户窗体 ”命令

7 屏幕上出现对话框，在文本框中输入“ chinabiddin.com ”，然后单击“ 输入 ”按钮，如图 10.140 所示。



图 10.140 输入数据

8 返回工作表中，单元格 A1 中显示刚才输入的信息，如图 10.141 所示。

	A
1	chinabidding.com
2	
3	

图 10.141 显示输入值

如何在窗体上控制指定单元格的填充颜色

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入/用户窗体 ”命令，显示窗体面板，如图 10.142 所示。



图 10.142 插入窗体

2 在窗体上添加两个按钮，分别将“ 名称 ”更改为“ 红色 ”和“ 黑色 ”，将 Caption 更改为“ 红色 ”和“ 黑色 ”，如图 10.143 所示的是红色按钮。



图 10.143　设置属性

3　分别双击这两个按钮，在切换的代码编辑区内输入相应的代码，如图 10.144 所示。

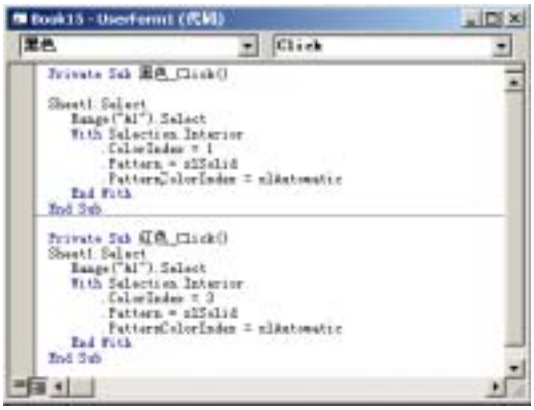


图 10.144　输入的代码

其中输入的代码如下：

```
Private Sub  黑色_Click()
Sheet1.Select
    Range("A1").Select
    With Selection.Interior
        .ColorIndex = 1
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
    End With
End Sub
Private Sub  红色_Click()
Sheet1.Select
    Range("A1").Select
    With Selection.Interior
        .ColorIndex = 3
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
    End With
End Sub
```

4　在工具栏上单击“ 运行子过程/用户窗体 ”按钮，如图 10.145 所示。

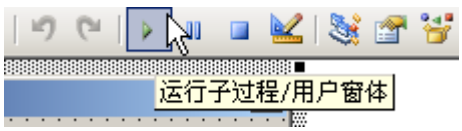


图 10.145　单击“ 运行子过程/用户窗体 ”按钮

5　屏幕上出现包含两个按钮的对话框，如图 10.146 所示。



图 10.146　弹出对话框

6　单击“ 红色 ”按钮，将会自动切换至工作表中，并且单元格 A1 被红色填充，如图 10.147 所示。

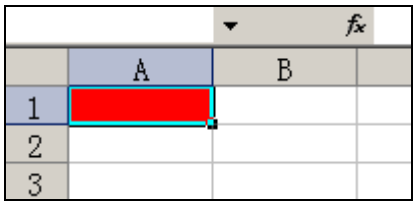


图 10.147　填充颜色

如何更改窗体上字体的格式

具体操作步骤如下。

1　在 Visual Basic 编辑器窗口下，创建如图 10.148 所示的窗体，单击文本框。

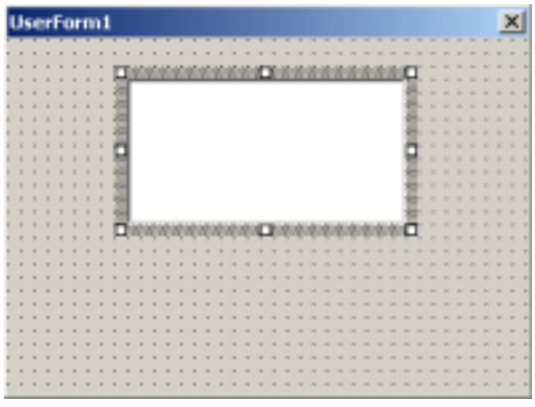


图 10.148　单击文本框

2　在其“ 属性 ”列表框中查找到 Font 选项，如

图 10.149 所示。



图 10.149 选择 Font 选项

3 单击…按钮，弹出“ 字体 ”对话框，可以进行需要的设置，如图 10.150 所示。



图 10.150 设置字体

4 单击“ 确定 ”按钮，运行此用户窗体后，用户在单元格中输入的字体如图 10.151 所示。



图 10.151 修改的字体效果

如何使用窗体控制弹出提示对话框

具体操作步骤如下。

1 在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入 | 用户窗体 ”命令，显示窗体面板，如图 10.152 所示。

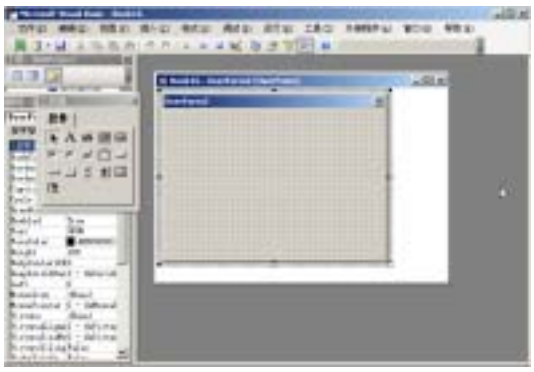


图 10.152 插入窗体

2 在窗体上添加一个按钮，将“ 名称 ”更改为“ 弹出提示 ”，将 Caption 更改为“ 弹出提示 ”，如图 10.153 所示。

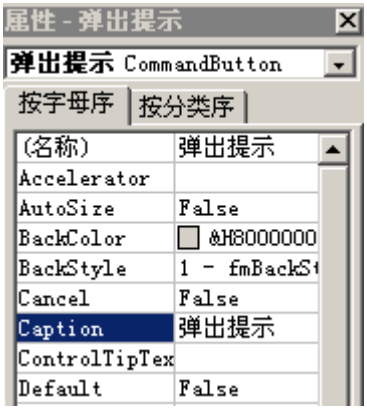


图 10.153 属性设置

3 返回窗体上，此时按钮上的名称已经做了相应的改变，如图 10.154 所示。

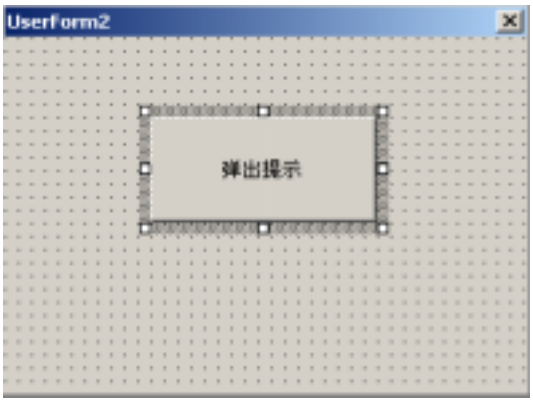


图 10.154　更改名称

4　双击此按钮，切换至代码编辑区内，输入相应的代码，如图 10.155 所示。



图 10.155　输入代码

输入的代码如下：

```
Private Sub 弹出提示_Click( )
    shown = MsgBox(〝眼睛不要离显示器屏幕太近哦！〞,vbOKCancel,提示)
End Sub
```

5　在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，如图 10.156 所示。

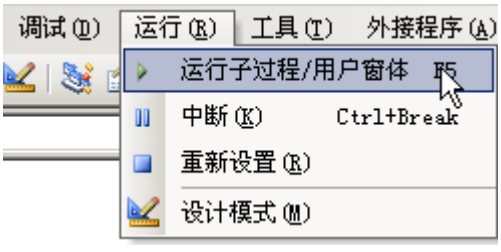


图 10.156　选择 “ 运行子过程/用户窗体 ” 命令

6　屏幕上弹出对话框，如图 10.157 所示。



图 10.157　弹出对话框

7　单击“ 弹出提示 ”对话框，弹出提示对话框，如图 10.158 所示。



图 10.158　提示对话框

如何将单选按钮与单元格建立链接关系

具体操作步骤如下。

1　在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，如图 10.159 所示。

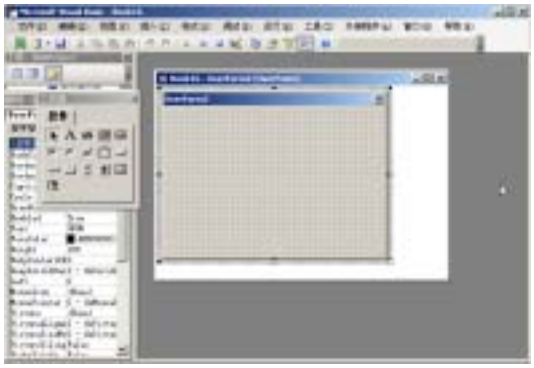


图 10.159　插入窗体

2　在工具箱上单击“ 选项按钮 ”按钮，如图 10.160 所示。



图 10.160 单击“ 选项按钮 ”按钮

3 在窗体上面绘制两个选项按钮框，分别在其左侧的“ 属性 ”列表框中，将这两个选项按钮的 Caption 更改为“ 男 ”和“ 女 ”。

4 返回窗体上，两个选项按钮如图 10.161 所示。

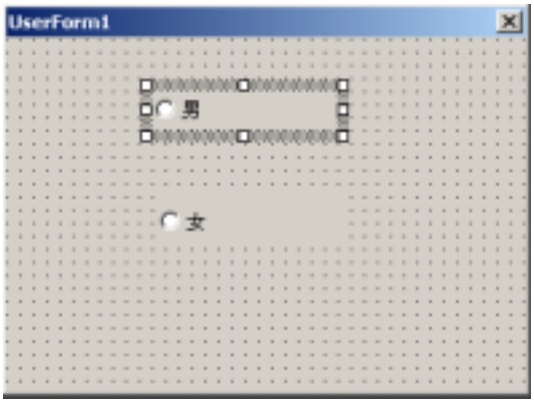


图 10.161 显示选项按钮

5 分别双击这两个选项按钮，在切换的代码编辑区内，输入相应的代码，如图 10.162 所示。

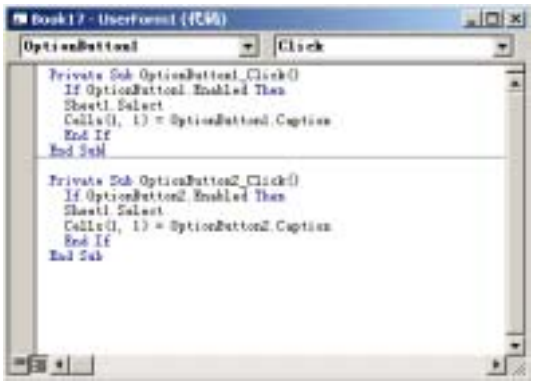


图 10.162 输入代码

输入的代码如下：

```
Private Sub OptionButton1_Click( )
    If OptionButton1.Enabled Then
        Sheet1.Select
        Cells(1,1) = OptionButton1.Caption
        End If
    End Sub
    Private Sub OptionButton2_Click( )
        If OptionButton2.Enabled Then
        Sheet1.Select
        Cells(1,1) = OptionButton2.Caption
        End If
    End Sub
```

6 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，屏幕上显示对话框，选择“ 男 ”单选按钮，如图 10.163 所示。

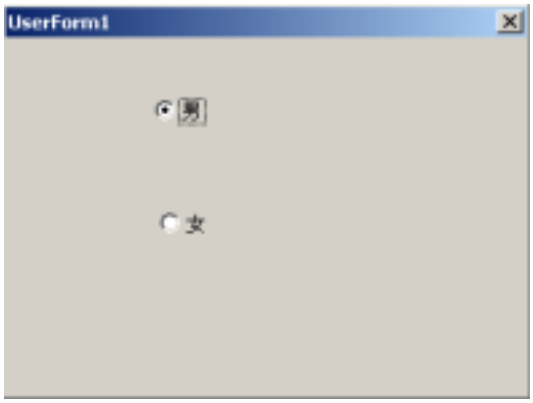


图 10.163 显示信息

7 此时在工作表的单元格 A1 上也会相应的显示数据“ 男 ”，如图 10.164 所示。

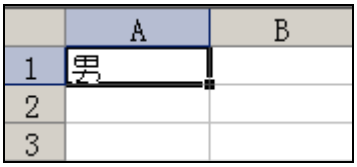


图 10.164 显示效果

如何将切换按钮与单元格建立链接关系

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上选择“ 插入｜用户窗体 ”命令，显示窗体面板，如图 10.165 所示。

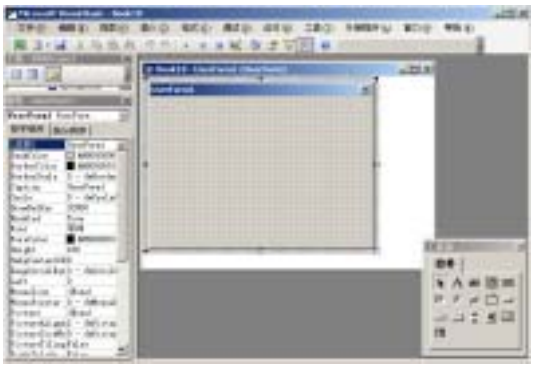


图 10.165 插入窗体

2 在工具箱上单击“ 切换按钮 ”按钮，如图 10.166 所示。



图 10.166 单击“ 切换按钮 ”按钮

3 返回窗体上，在适当位置添加一个切换按钮，然后在其左侧的“ 属性 ”列表框中，将“ 名称 ”更改为“ 切换提示 ”，将 Caption 更改为“ 切换 ”。

4 返回窗体上，切换按钮的名称已经改变，如图 10.167 所示。

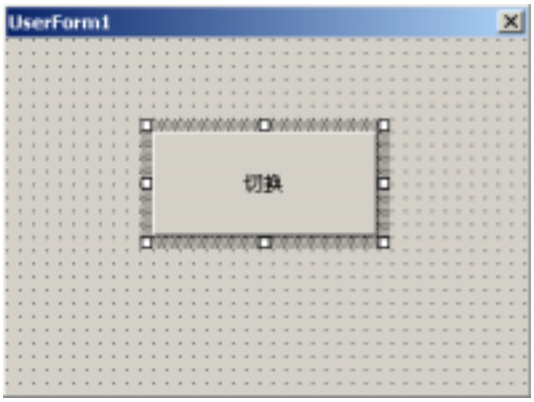


图 10.167 更改名称

5 双击“ 切换 ”按钮，在切换的代码编辑区输入代码，如图 10.168 所示。



图 10.168 输入代码

其中输入的代码如下：

```
Private Sub 切换_Click()
    If 切换.Value = ture Then
    Sheet1.Select
    Cells(1, 1) = "外面下雨了!"
    Else
    Sheet1.Select
    Cells(1, 1) = "外面没有下雨!"
    End If
End Sub
```

6 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，屏幕上显示对话框，单击“ 切换 ”按钮，单元格 A1 中便会显示“ 外面没有下雨 !”的信息。如图 10.169 所示。



图 10.169 显示信息 1

7 再次单击“ 切换 ”按钮，单元格 A1 中便会显示“ 外面下雨了 !”的信息，如图 10.170 所示。



图 10.170 显示信息 2

如何在窗体上使用按钮控制标签的显示内容

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入 | 用户窗体 ”命令，显示窗体面板，如图 10.171 所示。



图 10.171 插入窗体

2 在工具箱上单击“ 标签 ”按钮，如图 10.172 所示。



图 10.172 单击“ 标签 ”按钮

3 在窗体上单击并拖动出标签区域，在其左侧的“ 属性 ”列表框中双击 Font 字样，弹出“ 字体 ”对话框，进行如图 10.173 所示的设置。



图 10.173 设置“ 字体 ”对话框

4 单击“ 确定 ”按钮，返回“ 属性 ”列表框中，将 Caption 清空，如图 10.174 所示。

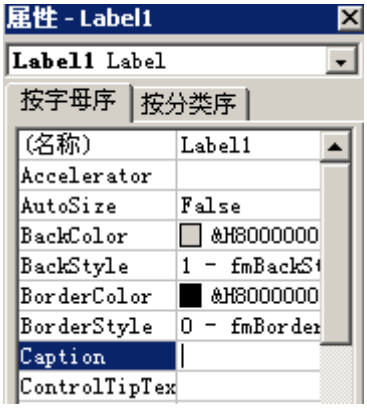


图 10.174 清空 Caption

5 在窗体上添加一个按钮，将“ 名称 ”更改为“ 显示 ”，将 Caption 更改为“ 显示 ”。

6 返回窗体中，添加的标签和按钮，如图 10.175 所示。

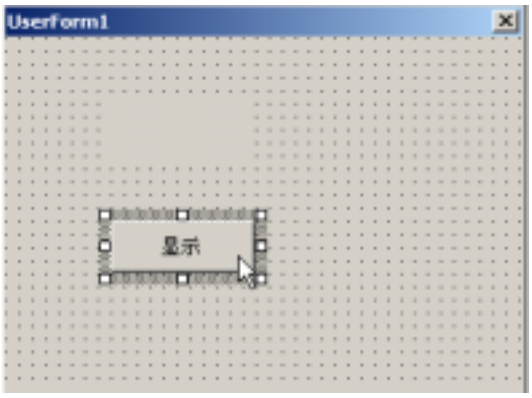


图 10.175　添加的标签和按钮

7　双击“ 显示 ”按钮，切换至代码编辑区下，在其中输入相应的代码，如图 10.176 所示。

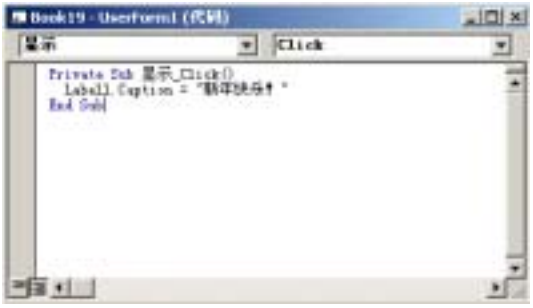


图 10.176　输入代码

输入的代码如下：

```
Private Sub  显示_Click( )
   Labell.Caption = “新年快乐！”
End Sub
```

8　在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，屏幕上显示对话框，单击“ 显示 ”按钮，这样，在标签区域内就可以显示预先设定的文本信息，如图 10.177 所示。



图 10.177　显示文本信息

如何使用窗体来选择工作表中的单个单元格

具体操作步骤如下。

1　在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，如图 10.178 所示。



图 10.178　插入窗体

2　在窗体上添加一个文字框和一个按钮，将按钮的 Caption 属性设置为“ 选择 ”，并将 Font 属性进行相应的设置，如图 10.179 所示。



图 10.179　设置属性

3　返回窗体，添加的文字框和按钮，如图 10.180 所示。

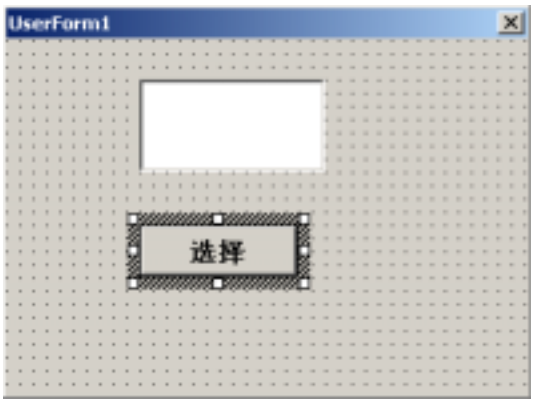


图 10.180 添加控件

4 双击“ 选择 ”按钮，切换至代码编辑区内，输入相应的代码，如图 10.181 所示。

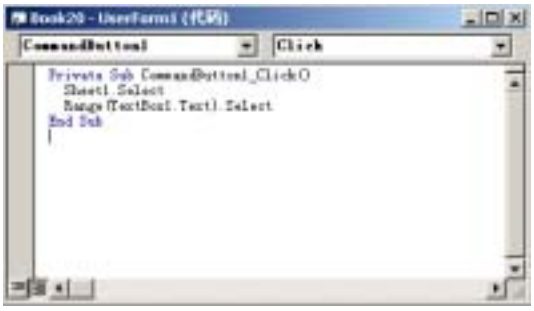


图 10.181 输入代码

其中输入的代码如下：

```
Private Sub CommandButton1_Click( )
   Sheet1.Select
   Range(TextBox1.Text).Select
End Sub
```

5 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，如图 10.182 所示。

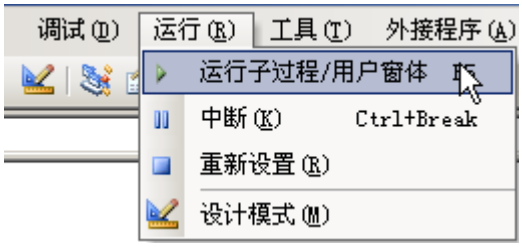


图 10.182 选择“ 运行子过程/用户窗体 ”命令

6 屏幕上出现对话框，在文本框中输入 B2，单击“ 选择 ”按钮，系统就可以在工作表中选定单元格 G4，如图 10.183 所示。



图 10.183 选定单元格

如何在窗体上实现两个数的加法运算

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，在窗体上添加标签和文字框，如图 10.184 所示。



图 10.184 插入窗体

2 在窗体上添加 3 个文字框和两个标签，如图 10.185 所示。

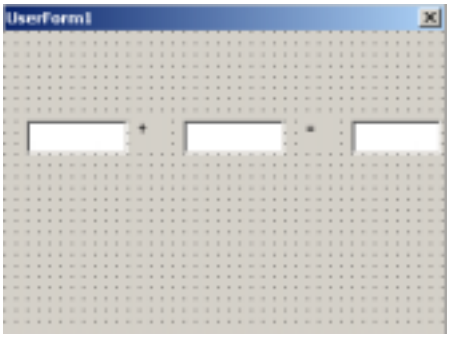


图 10.185 添加控件

3 再在窗体上添加 1 个按钮，并将这个按钮的 Caption 属性和名称更改为“ 计算 ”，如图 10.186 所示。



图 10.186 更改属性

4 返回窗体上，此时添加的控件如图 10.187 所示。

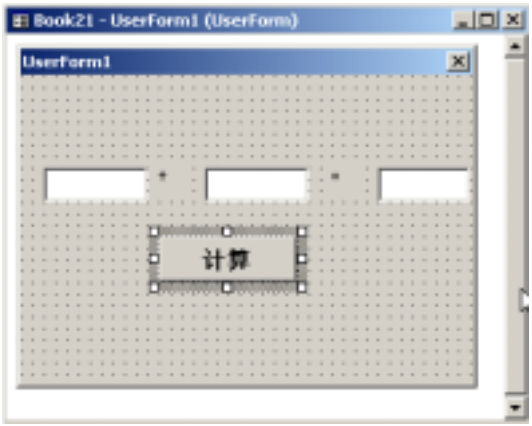


图 10.187 添加控件

5 双击“ 计算 ”按钮，在代码编辑区输入相应的代码，如图 10.188 所示。

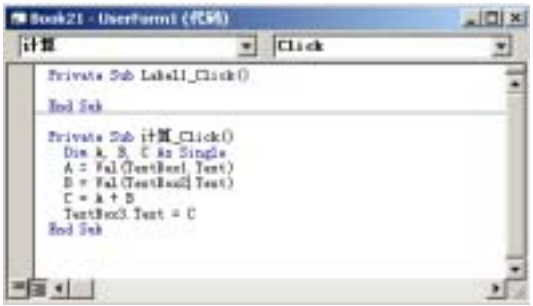


图 10.188 输入代码

其中输入的代码如下：

```
Private Sub 计算_Click( )
    Dim A,B,C As Single
    A = Val(TextBox1.Text)
    B = Val(TextBox2.Text)
    C = A + B
    TextBox3.Text = C
End Sub
```

6 在菜单栏上选择“ 运行｜运行子过程/用户窗体 ”命令，如图 10.189 所示。

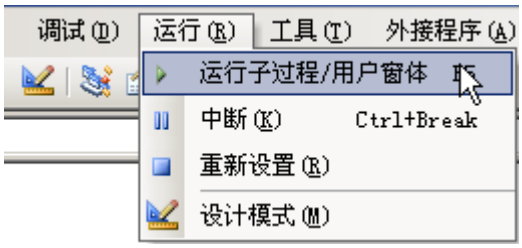


图 10.189 选择“ 运行子过程/用户窗体 ”命令

7 屏幕上出现对话框，在前两个文本框中分别输入“ 456 ”和“ 789 ”，单击“ 计算 ”按钮，第 3 个文本框中便显示出计算值“ 1245 ”，如图 10.190 所示。

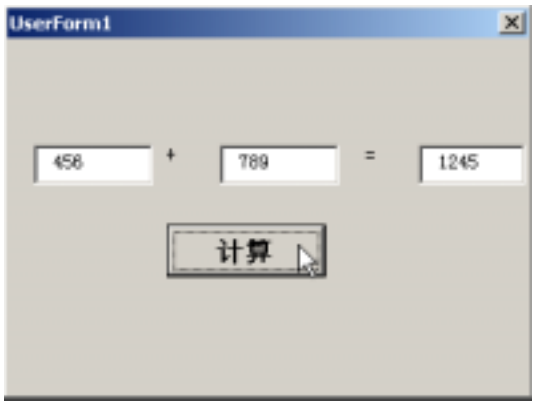


图 10.190 计算和值

如何测试窗体的可靠性

主要有以下几种方法。

1 在 Visual Basic 编辑器窗口中调用窗体进行测试。

2 在菜单栏上选择“ 运行｜运行子程序/用户窗

体”命令。

3 直接按下 F5 键。

4 在工具栏上单击“运行子过程/用户窗体”按钮。

如何在工作簿启动后自动弹出窗体

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口左侧的“工程”列表框中，双击“ThisWorkbook”选项。

2 在其右侧的代码编辑区内，单击第 1 个下拉列表按钮，在其中选择“Workbook”选项，如图 10.191 所示。



图 10.191 选择“Workbook”选项

3 然后在第 2 个下拉列表中，选择“Open”选项，在其下方的代码编辑区内，输入相应的代码，如图 10.192 所示。

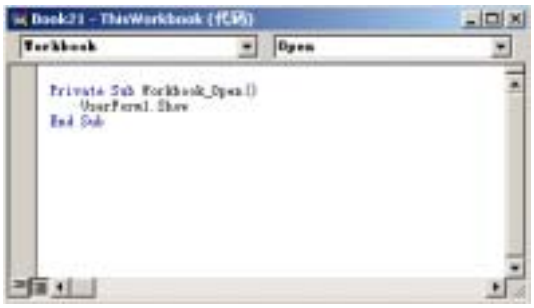


图 10.192 输入代码

其中输入的代码如下：

```
Private Sub Workbook_Open( )
    UserForm1.Show
End Sub
```

4 关闭此工作簿文件，再次重新打开它时，就会自动弹出如图 10.193 所示的对话框。

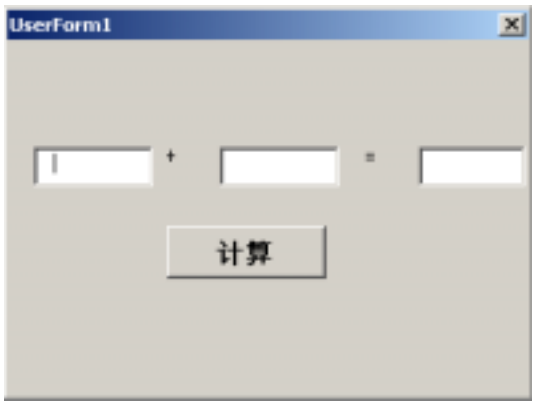


图 10.193 自动弹出对话框

如何为窗体添加背景图案

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中的菜单栏上，选择“插入｜用户窗体”命令，显示窗体面板，在窗体上添加标签和文字框，如图 10.194 所示。



图 10.194 插入窗体

2 单击窗体，在其左侧的“属性”列表框中，将 Caption 属性设置为“添加背景图案”,如图 10.195 所示。

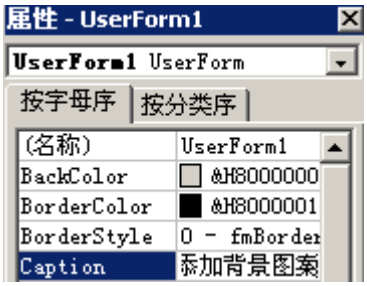


图 10.195 更改属性

3 双击 Pictuer 属性按钮，弹出“加载图片”对话框，在其中选择需要的图片，如图 10.196 所示。



图 10.196 插入图片

4 单击“ 打开 ”按钮，选定的图片便被插入至窗体中，如图 10.197 所示。



图 10.197 插入背景图片

5 在窗体上添加一个标签，输入“ 湖边 ”，在其左侧的“ 属性 ”列表框，双击 Font 字体，弹出“ 字体 ”对话框，进行如图 10.198 所示的设置。



图 10.198 设置“ 字体 ”对话框

6 然后将 BackStyle（背景模式）设置为“ 0-fmBack ”，如图 10.199 所示。

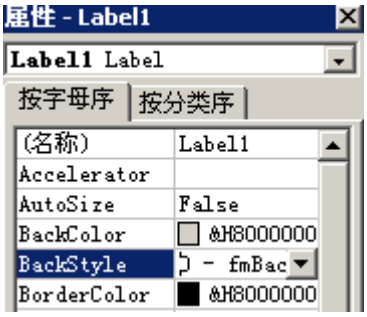


图 10.199 设置属性

7 返回工作表中，添加的背景和标签如图 10.200 所示。



图 10.200 添加的背景和标签效果

如何制作修改化的 Splash 屏幕

具体操作步骤如下。

1 在 Visual Basic 编辑器窗口中，打开上一技巧制作的窗体，效果如图 10.201 所示。



图 10.201 打开窗体

2 在 Visual Basic 编辑器窗口左侧的“ 工程 ”列表框中，双击“ ThisWorkbook ”选项，在其右侧

的代码编辑区内，单击第 1 个下拉列表按钮，在其中选择“ Workbook ”选项，然后在第 2 个下拉列表中选择“ Open ”选项，然后在代码编辑区内输入相应的代码，如图 10.202 所示。

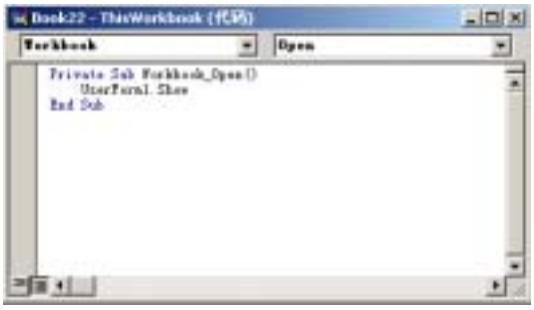


图 10.202 输入代码

其中输入的代码如下：

```
Private Sub Workbook_Click( )
    UserForm1.Show
End Sub
```

3 右击窗体，弹出快捷菜单，选择“ 查看代码 ”命令。

4 切换至代码编辑区内，在其中输入相应的代码，如图 10.203 所示。

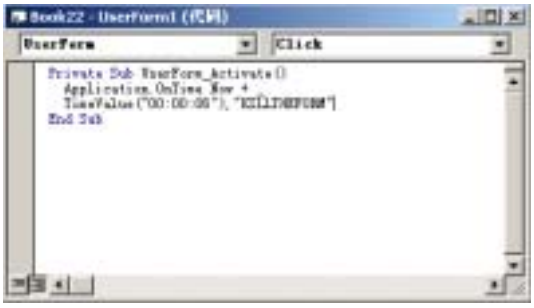


图 10.203 输入代码

其中输入的代码如下：

```
Private Sub UserForm_Activate( )
    Application.OnTime Now + _
    TimeValue(“00:00:06”),”KILLTHEFORM”
End Sub
```

5 在 Visual Basic 编辑器窗口左侧的“ 工程 ”列表框中，右击 VBAProject，选择“ 插入|模块 ”命令，如图 10.204 所示。



图 10.204 选择“ 插入 | 模块 ”命令

6 在切换至的代码编辑区内输入相应的代码，如图 10.205 所示。



图 10.205 输入代码

其中输入的代码如下：

```
Sub KILLTHEFORM( )
    Unload UserForm1
End Sub
```

7 关闭此工作簿文件，重新打开时，系统会同时显示如图 10.206 所示的对话框，6 秒钟后自动消失。



图 10.206 自动显示

如何禁用窗体的关闭按钮

具体操作步骤如下。

1 在窗体上添加一按钮并双击，进入代码编辑

区，输入相应的代码，如图 10.207 所示。

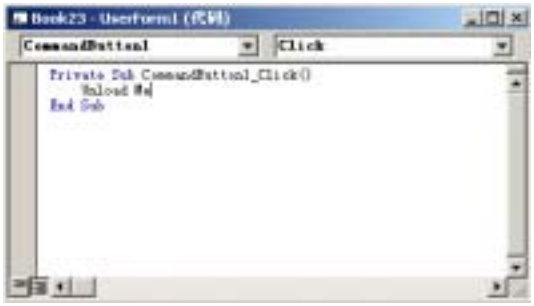


图 10.207 输入代码

其中输入的代码如下：

```
Private Sub CommandButton1_Click( )
    Unload Me
End Sub
```

2 右击窗体，弹出快捷菜单，选择“ 查看代码 ”命令。

3 切换至代码编辑区内，单击第 2 个下拉列表按钮，弹出下拉列表，选择“ Queryclose ”选项，如图 10.208 所示。

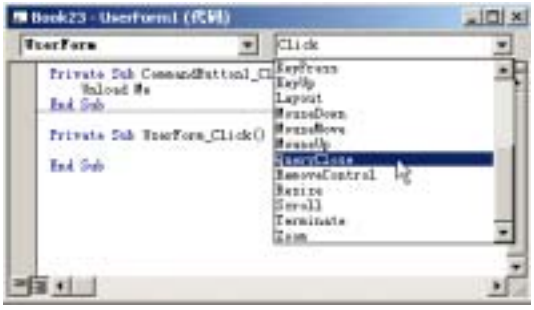


图 10.208 Visual Basic 编辑器窗口

4 在代码编辑区内输入相应的代码，如图 10.209 所示。

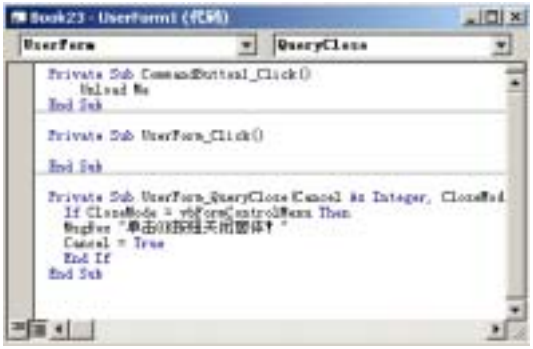


图 10.209 输入代码

其中输入的代码如下：

```
Private Sub UserForm_QueryClose (Cancel A
s Integer,CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
    MsgBox〝单击 OK 按钮关闭窗体！〞
    Cancel = True
End If
End Sub
```

5 运行此用户窗体，这样，当单击窗体右上角的“ 关闭 ”按钮关闭窗体时，就会弹出提示对话框禁止直接关闭，如图 10.210 所示。

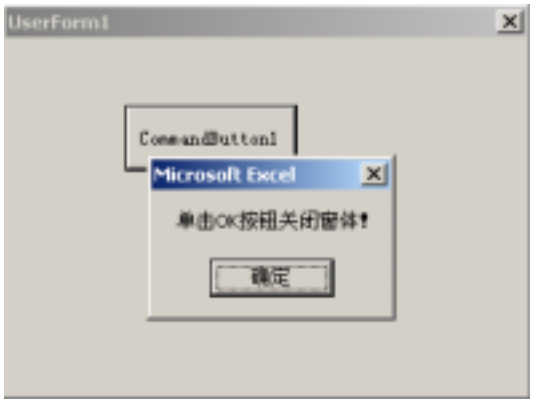


图 10.210 提示对话框

如何自定义用户窗体的大小

具体操作步骤如下。

1 在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，再在窗体上添加一个按钮，如图 10.211 所示。



图 10.211 添加按钮

2 单击此按钮，在其左侧的属性列表框中，将 Caption 属性设置为“ 缩小 ”，如图 10.212 所示。



图 10.212 更改属性

3 返回窗体上，双击添加的按钮，切换至代码编辑区内，在其中输入相应的代码，如图 10.213 所示。



图 10.213 输入代码

其中输入的代码如下：

```
Private Sub CommandButton1_Click()
  If CommandButton1.Caption = "缩小" Then
  UserForm1.Height = 150
  CommandButton1.Caption = "放大"
  Else
  UserForm1.Height = 180
  CommandButton1.Caption = "缩小"
  End If
```

End Sub4 运行此窗体，此时对话框上的按钮名称是缩小，如图 10.214 所示。

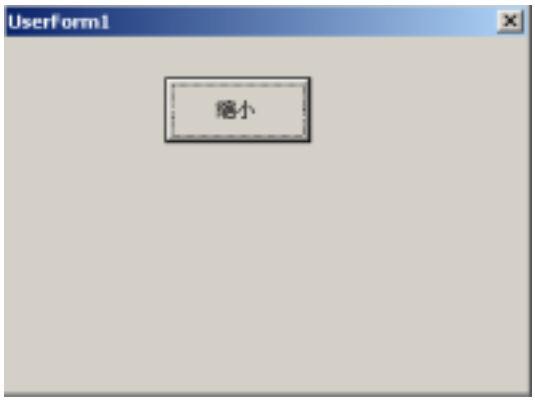


图 10.214 显示对话框

5 单击“ 缩小 ”按钮，此时按钮名称自动更改为“ 放大 ”，而且窗体的大小变小，如图 10.215 所示。



图 10.215 更改大小

如何在用户窗体上实现工作表的滚动

具体操作步骤如下。

1 在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，在其中添加滚动条，如图 10.216 所示。

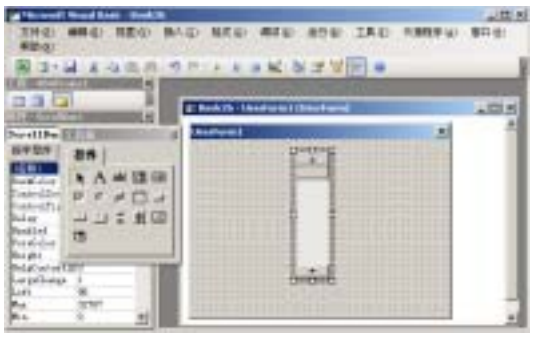


图 10.216 添加滚动条

2 双击此滚动条，切换至代码编辑区内，在其

中输入相应的代码，如图 10.217 所示。

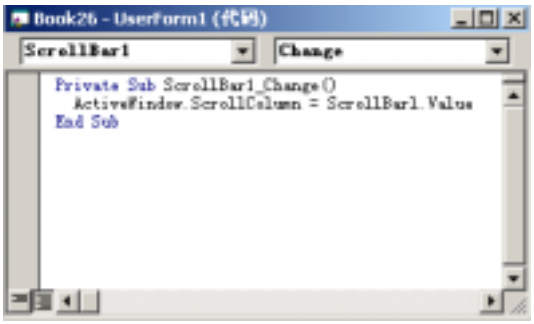


图 10.217　输入代码

其中输入的代码如下：

```
Private Sub ScrollBar1_Change( )
   ActiveWindow.ScrollColumn =
ScrollBar1.Value
End Sub
```

3　右击窗体空白处，弹出快捷菜单，选择“ 查看代码 ”命令。

4　在切换的代码编辑区内，输入相应的代码，如图 10.218 所示。

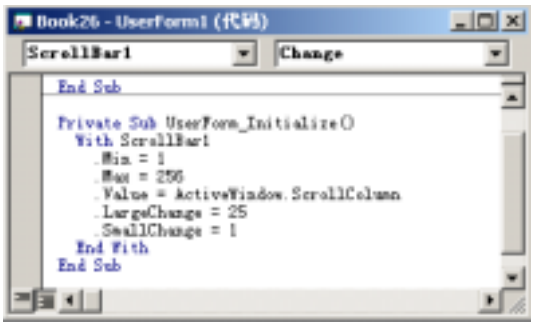


图 10.218　输入代码

其中输入的代码如下：

```
Private Sub UserForm_Initialize( )
   With ScrollBar1
      .Min = 1
      .Max = 256
      .Value = ActiveWindow.ScrollColumn
      .LargeChange = 25
      .SmallChange = 1
   End With
End Sub
```

5　运行此窗体，在对话框中移动滚动条，工作表便会发生相应的变化，如图 10.219 所示。

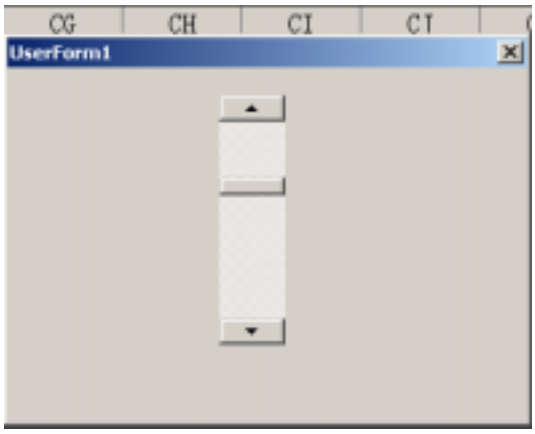


图 10.219　滚动工作表

如何创建专业符号工具栏

具体操作步骤如下。

1　在菜单栏中选择“ 工具 ”|“ 宏 ”|“ 录制新宏 ”，弹出“ 录制新宏 ”对话框。

2　在“ 宏名 ”文本框中输入要保存的宏名，在“ 快速键 ”栏中可设置录制宏的快捷键，并在“ 保存在 ”下拉列表框中选择宏保存的位置，然后单击“ 确定 ”按钮开始录制。

3　单击“ 录制宏 ”工具栏上的“ 相对引用 ”按钮，然后将需要的特殊符号输入到某个单元格中，再单击“ 录制宏 ”工具栏上的“ 停止 ”按钮即完成宏的录制。

4　在菜单栏中选择“ 工具 ”|“ 自定义 ”，弹出“ 自定义 ”对话框，在“ 工具栏 ”选项卡中单击“ 新建 ”按钮，弹出“ 新建工具栏 ”对话框。

5　在“ 工具栏的名称 ”文本框中输入要存放特殊符号的工具栏名称，然后单击“ 确定 ”按钮，新建的工具栏便显示在“ 工具栏 ”的列表框和窗口中。

6　单击“ 命令 ”选项卡，选中“ 类别 ”列表中的“ 宏 ”复选框，再将“ 命令 ”列表中的“ 所有 ”项拖到窗口中的新建工具栏上。

7　在新建工具栏的“ 自定义按钮 ”按钮上，单击右键从快捷菜单中选择“ 指定宏 ”命令，打开“ 指定宏 ”对话框。

8　在“ 宏名 ”列表中选中相应的宏，然后单击“ 确定 ”按钮。

9　重复第 (7) 步，将按钮与相应的宏链接起来。

10　关闭“ 自定义 ”对话框，以后就可以象使用普通工具栏一样使用新建的工具栏。

如何使用 ListBox 控件添加选项（一）

使用 ListBox 控件添加选项有很多种，下面介绍第一种方法，具体操作步骤如下。

1 在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，如图 10.220 所示。

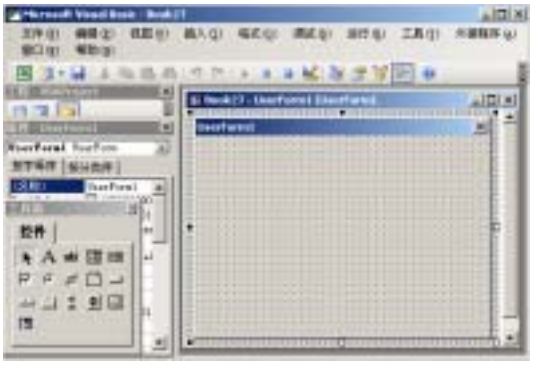


图 10.220 添加窗体

2 在工具箱上单击“ 列表框 ”按钮，在窗体上添加控件，如图 10.221 所示。



图 10.221 添加“ 列表框 ”控件

3 在其左侧的“ 属性 ”列表框中，将 RowSource 属性设置为 Sheet1!A1:A10，如图 10.222 所示。



图 10.222 设置属性

4 在工作表中的单元格区域 A1:A10 中输入如图 10.223 所示的数据。

	A	B
1	1995	
2	1996	
3	1997	
4	1998	
5	1999	
6	2000	
7	2001	
8	2002	
9	2003	
10	2004	
11		

图 10.223 输入数据

5 运行此窗体，在显示的对话框窗体中，“ 列表框 ”中显示出输入的数据，如图 10.224 所示。

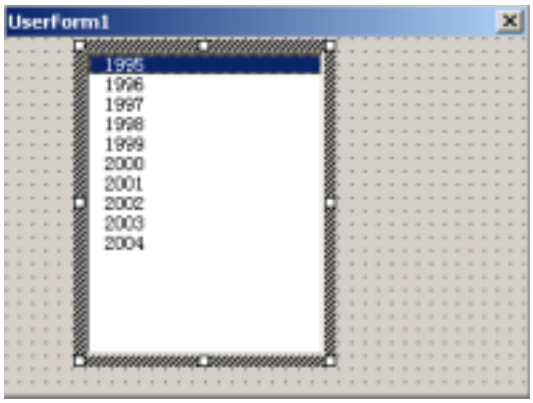


图 10.224 选择选项

如何使用 ListBox 控件添加选项（二）

第 2 种使用 ListBox 控件添加选项的方法，具体操作步骤如下。

1 右击窗体的空白处，弹出快捷菜单，选择“ 查看代码 ”命令，切换至代码编辑区内，在第 2 个下拉列表中选择“ Activate ”选项，然后在其下侧的代码编辑区内输入相应代码，如图 10.225 所示。

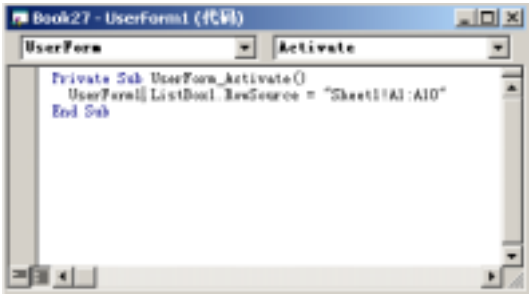


图 10.225　输入代码

其中输入的代码如下：

```
Private Sub UserForm_Activate( )
    UserForm1.ListBox1.RowSource = “Sheet
1!A1:A10”
End Sub
```

2　运行此窗体，显示的对话框中同样会显示如图 10.226 所示的列表框。

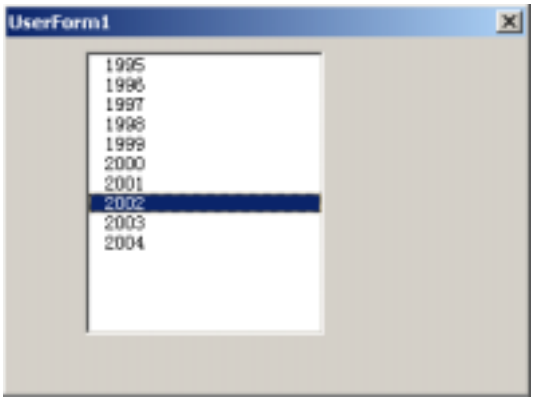


图 10.226　显示列表框

如何将“ 列表框 ”控件中的任意选项输入指定工作表的单元格内

具体操作步骤如下。

1　打开上面创建的的窗体，如图 10.227 所示。

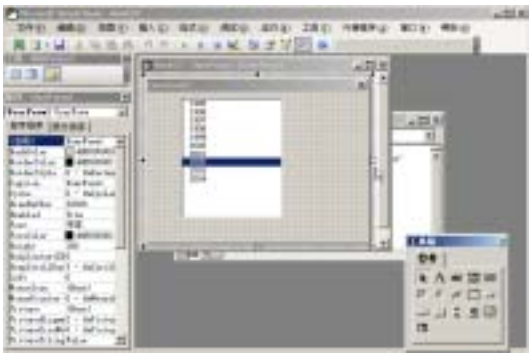


图 10.227　打开窗体

2　在此窗体中添加一个按钮，然后在其左侧的“ 属性 ”列表框中，将 Caption 属性设置为“ 添加 ”，将“ 名称 ”设置为“ 添加 ”。

3　返回窗体上，双击此按钮，切换至代码编辑区下，在其中输入如图 10.228 所示的代码。

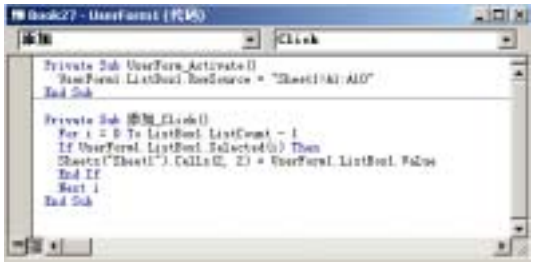


图 10.228　输入代码

输入的代码如下：

```
Private Sub  添加_Click()
    For i = 0 To ListBox1.ListCount - 1
    If UserForm1.ListBox1.Selected(i) Then
    Sheets("Sheet1").Cells(2,          2)          =
UserForm1.ListBox1.Value
    End If
    Next i
End Sub
```

4　运行此窗体，在列表框中选择需要输入的选项，然后单击“ 添加 ”按钮，工作表中的相应单元格便会输入选定的选项，如图 10.229 所示。

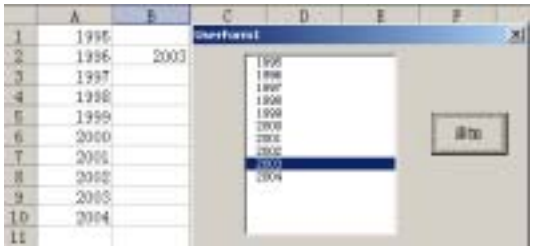


图 10.229　输入选项

如何创建含有更改内容的“ 列表框 ”控件

具体操作步骤如下。

1　在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，在其中添加两个单选按钮和一个“ 列表框 ”控件，如

图 10.230 所示。

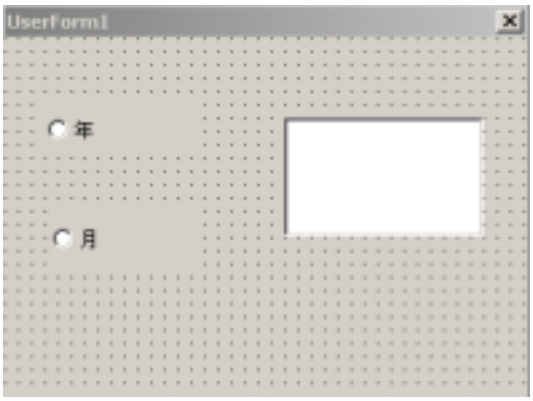


图 10.230 添加控件

2 分别双击“ 年 ”和“ 月 ”单选按钮，切换至相应的代码编辑区内，分别输入相应的代码如图 10.231 所示。

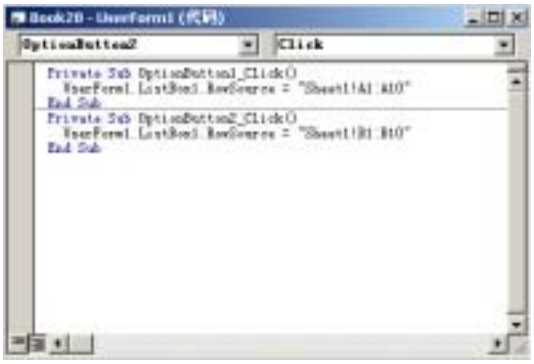


图 10.231 输入代码

其中输入的代码如下：

```
Private Sub OptionButton1_Click()
    UserForm1.ListBox1.RowSource = "Sheet1!A1:A10"
End Sub
Private Sub OptionButton2_Click()
    UserForm1.ListBox1.RowSource = "Sheet1!B1:B10"
End Sub
```

3 运行此窗体，选择“ 年 ”单选按钮，列表框中显示年选项，如图 10.232 所示。

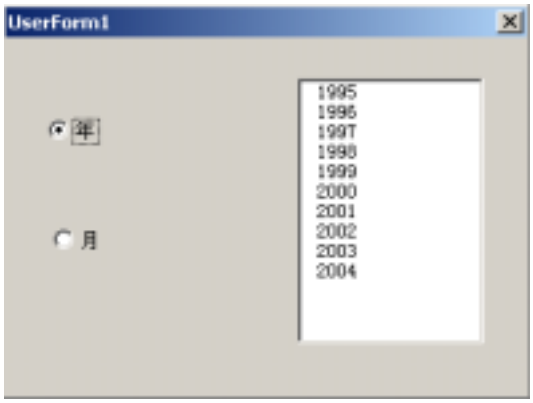


图 10.232 显示年

4 选择“ 月 ”单选按钮，列表框中显示月选项，如图 10.233 所示。

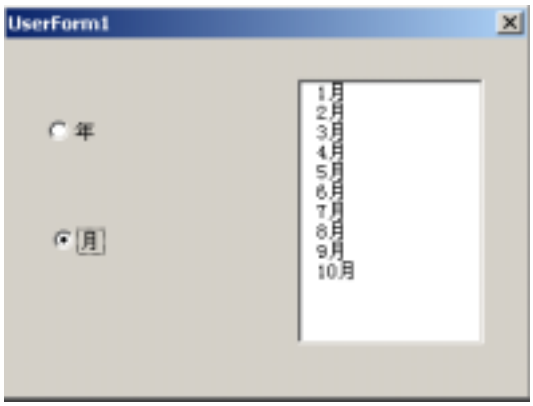


图 10.233 显示月

如何从另一个列表建立列表框

具体操作步骤如下。

1 在“ Visual Basic 编辑器 ”窗口中的菜单栏上，选择“ 插入｜用户窗体 ”命令，显示窗体面板，并在其中添加两个“ 列表框 ”控件，如图 10.234 所示。

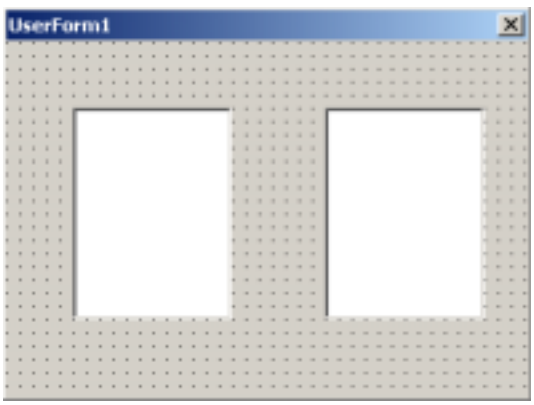


图 10.234　添加控件

2　将左边的“ 列表框 ”控件的 Row Source 属性设置为 Sheet1!B1:B10，如图 10.235 所示。

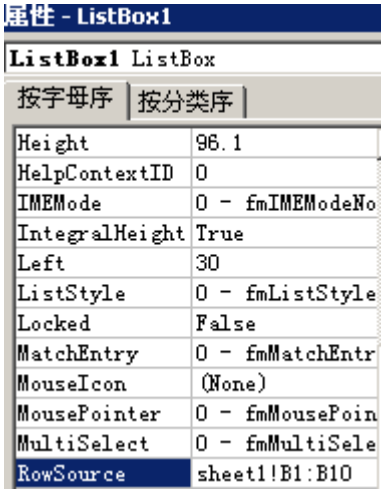


图 10.235　设置属性

3　在窗体上两个“ 列表框 ”控件之间添加一个按钮，并双击按钮，切换至代码编辑区内，输入如图 10.236 所示的代码。

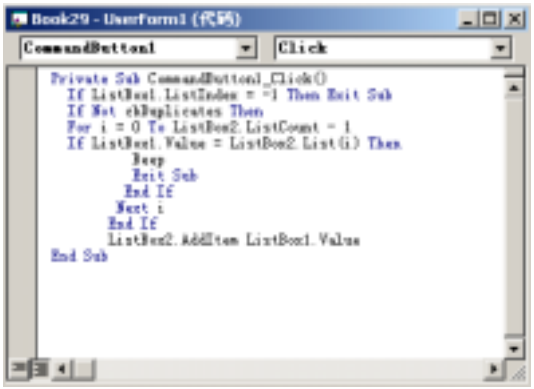


图 10.236　输入代码

其中输入的代码如下：

```
Private Sub CommandButton1_Click()
   If ListBox1.ListIndex = -1 Then Exit Sub
   If Not cbDuplicates Then
   For i = 0 To ListBox2.ListCount - 1
   If ListBox1.Value = ListBox2.List(i) Then
              Beep
              Exit Sub
            End If
          Next i
        End If
        ListBox2.AddItem ListBox1.Value
    End Sub
```

4　运行此窗体，在左边的列表框中，选择月份后，单击“ 添加 ”按钮，右边的列表框便会显示出相应的月份，如图 10.237 所示。

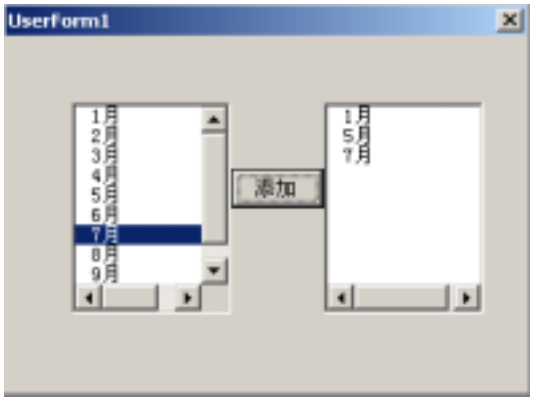


图 10.237　添加月份

如何设置一个定时提醒的功能

1　新建文档，取名保存（如“ 提示.xls ”）。

2　按“ Alt+F11 ”组合键进入 VBA 编辑状态。

3　在左侧“ 工程资源管理器 ”窗口中，选中“ VBAProject(提示.xls) ”选项，执行“ 插入模块 ”命令，插入一个新模块。

4　将下述代码输入到右侧的编辑区域中：

```
Sub auto_open()
Application.OnTime TimeValue("08:30:00"), "tx"
Application.OnTime TimeValue("09:30:00"), "tx"
End Sub
Sub tx()
msg = MsgBox("休息一下！", vbInformation, "小提示")
End Sub
```

5　输入完成后，退出 VBA 编辑状态，保存工作簿文档。

6　这样，到 08:30:00 就可以启动 Excel，打开上述工作簿文档。设定时间 09:30:00 一到，系统就会弹出提示框。