

自学收集

1、Application.CommandBars("Worksheet Menu Bar").Enabled = false

2、cells(activecell.row,"b").value '活动单元格所在行 B 列单元格中的值

3、Sub CheckSheet()'如果当前工作簿中没有名为 kk 的工作表的话，就增加一张名为 kk 的工作表，并将其排在工作表从左至右顺序排列的最左边的位置，即排在第一的位置

```
Dim shtSheet As Worksheet
```

```
For Each shtSheet In Sheets
```

```
    If shtSheet.Name = "KK" Then Exit Sub
```

```
Next shtSheet
```

```
Set shtSheet = Sheets.Add(Before:=Sheets(1))
```

```
shtSheet.Name = "KK"
```

```
End Sub
```

4、Sheet1.ListBox1.List = Array("一月", "二月", "三月", "四月")'一次性增加项目

5、Sheet2.Rows(1).Value = Sheet1.Rows(1).Value'将一个表中的一行全部拷贝到另一个表中

6、Sub pro_cell()'将此代码放入 sheet1,则 me=sheet1,主要是认识 me

```
Me.Unprotect
```

```
Cells.Locked = False
```

```
Range("D11:E11").Locked = True
```

```
Me.Protect
```

```
End Sub
```

7、Application.CommandBars("Ply").Enabled = False'工作表标签上快捷菜单失效

8、Sub aa()'把 B1 到 B12 单元格的数据填入 c1 到 c12

```
For i = 1 To 12
```

```
    Range("C" & i) = Range("B" & i)
```

```
Next i
```

```
End Sub
```

9、ActiveCell.AddComment

Selection.Font.Size = 12'在点选的单元格插入批注，字体为 12 号

10、Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)

```
    Cancel = True
```

```
End Sub
```

11、ScrollArea 属性

参阅应用于示例特性以 A1 样式的区域引用形式返回或设置允许滚动的区域。用户不能选定滚动区域之外的单元格。String 类型，可读写。

说明

可将本属性设置为空字符串 ("") 以允许对整张工作表内所有单元格的选定。

示例

本示例设置第一张工作表的滚动区域。

```
Worksheets(1).ScrollArea = "a1:f10"
```

```
12\if application.max([a1:e1])=10 then
```

```
msgbox""
```

```
commandbutton1.enabled=false
```

'A1—E1 最大的数值达到 10 时，自动弹出对话框，并冻结按钮

12、本示例将更改的单元格的颜色设为蓝色。

```
Private Sub Worksheet_Change(ByVal Target As Range)
```

```
    Target.Font.ColorIndex = 5
```

```
End Sub
```

13、Sub test()'求和

```
Dim rng As Range, rng2 As Range
```

```
For Each rng In ActiveSheet.UsedRange.Columns
```

```
    Set rng2 = Range(Cells(1, rng.Column), Cells(Cells(65536, rng.Column).End(xlUp).Row, rng.Column))
```

```
    rng2.Cells(rng2.Cells.Count).Offset(1, 0) = WorksheetFunction.Sum(rng2)
```

```
Next rng
```

```
End Sub
```

14、将工作簿中的全部 n 张工作表都在 sheet1 中建上链接

```
Sub test2()
```

```
Dim Pt As Range
```

```
Dim i As Integer
```

```
With Sheet1
```

```
    Set Pt = .Range("a1")
```

```
    For i = 2 To ThisWorkbook.Worksheets.Count
```

```
        .Hyperlinks.Add Anchor:=Pt, Address:="", SubAddress:=Worksheets(i).Name & "!A1"
```

```
        Set Pt = Pt.Offset(1, 0)
```

```
    Next i
```

```
End With
```

```
End Sub
```

15、保存所有打开的工作簿，然后退出 Microsoft Excel。

```
For Each w In Application.Workbooks
```

```
    w.Save
```

```
Next w
```

```
Application.Quit
```

16、让 form 标题栏上的关闭按钮失效

```
Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
```

```
If CloseMode <> 1 Then Cancel = True
```

```
End Sub
```

17、Sub countsh()'获得工作表的总数

```
MsgBox Sheets.Count
```

```
End Sub
```

18、Sub IE()'打开个人网页

```
ActiveWorkbook.FollowHyperlink "about:blank"
```

```
SendKeys "{F4}ykk1976.anyp.cn{ENTER}", True
```

```
End Sub
```

19、Sub delback()'一次性删除工作簿中所有工作表的背景

```
For Each shtSheet In Sheets
```

```
    shtSheet.SetBackgroundPicture Filename:=""
```

```
Next shtSheet
```

```
End Sub
```

20、[a1].formula="=b1+c1"A1 中设定公式为=B1+C1

21、Private Sub CommandButton1_Click()'将 A1 到 C6 中大于=3 的数依次放入 E 列

```
Dim i As Long
```

```
r = 1
```

```
For Each i In Range("a1:c6")
```

```
    If i >= 3 Then Cells(r, 5) = i: r = r + 1
```

```
Next
```

```
End Sub
```

22、Private Sub Workbook_SheetChange(ByVal Sh As Object, ByVal Target As Range)'显示带数字的表名

```
b = Split(Sh.Name, "(")
```

```
On Error GoTo ss
```

```
num = CInt(Left(b(1), Len(b(1)) - 1))
```

```
If num >= 1 And num < 20 Then
```

```
MsgBox Sh.Name
```

```
End If
```

```
Exit Sub
```

```
ss:
```

```
MsgBox "error", 16, ""
```

```
End Sub
```

23、Sub Test()'选择所有工作表名以"业报"开头的工作表或头两个字是业报的报表名引用

```
Set Sh = ActiveSheet
```

```
If Left(Sh.Name, 2) = "业报" Then ' 或 if sh.name like "业报*"then
```

```
MsgBox "你成功了", 64, ""
```

```
End If
```

```
End Sub
```

24、1.建立文件夹的方法

```
Mkdir "D:\Music"
```

2.打开文件夹的方法

```
ActiveWorkbook.FollowHyperlink Address:="D:\Music", NewWindow:=True
```

25、在当前工作表翻页

```
Application.SendKeys "{PGUP}", True
```

```
Application.SendKeys "{PGDN}", True
```

或者

```
ActiveWindow.LargeScroll Down:=1
```

```
ActiveWindow.LargeScroll Down:=-1
```

26、当 Target = "*小计"时如何写,*代表任何字符。

```
if instr(target.value,"小计")<>0 then
```

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
If Target.Value Like "*小计" Then MsgBox "OK"
```

```
End Sub
```

27、ActiveCell.FormulaR1C1 = "=SUM(R[1]C:R[14]C,R[59]C:R[78]C)"

这是相对引用的写法：根据推算你的函数是放在"AD6"单元格

你的函数：=SUM(R[1]C:R[14]C 中的 "R"表示行 "C"表示列。

R[1]表示"AD6+1 行", C 表示"列没有变化, 就是同列"那么: R[1]C 就表示 AD7

同理, R[14]表示 AD6+14 行, 表示: AD20。以此类推。

28、Private Sub CommandButton1_Click()'将 A1 到 C6 中大于=3 的数依次放入 E 列

```
Dim i As Long
```

```
Dim iRng As Range
```

```
For Each iRng In Sheets(1).Range("a1:c6")
```

```
    If iRng.Value >= 3 Then
```

```
        i = i + 1
```

```
        Sheets(1).Range("E" & i).Value = iRng.Value
```

```
    End If
```

```
Next
```

```
End Sub
```

29、工作表中的窗体按钮禁用后，按钮形状不变，字体不变，从外表上无法看出其已禁用，如何设置属性使其像控件按钮那样明显的禁用？

```
With ActiveSheet.Buttons(1)
```

```
    .Enabled = False
```

```
    ActiveSheet.Shapes(.Caption).DrawingObject.Font.ColorIndex = 15
```

```
End With
```

復原的方法

```
With ActiveSheet.Buttons(1)
```

```
    .Enabled = True
```

```
    ActiveSheet.Shapes(.Caption).DrawingObject.Font.ColorIndex = xlAutomatic
```

```
End With
```

30、Private Sub Worksheet_SelectionChange(ByVal Target As Range'选定 A1 时要输入密码

```
If Target.Address = "$A$1" Then
```

```
    A = InputBox("请输入密码", "officefans")
```

```
    If A = 1 Then [A1].Select Else [A2].Select
```

```
End If
```

```
End Sub
```

31、如何将工作簿中的命名单元格成批删除！

```
Dim Item As Name
```

```
For Each Item In ActiveWorkbook.Names
```

```
    Item.Delete
```

```
Next Item
```

32、平时只能看到表 1，如要看表 2 和表 3，只能通过表 1 的链接打开,且表 2 和表 3 回到表 1 后，又不可见。

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
    If Target.Address = "$A$3" Then '当点击"$A$3"单元格时...
```

```
        Sheet2.Visible = 1 '取消隐藏
```

```
        Sheet2.Activate '激活
```

```
        ActiveSheet.Range("A1").Select
```

```
End If
```

```
If Target.Address = "$A$6" Then
```

```
Sheet3.Visible = 1 '取消隐藏
```

```
Sheet3.Activate
```

```
ActiveSheet.Range("A1").Select
```

```
End If
```

End Sub

33、将 a2 单元格内容替换为 a1 内容

```
ActiveCell.Replace What:=[a2], Replacement:=[a1]
```

34、如果是要填入名称,则:

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
Selection.Value = ComboBox1.column(1)
```

End Sub

如果是要填入代码和名称的组合 :

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
Selection.Value = cstr(ComboBox1.column(0))+ " "+combobox1.column(1)
```

End Sub

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
Selection.Value = ComboBox1.Value
```

End Sub

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
```

```
    'target.row 代表行号
```

```
    'target.column 代表列号
```

```
    i=target.row '获取行号
```

```
    j=target.column '获取列号
```

End Sub

35、当激活工作表时, 本示例对 A1:A10 区域进行排序。

```
Private Sub Worksheet_Activate()
```

```
    Range("a1:a10").Sort Key1:=Range("a1"), Order:=xlAscending
```

End Sub

36、BeforePrint 事件

参阅应用于示例特性在打印指定工作簿（或者其中的任何内容）之前，产生此事件。

```
Private Sub Workbook_BeforePrint(Cancel As Boolean)
```

Cancel 当事件产生时为 False。如果该事件过程将本参数设为 True，则当该过程运行结束之后不打印工作簿。

示例

本示例在打印之前对当前活动工作簿的所有工作表重新计算。

```
Private Sub Workbook_BeforePrint(Cancel As Boolean)
```

```
    For Each wk in Worksheets
```

```
        wk.Calculate
```

```
    Next
```

End Sub

37、Open 事件

参阅应用于示例特性打开工作簿时，将产生本事件。

```
Private Sub Workbook_Open()
```

示例

每次打开工作簿时，本示例都最大化 Microsoft Excel 窗口。

```
Private Sub Workbook_Open()
```

```
    Application.WindowState = xlMaximized
```

End Sub

38、ActiveSheet 属性

参阅应用于示例特性返回一对象,该对象代表活动工作簿中的,或者指定的窗口或工作簿中的活动工作表(最上面的工作表)。只读。如果没有活动的工作表,则返回 **Nothing**。

说明

如果未给出对象识别符,本属性返回活动工作簿中的活动工作表。

如果某一工作簿在若干个窗口中出现,那么该工作簿的 **ActiveSheet** 属性在不同窗口中可能不同。

示例

本示例显示活动工作表的名称。

MsgBox "The name of the active sheet is " & ActiveSheet.Name

39、Calculate 方法

参阅应用于示例特性计算所有打开的工作簿、工作簿中的一张特定的工作表或者工作表中指定区域的单元格,如下表所示:

要计算 依照本示例

所有打开的工作簿 **Application.Calculate** (或只是 **Calculate**)

指定工作表 指定工作表

指定区域 **Worksheets(1).Rows(2).Calculate**

expression.Calculate

expression 对于 **Application** 对象可选,对于 **Worksheet** 对象和 **Range** 对象必需。该表达式返回“应用于”列表中的对象之一。

示例

本示例计算 **Sheet1** 已用区域中 **A** 列、**B** 列和 **C** 列的公式。

Worksheets("Sheet1").UsedRange.Columns("A:C").Calculate

程序的核心是算法问题

40、End 属性

参阅应用于示例特性返回一个 **Range** 对象,该对象代表包含源区域的区域尾端的单元格。等同于按键 **End+** 向上键、**End+** 向下键、**End+** 向左键或 **End+** 向右键。**Range** 对象,只读。

expression.End(Direction)

expression 必需。该表达式返回“应用于”列表中的对象之一。

Direction **XlDirection** 类型,必需。所要移动的方向。

XlDirection 可为 **XlDirection** 常量之一。

xlDown

xlToRight

xlToLeft

xlUp

示例

本示例选定包含单元格 **B4** 的区域中 **B** 列顶端的单元格。

Range("B4").End(xlUp).Select

本示例选定包含单元格 **B4** 的区域中第 **4** 行尾端的单元格。

Range("B4").End(xlToRight).Select

本示例将选定区域从单元格 **B4** 延伸至第四行最后一个包含数据的单元格。

Worksheets("Sheet1").Activate

Range("B4", Range("B4").End(xlToRight)).Select

41、应用于 CellFormat 和 Range 对象的 Locked 属性。

本示例解除对 **Sheet1** 中 **A1:G37** 区域单元格的锁定,以便当该工作表受保护时也可对这些单元格进行修改。

Worksheets("Sheet1").Range("A1:G37").Locked = False

Worksheets("Sheet1").Protect

42、Next 属性

参阅应用于示例特性返回一个 Chart、Range 或 Worksheet 对象，该对象代表下一个工作表或单元格。只读。

说明

如果指定对象为区域，则本属性的作用是仿效 Tab，但本属性只是返回下一单元格，并不选定它。

在处于保护状态的工作表中，本属性返回下一个未锁定单元格。在未保护的工作表中，本属性总是返回紧靠指定单元格右边的单元格。

示例

本示例选定 sheet1 中下一个未锁定单元格。如果 sheet1 未保护，选定的单元格将是紧靠活动单元格右边的单元格。

Worksheets("Sheet1").Activate

ActiveCell.Next.Select

43、想通过 target 来设置(A1: A10)区域内有改动，就发生此事件。不知道如何

if target.row = 1 and target.column <=10 then

Sub 列举菜单项()

Dim r, s, i As Integer

r = 1

For i = 1 To CommandBars.Count

 ActiveSheet.Cells(r, 1) = "CommandBars(" & i & ").Name:" & CommandBars(i).Name

 r = r + 1

 For s = 1 To CommandBars(i).Controls.Count

 ActiveSheet.Cells(r, 1) = s & "、" & CommandBars(i).Controls(s).Caption

 r = r + 1

 Next

Next

End Sub

44、本示例设置 Microsoft Excel 每当打开包含链接的文件时，询问用户是否更新链接。

Application.AskToUpdateLinks = True

45、自定义函数

Public Function Now1()

Dim string1 As String

 string1 = VBA.Date

 Now1 = string1

End Function

46、复制

Sub copy1()

Sheet2.Range("C5:C10").Copy Sheet1.Range("C5:C10")

End Sub

47、如何统计表中 sheet 的个数？

msgbox sheets.count

Columns("G:G").Select

48、 Selection.EntireColumn.Hidden = True

这样隐藏有个毛病，如何解决？如果 A1:G1 单元格合并的话，就把 A: G 列均隐藏了。

Columns("G:G").EntireColumn.Hidden = True

49、在 VBA 中引用 excel 函数的方法

1). Worksheets("Sheet1").Range("A1").Formula = "=\$A\$4+\$A\$10"

2). Sheet1.Cells(1,1).Formula = "=" & Sheets(iii).Name & "!R1C4"

在宏中用 R1C1 方式写时表格 1 的 A1 中会在写为"=Sheet2!\$D\$1"

用这种方式,想用什么函数就用什么函数.

50、选定下(上)一个工作表

sheets(activsheet.index-1).select

sheets(activsheet.index+1).select

51、Private Sub Workbook_Open()

ActiveWindow.DisplayWorkbookTabs = False '取消工作表标签

Application.CommandBars("Sheet").Controls(1).Enabled = False '格式__工作表不能重命名

Application.CommandBars.FindControl(ID:=889).Enabled = False '右键菜单不能重命名

End Sub

52、[a65536].End(xlUp'A 列从下往上第一个非空的单元格

53、Sub macro()

Set rng = Range("C11:F13") 定义 RNG 为一个单元格区域

For Each cel In rng 定义 CEL 为 RNG 中的一个任一单元格

colo = cel.Interior.ColorIndex 定义 COLO 为单元格 CEL 的填充颜色

If colo <> -4142 Then 如果 COLO 的值不等于-4142

iR = [b65536].End(xlUp).Row + 1 IR 等于 B 列数据区域的行数+1

If [a65535].End(xlUp).Value <> Cells(cel.Row, 2) Then Cells(iR, 1) = Cells(cel.Row, 2)

如果 A 列最后一个非空值单元格 不等于 Cells(cel.Row, 2) 的值 那么单元格 Cells(iR, 1) 的值等于 Cells(cel.Row, 2) 的值
CEL.ROW 是 C11:F13 中任意单元格的行号

Cells(iR, 2) = Cells(10, cel.Column)

Cells(iR, 3) = cel.Value

Cells(iR, 4) = IIf(colo = 36, "Yellow", "Red") Cells(iR, 4) 的值如果 colo = 36 那么值为"Yellow",否则值为"RED"

next

End Sub

54、Private Sub Workbook_SheetSelectionChange(ByVal Sh As Object, ByVal Target As Range)

*****运行数据日志记录*****

Dim rng As Range

If ActiveSheet.Name <> "主界面" And ActiveSheet.Name <> "目录索引" Then

For Each rng In Target.Cells

ChangeCell = ActiveSheet.Name & ", 单元格: " & rng.Address(0, 0) & ", 更改为: " & rng.value & ". 更改时间: "

& Now

CritOrAddtext

Next

End If

End Sub

55、ActiveSheet.Unprotect '撤销当前工作表保护

If ActiveSheet.Name <> "主界面" And ActiveSheet.Name <> "目录索引" And Target.Row > 3 Then '行变色

On Error Resume Next

[ChangColor_With].FormatConditions.Delete

Target.EntireRow.Name = "ChangColor_With"

With [ChangColor_With].FormatConditions

.Delete


```
.Add xlExpression, , "TRUE"  
.Item(1).Interior.ColorIndex = 4
```

End With

End If

ActiveSheet.Protect

56、在 C1 中弄个下拉无表,实际是有效性,你可以选择 A1:A4 为 C1 单元格有效性的序列数据源,如果说 C1 不与 A1:A4 在同一表,则不能这么用,应当先对 A1:A4 命名,然后把数据源改为名称。

57、自动增加工作表

进入宏命令编辑窗口,在 Sub 自动增加工作表()命令后依次键入如下宏命令内容:

```
Dim i&, userinto  
i = 0  
userinto = InputBox("输入插入工作表数量: ")  
If IsNumeric(userinto) = True Then  
Do Until i = userinto  
Worksheets.Add  
i = i + 1  
Loop  
End If  
End Sub
```

58、方法一（共享级锁定）：

- 1、先对 EXCEL 文件进行一般的 VBAProject“工程密码保护。
 - 2、打开要保护的文件,选择: 工具--->保护--->保护并共享工作簿--->以追踪修订方式共享-->输入密码-->保存文件。
- 完成后,当你打开“VBAProject”工程属性时,就将会提示:“工程不可看!”

方法二（推荐,破坏型锁定）:

用 16 进制编辑工具,如 WinHex、Ultraedit-32（可到此下载）等,再厉害点的人完全可以用 debug 命令来做.....用以上软件打开 EXCEL 文件,查找定位以下地方:

ID="{00000000-0000-0000-0000-000000000000}" 注:实际显示不会全部为 0

此时,你只要将其中的字节随便修改一下即可。保存再打开,就会发现大功告成!

当然,在修改前最好做好你的文档备份。至于恢复只要将改动过的地方还原即可（只要你记住了呵呵）。

顺便说一句,这种方法仍然是可破解的,因为加密总是相对的。

59、Sub AddComments()

```
'自動對 ActiveSheet 所有有公式格位加上註解  
Set RG = Cells.SpecialCells(xlCellTypeFormulas)  
For Each c In RG  
c.AddComment  
c.Comment.Text Text:=c.Formula  
Next c  
End Sub  
Sub De_Comments()  
'自動消除所有註解  
Set RG = Cells.SpecialCells(xlCellTypeFormulas)  
For Each c In RG  
c.ClearComments  
Next c
```

End Sub

60、如何在 Excel 里使用定时器

www.aspsky.net 2002-3-12 20:53:27 动网先锋

用过 Excel 97 里的加载宏 "定时保存" 吗? 可惜它的源程序是加密的, 现在就上传一篇介绍实现它的文档。

在 Office 里有个方法是 application.ontime , 具体函数如下:

expression.OnTime(EarliestTime, Procedure, LatestTime, Schedule)

如果想进一步了解, 请参阅 Excel 的帮助。

这个函数是用来安排一个过程在将来的特定时间运行, (可为某个日期的指定时间, 也可为指定的时间段之后)。通过这
个函数我们就可以在 Excel 里编写自己的定时程序了。下面就举两个例子来说明它。

1.在下午 17:00:00 的时候显示一个对话框。

Sub Run_it()

Application.OnTime TimeValue("17:00:00"), "Show_my_msg"

'设置定时器在 17:00:00 激活, 激活后运行 Show_my_msg 。

End Sub

Sub Show_my_msg()

msg = MsgBox("现在是 17:00:00 ! ", vbInformation, "自定义信息")

End Sub

2.模仿 Excel 97 里的 "自动保存宏",在这里定时 5 秒出现一次

Sub auto_open()

MsgBox "欢迎你, 在这篇文档里, 每 5 秒出现一次保存的提示! ", vbInformation, "请注意! "

Call runtimer '打开文档时自动运行

End Sub

Sub runtimer()

Application.OnTime Now + TimeValue("00:00:05"), "saveit"

' Now + TimeValue("00:15:00") 指定在当前时间过 5 秒钟开始运行 Saveit 这个过程。

End Sub

Sub SaveIt()

msg = MsgBox("朋友, 你已经工作很久了, 现在就存盘吗? " & Chr(13) _

& "选择是: 立刻存盘" & Chr(13) _

& "选择否: 暂不存盘" & Chr(13) _

& "选择取消: 不再出现这个提示", vbYesNoCancel + 64, "休息一会吧! ")

'提示用户保存当前活动文档。

If msg = vbYes Then ActiveWorkbook.Save Else If msg = vbCancel Then Exit Sub

Call runtimer '如果用户没有选择取消就再次调用 Runtimer

End Sub

以上只是两个简单的例子, 有兴趣的话, 可以利用 Application.OnTime 这个函数写出更多更有用的定时程序。

Sub Show_my_msg()

msg = MsgBox("现在是 17:00:00 ! ", vbInformation, "自定义信息")

End Sub

2.模仿 Excel 97 里的 "自动保存宏",在这里定时 5 秒出现一次

Sub auto_open()

MsgBox "欢迎你, 在这篇文档里, 每 5 秒出现一次保存的提示! ", vbInformation, "请注意! "

Call runtimer '打开文档时自动运行

End Sub

```
Sub runtime()
Application.OnTime Now + TimeValue("00:00:05"), "saveit"
' Now + TimeValue("00:15:00") 指定在当前时间过 5 秒钟开始运行 Saveit 这个过程。
End Sub
```

接上面

61、Excel 最重要的应用就是利用公式进行计算。无论输入是纯粹的数字运算，还是引用其他单元格计算，只要在一个单元格中输入公式，就能得到结果。这个直接显示结果的设计对于绝大多数场合来说都是适用的，但某些情况下就不那么让人满意了。比如说在做工程施工的预结算编写，使用 Excel，既要写出工程量的计算式，也要看到它的结果，于是这样相同的公式在 Excel 里面要填两次，一次在文本格式的单元格中输入公式，一次是在数据格式的单元格中输入公式让 Excel 计算结果。如何既能看到公式又能看到结果呢？这个问题笔者认为可以从两个方面考虑：一种方法是所谓“已知结果，显示公式”，先在数据格式单元格中输入公式让 Excel 计算结果，然后在相邻的单元格中看到公式；另一种方法所谓“已知公式，显示结果”，就是先在一个文本格式的单元格中输入公式，在相邻的单元格中看到结果。★已知结果，显示公式

假设 C 列为通过公式计算得到的结果（假设 C1 为“=A1+B1”，或者直接是数字运算“=2+3”），而相邻的 D 列是你需要显示公式的地方（即 D1 应该显示为“=A1+B1”或者“=2+3”）。

1. 打开“工具”菜单选择“选项”命令，出现“选项”对话框。
2. 在“常规”选项卡中，选中“R1C1 引用方式”选项。
3. 定义名称，将“引用位置”由“=GET.CELL(6,Sheet1!RC[-1])”即可。这里的 RC[-1]含义是如果在当前单元格的同行前一列单元格中有公式结果，则在当前单元格中得到公式内容，即在含公式结果单元格的同行后一列单元格显示公式内容；如果将 RC[-1]改为 RC[1]，则在公式结果的同行前一列单元格显示公式内容。
4. 如果“引用位置”中含有“RC[-1]”，则在含公式结果单元格的同行后一列单元格中输入“=FormulaofResult”即可得到公式；如果“引用位置”中含有“RC[1]”，则在含公式结果单元格的同行前一列单元格中输入“=FormulaofResult”即可得到公式。

提示：如果想要在含公式结果单元格的同行后数第 2 列中显示公式内容，则需要把“引用位置”中的“RC -1”改为“RC -2”。

★已知公式，显示结果

假设 C 列为输入的没有等号公式（假设 C1 为“A1+B1”），而相邻的 D 列是你需要存放公式计算结果的地方（即 D1 显示 A1 和 B1 单元格相加的结果）。

1. 选中 D1，然后打开“插入”菜单选择“名称”命令中的“定义”子命令，出现“定义名称”对话框。
2. 在“在当前工作表中的名称”输入栏中输入定义的名称“ResultofFomula”，在下方的“引用位置”编辑栏中输入“=EVALUATE(Sheet1!C1)”，单击[确认]按钮退出。
3. 在 D1 中输入“=ResultofFomula”，然后选中按右下角的填充柄向下拉动填充即可。

提示：EVALUATE 是 Excel 4.0 版的宏表函数，Excel 2000 和 Excel 2002 中还支持，但只可用于名称定义中。

4. 填充后要按[F9]进行重算，如果 C 列的公式有改动，也需要及时按[F9]进行重算。

巧施妙计，就能让公式和结果在 Excel 中和平共处了，你也试试吧。

62、http://www.yesky.com/20010724/190094_3.shtml Microsoft Excel 是一款功能非常强大的电子表格软件。它可以轻松地完成数据的各类数学运算，并用各种二维或三维图形形象地表示出来，从而大大简化了数据的处理工作。但若仅利用 Excel 的常用功能来处理较复杂的数据，可能仍需进行大量的人工操作。但 Excel 的强大远远超过人们的想象--宏的引入使其具有了无限的扩展性，因而可以很好地解决复杂数据的处理问题。

随着支持 Windows 的应用程序的不断增多和功能的不断增强，越来越多的程序增加了宏处理来方便用户的自由扩展。但初期各应用程序所采用的宏语言并不统一，这样用户每使用一种应用程序时都得重新学习一种宏语言。为了统一各种应用程序下的宏，Microsoft 推出了 VBA（Visual Basic for Applications）语言。

VBA 是从流行的 Visual Basic 编程语言中派生出来的一种面向应用程序的语言，它适用于各种 Windows 应用程序，可

以解决各应用程序的宏语言不统一的问题。除此之外，使用 VBA 语言还有如下优点：1、VBA 是一种通用程序语言，通过它不仅可以共享 Microsoft 相关的各种软件（如 Excel、Word、Access）.....，而且随着其它的一些软件（如大名鼎鼎的 AutoCAD2000）等对 VBA 的支持，这些软件也已进入到了 VBA 的控制范围；2、可以将用 VBA 编写的程序复制到 Visual Basic 中调试并运行，从而实现用 Visual Basic 来控制有关的应用程序；3、VBA 提供的大量内部函数大大简化了用户的操作。

对于而今的宏，不仅语言统一规范，而且其功能也已非常强大。但在大多数介绍 Excel 的"傻瓜书"、"指南"、"入门与提高"等参考书中往往略过不提，或浅浅带过，读者从中获得的有关知识往往不足以应付处理复杂数据的需求。为了完成工作，就让我们一起来学习"宏"的妙用吧。

一)、宏的自学

首先需要明确的是，本文不可能教会您关于宏的所有内容。您需要学会利用"录制宏"的方法来学习宏：点击 Excel"工具"下拉菜单中"宏"下"录制新宏"，此后可象平时一样进行有关操作，待完成后停止录制。然后再点击"工具"下拉菜单中"宏"下"宏"的"编辑"选项即可打开刚才所录制的宏的 Visual Basic 源程序，并且可以在此时的"帮助"下拉菜单中获得有关的编程帮助。对录制宏进行修改不仅可以学习宏的使用，还能大大简化宏的编写。

二)、基本概念

为了学习 Excel 中的宏，我们需要先了解以下一些基本概念。

1、工作簿：Workbooks、Workbook、ActiveWorkbook、ThisWorkbook

Workbooks 集合包含 Excel 中所有当前打开的 Excel 工作簿，亦即所有打开的 Excel 文件；Workbook 对应 Workbooks 中的成员，即其中的 Excel 文件；ActiveWorkbook 代表当前处于活动状态的工作簿，即当前显示的 Excel 文件；ThisWorkbook 代表其中有 Visual Basic 代码正在运行的工作簿。

在具体使用中可用 Workbooks(index)来引用 Workbook 对象，其中 index 为工作簿名称或编号；如 Workbooks(1)、Workbooks("年度报表.xls")。而编号按照创建或打开工作簿的顺序来确定，第一个打开的工作簿编号为 1，第二个打开的工作簿为 2.....。

2、工作表：Worksheets、Worksheet、ActiveSheet

Worksheets 集合包含工作簿中的所有的工作表，即一个 Excel 文件中的所有数据表页；而 Worksheet 则代表其中的一个工作表；ActiveSheet 代表当前处于的活动状态工作表，即当前显示的一个工作表。

可用 Worksheets(index)来引用 Worksheet 对象，其中 index 为工作表名称或索引号；如 Worksheets(1)、Worksheets("第一季度数据")。工作表索引号表明该工作表在工作表标签中的位置：第一个（最左边的）工作表的索引号为 1，最后一个（最右边的）为 Worksheets.Count。需要注意的是：在使用过程中 Excel 会自动重排工作表索引号，保持按照其在工作表标签中的从左至右排列，工作表的索引号递增。因此，由于可能进行的工作表添加或删除，工作表索引号不一定始终保持不变。

3、图表：Chart、Charts、ChartObject、ChartObjects、ActiveChart

Chart 代表工作簿中的图表。该图表既可为嵌入式图表（包含在 ChartObject 中），也可为一个分开的（单独的）图表工作表。

Charts 代表指定工作簿或活动工作簿中所有图表工作表的集合，但不包括嵌入式在工作表或对话框编辑表中的图表。使用 Charts(index) 可引用单个 Chart 图表，其中 index 是该图表工作表的索引号或名称；如 Charts(1)、Charts("销售图表")。图表工作表的索引号表示图表工作表在工作簿的工作表标签栏上的位置。Charts(1)是工作簿中第一个（最左边的）图表工作表；Charts(Charts.Count)为最后一个（最右边的）图表工作表。

ChartObject 代表工作表中的嵌入式图表，其作用是作为 Chart 对象的容器。利用 ChartObject 可以控制工作表上嵌入式图表的外观和尺寸。

ChartObjects 代表指定的图表工作表、对话框编辑表或工作表上所有嵌入式图表的集合。可由 ChartObjects(index)引用单个 ChartObject，其中 index 为嵌入式图表的编号或名称。如 Worksheets("Sheet1").ChartObjects(1)、Worksheets("sheet1").ChartObjects("chart1")分别对应"Sheet1"工作表中的第一个嵌入式图表、以及名为"Chart1"的嵌入式图表。

ActiveChart 可以引用活动状态下的图表，不论该图表是图表工作表，或嵌入式图表。而对于图表工作表为活动工作表时，还可以通过 ActiveSheet 属性引用之。

4、单元格：Cells、ActiveCell、Range、Areas

`Cells(row,column)`代表单个单元格，其中 `row` 为行号，`column` 为列号。如可以用 `Cells(1,1)`、`Cells(10,4)`来引用"A1"、"D10" 单元格。`ActiveCell` 代表活动工作表的活动单元格，或指定工作表的活动单元格。

`Range` 代表工作表中的某一单元格、某一行、某一列、某一选定区域（该选定区域可包含一个或若干连续单元格区域）或者某一三维区域。

可用 `Range(arg)`来引用单元格或单元格区域，其中 `arg` 可为单元格号、单元格号范围、单元格区域名称。如 `Range("A5")`、`Range("A1:H8")`、`Range("Criteria")`。虽然可用 `Range("A1")`返回单元格 A1，但用 `Cells` 更方便，因为此时可用变量指定行和列。

可将 `Range` 与 `Cells` 结合起来使用，如 `Range(Cells(1,1),Cells(10,10))` 代表单元格区域 "A1:J10"；而 `expression.Cells(row,column)`返回单元格区域中的一部分，其中 `expression` 是返回 `Range` 的表达式，`row` 和 `column` 为相对于该区域的左上角偏移量。如由 `Range("C5:C10").Cells(1,1)`引用单元格 C5。

`Areas` 为选定区域内的连续单元格块的集合，其成员是 `Range` 对象。而其中的每个 `Range` 对象代表选定区域内与其它部分相分离的一个连续单元格块。某些操作不能在选定区域内的多个单元格块上同时执行；必须在选定区域内的单元格块数 `Areas.Count` 上循环，对每个单独的单元格块分别执行该操作。此时，可用 `Areas(index)`从集合中返回单个 `Range` 对象，其中 `index` 为单元格块编号；如 `Areas(1)`。

5、行与列：Rows、Columns、Row、Column

`Rows`、`Columns` 分别代表活动工作表、单元格区域范围 `Range`、指定工作表中的所有行数、列数。对于一个多选单元格区域范围 `Range` 的 `Rows`、`Columns`，只返回该范围中第一个区域的行数、列数。例如，如果 `Range` 对象有两个区域（`areas`）A1:B2 和 C3:D4，`Rows.Count` 返回 2 而不是 4。

可通过 `Rows(行号)`、`Columns(列号)`来引用相应的行与列；如 `Rows(3)`、`Columns(4)`分别对应第三行、D 列。利用 `Rows`、`Column` 可以获得区域中第一块的第一行行号、第一列列号，所得值均以十进制数表示。

三)、处理单元格

1、直接赋值与引用

将变量、常量值直接赋给单元格、或将单元格的值直接赋给变量、常量，这是在 `Excel` 中最简单的单元格赋值及引用方法。如下例将工作表"Sheet1"A1 单元格的值赋给 `Integer` 变量 `I`，并将 `I+1` 的值赋给当前工作表中的 B1 单元格：

```
Dim I As Integer
```

```
I=Worksheets("Sheet1").Cells(1,1)
```

```
Cells(1,2).Select '选定 B1 单元格，使其成为当前单元格
```

```
ActiveCell=I+1 '以 I+1 为当前单元格赋值
```

2、用公式赋值

在宏的使用中，可能会更多地用公式来给单元格赋值。如下例将相对于活动单元格左侧第 4 列、向上第 6 行至向上第 2 行的单元格数值之和赋给活动单元格（以本行、本列为第 0 行、0 列）：

```
ActiveCell.Formula="=AVERAGE(R[-6]C[-4]:R[-2]C[-4])"
```

3、引用其它工作表中的单元格

当赋值公式中需要引用其它工作表中的单元格时，在被引用的单元格前加上"工作表名!"即可。如以下即在赋值中引用了"Sheet1"工作表中的 A1 至 A4 单元格：

```
Range("E10").Formula="=SUM(Sheet1!R1C1:R4C1)"
```

但需注意的是：当被引用的工作表名中含有某些可能引起公式歧义的字符时，需要用单引号'将工作表名括起来。如：

```
Worksheets("Sheet1").ActiveCell.Formula="=Max('1-1 剖面'!D3:D5)"
```

4、引用其它工作簿中的单元格

在被引用单元格所在工作表名前加上"[工作簿名]"，即可引用其它工作簿中的单元格。如：

```
ActiveCell.Formula="=MAX([Book1.xls]Sheet3!R1C:RC[4])"
```

同样需注意的是：当被引用的工作簿名中含有某些可能引起公式歧义的字符时，需要用中括号"[]"及单引号'将工作簿名括起来。如：

```
Cells(1,2).Formula="=MIN("[1995-2000 总结.xls]1995-1996 年"! $A$1:$A$6)"
```


5、避免循环引用

在上述公式赋值过程中，应避免在公式中引用被赋值的单元格，防止循环引用错误。

6、添加批注

可按如下方法给单元格添加批注：

Dim 批注文本 As String

批注文本="批注示例" '准备批注文本

ActiveCell.AddComment '添加批注

ActiveCell.Comment.Text Text:=临时 '写入批注文本

ActiveCell.Comment.Visible=False '隐藏批注

7、添加、删除、复制、剪切、粘贴单元格

Range("D10").Insert Shift:=xlToRight '在 D10 单元格处添加一新单元格，原 D10 格右移

Range("C2").Insert Shift:=xlDown '在 C2 单元格处添加一新单元格，原 C2 格下移

Rows(2).EntireRow.Insert '在第 2 行前添加一空白行，原第 2 行下移

Columns(3).EntireColumn.Insert '在 C 列前添加一空白列，原 C 列右移

Columns("A:D").Delete Shift:=xlToLeft '删除 A 列至 D 列，其右侧列左移

Rows("3:5").Delete Shift:=xlUp '删除第 3 行至第 5 行，其下方行上移

Range("B2").EntireRow.Delete '删除第 2 行

Range("C4").EntireColumn.Delete '删除 C 列

Range("B10:C13").Copy '复制 B10 至 C13 单元格区域

Cells(1,2).Cut '剪切 B1 单元格

Range("D10").Select

ActiveSheet.Paste '自 D10 单元格起粘贴剪贴板中的内容

四)、图表

1、工作表图表

以下为一添加工作表图表的实例。

Charts.Add after:=Worksheets("Sheet1") '在"Sheet1"工作表之后添加新图表工作表

ActiveChart.ChartType:=xlXYScatterSmooth '图表类型为 XY 平滑线散点图

ActiveChart.SetSourceData Source:=Worksheets("结点坐标").Range("A1:B69"), PlotBy:= _
xlColumns '图表数据来源于"结点坐标"工作表的 A1 至 B69 单元格，且按列绘图。

ActiveChart.Location Where:=xlLocationAsNewSheet

With ActiveChart

.HasTitle = True

.ChartTitle.Characters.Text = "节点坐标" '图表标题"节点坐标"

.Axes(xlCategory, xlPrimary).HasTitle = True

.Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "x" 'x 轴标题"x"

.Axes(xlValue, xlPrimary).HasTitle = True

.Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "y" 'y 轴标题"y"

End With

With ActiveChart.Axes(xlCategory)

.HasMajorGridlines = True '显示 x 轴主网格线，默认情况下为显示

.HasMinorGridlines = True '显示 x 轴次网格线，默认情况下为不显示

End With

With ActiveChart.Axes(xlValue)

.HasMajorGridlines = True '标出 x 轴主网格值，默认情况下为标注

.HasMinorGridlines = False '取消 x 轴次网格值标注，默认情况下为不标注

End With

ActiveChart.Legend.Position = xlRight '图例显示在图表右侧

2、嵌入式图表

嵌入式图表仅在添加方式及引用格式上与工作表图表有所不同，而对图表的设置基本类似。详见下例。

Set 嵌入表=ActiveSheet.ChartObjects.Add(0,0,200,300) '在当前工作表(0,0)坐标处添加宽 200，高 300 的嵌入式图表

嵌入表.Chart.ChartType = xlColumnClustered '图表类型为簇状柱形图

嵌入表.Chart.SetSourceData Source:=Sheets(1).Range("A2:B2"), PlotBy:=xlRows '设置图表数据来源

With 嵌入表.Chart

.HasTitle = False '无图表标题

.Axes(xlCategory, xlPrimary).HasTitle = False '无 x 轴标题

.Axes(xlValue, xlPrimary).HasTitle = False '无 y 轴标题

End With

五)、工作表

1、添加

Sheets.Add before:=Sheets(1) '在第 1 工作表前添加新工作表

Sheets.Add after:=Sheets(Sheets.Count) '在最后工作表后添加新工作表

2、移动

ActiveSheet.Move before:=Sheets(2) '将当前工作表移动至第 2 工作表之前

3、命名

ActiveSheet.Name="工作表名" '将当前工作表命名为"工作表名"

4、删除

可以用以下语句删除当前工作表。

ActiveSheet.Delete

但在删除前 Excel 会自动弹出提示框，需在用户确认后方可执行删除。为避免这一干扰，可以先用以下语句关闭 Excel 的警告提示。

Application.DisplayAlerts = False

在删除完成后，再重新打开 Excel 的警告提示

Application.DisplayAlerts = True

六)、工作簿

Excel 的宏对工作簿的操作主要为保存。

Dim 存盘文件名 As String

ActiveWorkbook.Save '保存当前工作簿

存盘文件名="工作表名"

ActiveWorkbook.SaveAs Filename:= 存盘文件名 '当前工作簿另存为"工作表名.xls"

在另存时，若指定的存盘文件名不包含路径，则保存在该工作簿的打开目录下。而若此存盘文件已存在，也可用关闭 Excel 警告提示的方法以免其自动弹出提示框。

63、用 VBA 的几个误区

1、想通过 Excel+vba 来编程发财，不管怎么说，目前这种可能性很小。

2、录制宏后的代码不进行简化，这种情况初学者常犯；或者不屑于用录制宏，这种相反。

3、喜欢用.select、.copy、.active 等，如果不是迫不得已，还是不要用这些东西好。

4、不喜欢用公式、函数，什么都喜欢用程序；活用公式函数是我们学 EXCEL 中重要的重点，学会嵌套着用，很多事情都可以解决。譬如表 1 的 A1 单元格如果要等于表 2 的 A 列中的某个单元格，很多人都是用程序来直接赋值，数量少这也可以，但如果数量很多的话，为何不试用一下 offset 呢？偏移都指向同一个单元格，就当是一个变量，改一下这个单元格的值，表

1 用 offset 的单元格的值就全部改了。

5、设计输入数据的界面竟然用用户窗口+文本框之类的东西，天，我为什么要用 EXCEL，原因之一不就是图个方便吗？！直接在工作表中操作就是输入数据最好的方法，如果一些功能你不想给别人用，顶多你屏蔽掉吧。

6、关闭程序的时候很喜欢退出整个 EXCEL，这样自我感觉更象一个完整的程序。其实完全有必要考虑其他用户是否也在用 EXCEL，所以，必须做好一下判断，不要自作聪明全部退出来，还帮忙把每个工作簿都保存一下。

7、不喜欢用 with，总是 workbook("").worksheet("").range("").***一行行排得挺整齐的，还说都用复制，其实也不麻烦。

8、总想把 VBA+EXCEL 做得象一个程序一样，其实，就因为它不独立，所以才更加为我们喜爱，不是吗？

64、用过 VB 的人都应该知道如何声明变量，在 VBA 中声明变量和 VB 中是完全一样的！

使用 Dim 语句

```
Dim a as integer      '声明 A 为整形变量
Dim a                  '声明 A 为变体变量
Dim a as string        '声明 A 为字符串变量
Dim a,b,c as currency  '声明 A,b,c 为货币变量
```

声明变量可以是：Byte、Boolean、Integer、Long、Currency、Single、Double、Decimal（当前不支持）、Date、String（只限变长字符串）、String * length（定长字符串）、Object、Variant、用户定义类型或对象类型。

强制声明变量

Option Explicit

说明：该语句必在任何过程之前出现在模块中。

声明常数，用来代替文字值。

Const

' 常数的默认状态是 Private。

Const My = 456

' 声明 Public 常数。

Public Const MyString = "HELP"

' 声明 Private Integer 常数。

Private Const MyInt As Integer = 5

' 在同一行里声明多个常数。

Const MyStr = "Hello", MyDouble As Double = 3.4567

在 EXCEL97 中，有一个十分好的功能，他就是把鼠标放置在一个有效数据单元格中，执行该段代码，你就可以将连在一起的一片数据全部选中。只要将该段代码加入到你的模块中。

Sub My_Select

Selection.CurrentRegion.Select

End sub

删除当前单元格中数据的前后空格。

sub my_trim

Trim(ActiveCell.Value)

end sub

使单元格位移

sub my_offset

ActiveCell.Offset(0, 1).Select'当前单元格向左移动一格

ActiveCell.Offset(0, -1).Select'当前单元格向右移动一格

ActiveCell.Offset(1, 0).Select'当前单元格向下移动一格

ActiveCell.Offset(-1, 0).Select'当前单元格向上移动一格

end sub

如果上述程序产生错误那是因为单元格不能移动，为了解除上述错误，我们可以往

sub my_offset 之下加一段代码 on error resume next

注意以下代码都不再添加 sub “代码名称” 和 end sub 请自己添加!

给当前单元格赋值:

ActiveCell.Value = "你好!!!"

给特定单元格加入一段代码:

例如: 在 A 1 单元格中插入"HELLO"

Range("a1").value="hello"

又如: 你当前的工作簿在 sheet1 上, 你要往 sheet2 的 A 1 单元格中插入"HELLO"

1.sheets("sheet2").select

range("a1").value="hello"

或 2.Sheets("sheet1").Range("a1").Value = "hello"

说明:

1.sheet2 被打开, 然后在将"HELLO"放入到 A1 单元格中。

2.sheet2 不被打开, 将"HELLO"放入到 A1 单元格中。

隐藏工作表

'隐藏 SHEET1 这张工作表

sheets("sheet1").Visible=False

'显示 SHEET1 这张工作表

sheets("sheet1").Visible=True

有时候我们想把所有的 EXCEL 中的 SHEET 都打印预览, 请使用该段代码, 它将在你现有的工作簿中循环, 直到最后一个工作簿结束循环预览。

Dim my As Worksheet

For Each my In Worksheets

my.PrintPreview

Next my

得到当前单元格的地址

msgbox ActiveCell.Address

得到当前日期及时间

msgbox date & chr(13) & time

保护工作簿

ActiveSheet.Protect

取消保护工作簿

ActiveSheet.Unprotect

给当前工作簿改名为 "liu"

ActiveSheet.Name = "liu"

打开一个应用程序

AppActivate (Shell("C:\WINDOWS\CALC.EXE"))

增加一个工作簿

Worksheets.Add

删除当前工作簿

activesheet.delete

打开一个文件

Workbooks.Open FileName:="C:\My Documents\Book2.xls"

关闭当前工作簿

ActiveWindow.Close

当前单元格定为：左对齐

Selection.HorizontalAlignment = xlLeft

当前单元格定为：中心对齐

Selection.HorizontalAlignment = xlCenter

当前单元格定为：右对齐

Selection.HorizontalAlignment = xlRight

当前单元格为百分号风格

Selection.Style = "Percent"

当前单元格字体为粗体

Selection.Font.Bold = True

当前单元格字体为斜体

Selection.Font.Italic = True

当前单元格字体为宋体 20 号字

With Selection.Font

.Name = "宋体"

.Size = 20

End With

With 语句

With 对象

.描述

End With

让你的机器发出响声

BEEP

清除单元格中所有文字、批注、格式、所有的东西！

ActiveCell.Clear

测试选择状态的单元格的行数

MsgBox Selection.Rows.Count

测试选择状态的单元格的列数

MsgBox Selection.Columns.Count

测试选择状态的单元格的地址

Selection.Address

让所有的错误不再发生

ON ERROR RESUME NEXT

产生错误时让错误转到另一个地方

on error goto l

'code

l:

'code

删除一个文件

kill "c:\1.txt"

定制自己的状态栏

Application.StatusBar = "现在时刻: " & Time

恢复自己的状态栏

```
Application.StatusBar = false
```

在运行期执行一个宏

```
Application.Run macro:="text"
```

滚动窗口到 a1 的位置

```
ActiveWindow.ScrollRow = 1
```

```
ActiveWindow.ScrollColumn = 1
```

定制系统日期

```
Dim MyDate, MyDay
```

```
MyDate = #12/12/69#
```

```
MyDay = Day(MyDate)
```

今天的年限

```
Dim MyDate, MyYear
```

```
MyDate = Date
```

```
MyYear = Year(MyDate)
```

```
MsgBox MyYear
```

产生一个 inputbox<输入框>

```
InputBox ("Enter number of months to add")
```

得到一个文件名:

```
Dim kk As String
```

```
kk = Application.GetOpenFilename("EXCEL (*.XLS), *.XLS", Title:="提示: 请打开一个 EXCEL 文件: ")
```

```
msgbox kk
```

打开 zoom 显示比例对话框

```
Application.Dialogs(xlDialogZoom).Show
```

激活字体对话框

```
Application.Dialogs(xlDialogActiveCellFont).Show
```

打开另存对话框

```
Dim kk As String
```

```
kk = Application.GetSaveAsFilename("excel (*.xls), *.xls")
```

```
Workbooks.Open kk
```

此段代码寻找字符串中特定字符的位置，需要建立一个窗体，并在窗体中，放入 TEXTBOX1,TEXTBOX2 和 TEXTBOX3(3 个文本框) 及 COMMANDBUTTON1(按钮)

```
Private Sub CommandButton1_Click()
```

```
a = TextBox1.Text
```

```
b = TextBox2.Text
```

```
aa = Len(a)
```

```
i = 1
```

```
Dim YY As String
```

```
If b = "" Then Exit Sub
```

```
Do
```

```
If InStr(i, a, b, vbTextCompare) = 0 Then Exit Do
```

```
kk = InStr(i, a, b, vbTextCompare)
```

```
YY = YY & CStr(kk) & "/"
```

```
i = kk + 1
```

```
Loop While aa >= i
TextBox3.Text = YY
End Sub
```

ActiveX 控件在 Excel97 中的运用

Excel97 在工作表或图表上可使用 ActiveX 控件, 根据我使用的体会, 在工作上处理控件时, 必须注意和了解如下事项:

(一)用 Excel5.0/95 工作簿文件格式保存 Excel97 工作簿时, 将选择 ActiveX 控件信息。当用户通过双击鼠标来编辑内嵌在其它应用程序文档中的 Excel97 工作簿时, 该工作簿上的控件将不会正常工作。如果用户是通过用右键单击工作簿, 然后选中快捷菜单上的“打开”命令来编辑工作簿的话, 工作簿上的控件就能正常工作了。

(二)当 ActiveX 控件处于激活状态时, 将禁用某些 MicrosoftExcelVisualBasic 方法和属性。例如, 当某一控件激活时, 就不能使用 Sort 方法, 故下述按钮单击事件处理过程中的代码将失败(因为用户单击按钮后, 该按钮就处于激活状态)。

```
PrivateSubCommandButton1 Click
Range("a1:a10") SortKey1:=Range("a1")
EndSub
```

解决办法是通过选激活工作表上其它元素的方法来绕过这种问题。例如, 可用下列代码对单元格区域排序:

```
PrivateSubCommandButton1 Click
Range("a1") Activate
Range("a1:a10") SortKey1:=Range("a1")
CommandButton1 Activate
End Sub
```

(三)在 MicrosoftExcel 中, 用 OLEObjects 集合中的 OLEObject 对象代表 ActiveX 控件。如果要用编程的方式向工作表添加 ActiveX 控件, 可用 OLEObjects 集合的 Add 方法。例如向第一张工作表添加命令按钮。

```
Worksheets(1) OLEObjects Add"Forms CommandButton 1", _
Left:=10,Top:=10,Height:=20,Width:=100
```

因为 ActiveX 控件也可用 OLEObjects 集合中的 OLEObject 对象代表, 所以也可用该集合中的对象来设置控件的属性。例如要设置控件“CommandBotton1”的“左边位置”属性。

```
Worksheets(1) OLEObjects("CommandButton1") Left=10
```

那些不属于 OLEObject 对象属性的控件属性, 可通过由 Object 属性返回的实际控件对象来设置。例如要设置控件“CommandButton1”的标题。

```
Worksheets(1) OLEObjects("CommandButton1")
Object Caption="runme"
```

因为所有的 OLE 对象也是 Shapes 集合的成员, 所以也可用该集合设置若干控件的属性。例如要对齐第一张工作表上所有控件的左边框。

```
ForEachsInWorksheets(1) Shapes
Ifs Type=msoOLEControlObjectThens Left=10
Next
```

请注意, 当在控件所在工作表的类模块之外使用控件的名称时, 必须用工作表的名称限定该控件的名称。在工作表上 ActiveX 控件的事件处理过程中, Me 关键字所指向的是工作表, 而非控件

65、在 Excel 中利用 VBA 创建多级选单

Excel 是我们常用的报表处理软件之一, 对于大多数人来说只是使用它进行打印报表, 没有注意其他功能, 其实利用 Excel 内嵌的 VBA 语言完全可以快速开发出自己企业的应用系统来, 而且应用系统界面与其它专业编程语言相当相似。下面笔者简单通过一个实例说明如何利用 VBA 创建多级选单。

首先, 我们对 Excel 中两个重要的内置函数 auto_open() 和 auto_close()作一简单说明。

auto_open() : 在打开工作簿时系统将自动执行该函数, 因此我们可以在该函数中调用自己应用程序的选单函数以及其它需要初始化设置的函数及宏语句;

auto_close(): 在关闭工作簿时系统将自动执行该函数。所以我们需要在该函数中放置删除用户自定义选单语句, 否则只有退出 Excel 才能恢复 EXCEL 的系统选单。

在以下语句中, 我们定义了选单设置函数 OpenMyMenu(), 用于设置多级选单, 其他有关说明见程序内注释, 详细代码如下:

```
Sub OpenMyMenu() '自定义多级选单函数
On Error Resume Next '忽略错误
MenuBars("MyMenu").Delete '删除自定义选单
MenuBars.Add ("MyMenu") '自定义选单项
Sheets("sheet1").Select
MenuBars("MyMenu").Menus.Add Caption:="金融"
'增加第一个选单项"金融"
'以下三句为在"金融"选单下增加"银行法、货币政策和条例"三项选单项
MenuBars("MyMenu").Menus("金融").MenuItems.Add Caption:="银行法", OnAction:="银行法"
MenuBars("MyMenu").Menus("金融").MenuItems.Add Caption:="货币政策", OnAction:="货币政策"
MenuBars("MyMenu").Menus("金融").MenuItems.Add Caption:="条例", OnAction:="条例"
'以下为创建如图所示的多级选单
MenuBars("MyMenu").Menus.Add Caption:="经济" '建立选单项"经济"
'以下三句为在"经济"选单下增加"农业、工业和第三产业"三项选单项
MenuBars("MyMenu").Menus("经济").MenuItems.Add Caption:="农业", OnAction:="农业"
MenuBars("MyMenu").Menus("经济").MenuItems.Add Caption:="工业", OnAction:="工业"
MenuBars("MyMenu").Menus("经济").MenuItems.AddMenu Caption:="第三产业"
'以下三句为在"第三产业"选单下增加"概况、范畴"二项选单项和"饮食服务业"子选单
MenuBars("MyMenu").Menus("经济").MenuItems("第三产业").MenuItems.Add Caption:="概况", OnAction:="概况"
MenuBars("MyMenu").Menus("经济").MenuItems("第三产业").MenuItems.Add Caption:="范畴", OnAction:="范畴"
MenuBars("MyMenu").Menus("经济").MenuItems("第三产业").MenuItems.AddMenu Caption:="饮食服务业"
'以下二句为在"饮食服务业"选单下增加"酒店 1、酒店 2"二项选单项
MenuBars("MyMenu").Menus("经济").MenuItems("第三产业").MenuItems("饮食服务业").MenuItems.Add Caption:="酒店 1", OnAction:="酒店 1"
MenuBars("MyMenu").Menus("经济").MenuItems("第三产业").MenuItems("饮食服务业").MenuItems.Add Caption:="酒店 2", OnAction:="酒店 2"
MenuBars("MyMenu").Activate '激活自定义选单
End Sub

Sub auto_open() '系统自动打开运行宏
OpenMyMenu '调用用户选单函数
End Sub

Sub auto_close() '系统自动关闭运行宏
On Error Resume Next '忽略错误
MenuBars("MyMenu").Delete '删除自定义选单
End Sub
```

读者可以在自己的工作簿选单"工具"中的"宏"下, 创建以上三个函数并将以上函数语句拷贝到其中即可运行。

66、.用 VBA(编程)保护 Excel 文档

VBA (Visual Basic for Application) 是 Excel 应用程序中功能非常强大的编程语言, 为了规范不同的用户对 Excel 应用程序的访问能力, 需要对 Excel 文档及有关的数据进行有效的保护, 这里根据自己及同行们的体会, 从以下二个方面介绍用 VBA 编程法实现对 Excel 文档的保护。

对工作簿的保护

1.利用 VBA 中 **Workbook** 对象的 **SaveAs** 方法实现对工作簿的保护，下面就对 **SaveAs** 有关的参量作一介绍：

Filename：该字符串表示要保存的文件名。可包含完整路径。如果不指定路径，**Microsoft Excel** 将文件保存到当前文件夹。

FileFormat：可选，文件的保存格式。

Password：为一个区分大小写的字符串（不超过 15 个字符），用于指定文件的保护密码。

WriteResPassword：该字符串表示文件的写保护密码。如果文件保存时带有密码，但打开文件时不输入密码，则该文件以只读模式打开。

ReadOnlyRecommended：如果为 **True** 则在打开文件时显示一条信息，提示该文件以只读模式打开。

下例就是在 **Excel** 应用程序中添加一工作簿，将工作簿按常规文件格式存为“C:\pj\obj\经济评价.xls”文件，并给该文件指定保护密码“12”以及写保护密码“23”。

Sub 保护工作簿（）

NewWorkbook = Workbooks.add

NewWorkbook.SaveAs FileName: ="C: \pj\obj\经济评价.xls", FileFormat: =
XlNormal,Password:="12",WriteResPassword:="23"

End sub

2.利用 VBA 中 **Workbook** 对象的 **Protect** 方法对工作簿的结构和窗口进行保护，**Workbook** 对象的 **Protect** 方法带有以下三个参量：

Password：为加在工作表或工作簿上区分大小写的密码字符串。如果省略本参数，不用密码就可以取消对该工作簿的保护。否则，取消对该工作表或工作簿的保护时必须提供该密码。如果忘记了密码，就无法取消对该工作表或工作簿的保护。最好在安全的地方保存一份密码及其对应文档名的列表。

Structure：若为 **True** 则保护工作簿结构（工作簿中工作表的相对位置）。默认值为 **False**。

Windows 若为 **True** 则保护工作簿窗口。

下例就是对一名为“经济评价.xls”的工作簿实现“结构”和“窗口”保护：

Sub 保护工作簿（）

Workbooks（“经济评价”）.Protect Password: ="1234", Structure: =True,
Windows:=True

End sub

Workbook 对象的 **Unprotect** 方法用于取消对工作簿的保护。**Unprotect** 方法只有一个参量，就是保护工作表时所用的口令。

3.对工作簿进行隐藏保护，可使他人无法看到其对应的窗口。操作方法如下：

在 VBA 中使用 **Workbook** 对象下面的 **Windows** 对象的 **Visible** 属性对工作簿进行隐藏和取消隐藏，**Visible** 属性的值可取“**True**”和“**False**”两种。

下面程序代码完成对工作簿“book.xls”的隐藏：

Sub 隐藏工作簿（）

Workbooks("book").Activate

ActiveWindow.Visible = False

End sub

或：

Sub 隐藏工作簿（）

Workbooks("book").Windows（1）.Visible=False

End sub

对工作表的保护

1. 对工作表实现口令保护利用 VBA 调用 **Worksheet** 对象的 **Protect** 方法对工作表进行保护，**Protect** 带有以下参量：

Password 用于保护工作表的口令。

Drawingobjects 若为 **True**，则对工作表中的 **Drawingobjects** 对象进行保护，缺省值为 **True**。

Contents 若为 True, 则对单元格内容进行保护, 缺省值为 True。

下面程序代码完成对工作表“基础数据表”的保护:

Sub 保护工作表 ()

Worksheets("基础数据表").Protect Password:="1234"

End sub

2. 对工作表实现隐藏保护, 使他人无法看到工作表:

利用 VBA 设置 Worksheet 对象的 Visible 属性来隐藏工作表: Visible 属性的值为以下三个值中的一个:

True 工作表为显示状态。

False 工作表为隐藏状态。

XlVerHidden 工作表为隐藏状态, 且用户不能通过“取消隐藏”对话框将其改为显示状态。当 Visible 的值为 XlVerHidden 时, 只能利用 VBA 将其重新设置为 True。

Sub 隐藏工作表 ()

Worksheets("基础数据表").Visible=False

End sub

67、求:将所选区域中的数值全部转化为"万元"的最简代码。

求:将所选区域中的数值全部转化为"万元"的最简代码。

或能完成此功能的最便捷的命令操作。

写了一个,抛砖引玉:

Sub convt()

Dim cel As Range

Dim dec As Variant

Application.EnableEvents = False

yesorno = MsgBox("确实将区域所有数值转换为"万元"? ", vbYesNo + vbQuestion + vbDefaultButton1)

If yesorno = vbYes Then

1: dec = Application.InputBox("请输入小数位数:", Default:=0, Type:=1)

If dec = "" Then

GoTo 1

End If

For Each cel In Selection

If IsNumeric(cel.Value) Then

'cel = (Round(cel.Value / 10000, 2)) & "万元"

cel = (Round(cel.Value / 10000, dec))

End If

Next

ElseIf yesorno = vbNo Then

Exit Sub

End If

End Sub

以下当为最简代码吧。

Sub Macro2()

Selection.NumberFormatLocal = "#"".""#,"

End Sub

68、在 VBA 开发环境中, 点击工具——附加控件, 出现如下窗口:

在可选控件中选择 microsoft progressbar control,再点击确定,这时,你会发现你的工具箱中增加了进度条工具,如图:

下面说说进度条的用法:

属性: Max: 设置进度条控件的最大值

Min: 设置进度条控件的最小值

Value: 设置进度条控件对象的当前值

Scrolling: 进度条的样式, 即一格一格的进度条或是没有间隔的进度条。

例子: Private Sub UserForm_Activate()

For i = 1 to 5000

ProgressBar1.Max = 5000 `设置进度条控件的最大值

ProgressBar1.Value = I `进度条控件对象的当前值

next

End Sub

在程序中加入进度条的办法是将 ProgressBar1.Max=最大值, ProgressBar1.Value = 当前值放入循环中去。

69、Excel 是一个优秀的电子表格软件, 如果你编的程序需要以报表的形式显示最终结果, 它是个不错的选择。你可以通过 VB 控制 Excel 显示数据表格。如果你不愿意深入了解 Excel 这些琐碎的细节, 也不是没有权宜之计: 你可以打开 Excel 中的工具→宏→录制新宏, 然后进行手工操作, 结束后把所录的宏代码贴进你的 VB 程序就行。这样得到的程序一般都能正常运行, 但其中的宏代码往往不够简洁, 效率和可读性都不高。

Excel 编程碰到的第一个问题是表头。有时表头的形式比较复杂, 需要横向或纵向合并单元格。请放心, 只要没有斜杠, Excel 都能应付得了。

例如合并 A2~A5 这 4 个单元格, 你录制的宏代码会是这样:

Range("A2:A5").Select

With Selection

.HorizontalAlignment = xlCenter

.VerticalAlignment = xlBottom

.WrapText = False

.Orientation = 0

.AddIndent = False

.ShrinkToFit = False

.MergeCells = False

End With

Selection.Merge

而自己编程只要一句 Range("A2:A5").mergecells=True 就可以解决问题。

表头形式定了, 再就是表头的内容。如果单元格中的文本长度超过了列宽, 往往只能显示部分内容, 行尾那一格的内容则会“越境”进入右边那个空白单元格, 很不美观。这个问题可以通过在程序中设置列宽加以解决。

Columns(14).columnwidth=12 `设置第 14 列列宽为 12 (缺省列宽为 8.38)

如果你不愿意劳神去逐列估计实际所需的列宽, 干脆来一行

Columns("a:i").autofit `a 到 i 列自动调整列宽

让 Excel 随机应变吧。

但也许你不喜欢这种方法, 认为表头撑大了列宽, 弄得浏览一张小表格还得向右滚动, 太不方便了。要是能保持默认列宽, 让文本自动换行就好了。没问题, Excel 包你满意。

Rows(3).WrapText=True `让第三行各单元格中的文本自动换行

不过你最好再加一句 Rows(3).VerticalAlignment = xlTop 让表头自动向上对齐, 这样比较符合习惯。

你还可以给表头打上底色, 让你的读者不至于看了打哈欠。

Rows(2).Interior.ColorIndex = 5 设置第 2 行底色为蓝色

再给表格的标题上色，这样更醒目一点。

Rows(1).Font.ColorIndex=4

表头完成后该填数据了，一个一个地填实在是太？，如果你的数据是存放在一个二维数组中，那问题就简单多了。

Dim Data(3,4)

..... `数据处理

Range("a2:d4").Value=Data

这样可以一次填入一个表的所有数据，够快了吧！不过提醒一句，Range 对象大小最好与数组匹配，小了无法显示所有数据，大了则会在空白单元格只填入"N/A"表示没有取得数据。

如果需要在结果中显示多个同样规格的数据表，想在 Range 对象中加入循环变量，这也好办。

Dim cell1,cell2

Dim Data(3,4)

.....

For I =1 to 40

..... `数据处理

Set cell1=Worksheets("Sheet1").Cells(5*I-4,1)

Set cell2=Worksheets("Sheet1").Cells(5*I-2,4)

Worksheets("Sheet1").Range(cell1,cell2).value=Data

Next I

表格填完了，现在该打表格线了，以下几条语句可以满足你的要求：

With Worksheets("Sheet1").Range(cell1,cell2).borders

.LineStyle=xlContinuous

.weight=xlThin

End With

70、《计算机实用软件》2001-2002 第一学期试卷（2）

一、填空题（每空 1 分，共 20 分）

1.在 Excel 的系统默认状态下，某个工作表的标签呈白色显示，表示其为_____。在工作表中被粗线框住的格子称为_____。

2. 在 Excel 工作表中可以输入两类数据它们是_____和_____

3. 工作表由_____行和_____列组成

4.输入 Excel 公式应以_____或者_____开头，然后再输入具体的计算公式

5.在 Excel 中，系统默认状态下，输入文字的对齐方式是_____，输入数字的对齐方式是_____

6.在 VB5 的标准控件中_____控件适用于在一组按钮中显示多个选项,但用户只能而且必须选择其中一项。_____控件适用于放置用户只能看到但不能修改的少量文本。

7.在 VB 中，要改变 Text1 文本框内显示的信息时，应当在其_____窗口中，将_____属性进行重新设置。

8.VB5 表达(40 Mod 7)\2 的值是_____ 2x2 的 VB5 表达式为_____

9.如下程序实现在当前窗口中显示一个消息框，消息框中显示有“是（Y）”和“否（N）”两个按钮，试将该程序填写完整，使用户点击“是(Y)”按钮后能显示出数字 1，点击“否（N）”按钮后能显示出数字 2

Private Sub Form_Click()

Msg='你想要继续吗？'

Style=vbYesNo

Response=msgbox(msg , _____)

If Response _____ vbYes

Print"1"

Else

End if

End Sub

二、单选题（每题 3 分，共 30 分）

1. Excel 主要用于（ ）领域。
A. 美术设计、装璜 B. 数据分析、财务管理
C. 工业设计、机械制造 D. 多媒体制作
2. 对话框窗口中的“确定”按钮的作用是（ ）
A. 确定输入的信息 B. 确定各选项并开始执行
C. 关闭对话框不做任何事 D. 退出对话框
3. 在个人计算机上，通常 Excel 是在（ ）系统环境下运行的。
A. office97 B. MS-DOS C. Windows D. UNIX
4. Sheet1 是 Excel 中的一个默认（ ）
A. 工作簿 B. 单元格 C. 工作表 D. 页
5. 在 Excel 中，如果输入分数，则需在数值前加上（ ）
A. 0 与空格 B. 空格与 0 C. 空格 D. 0
6. 在 Excel 中，公式“SUM+（\$D2: \$C\$7）”中的\$C\$7 属于（ ）引用。
A. 绝对 B. 相对 C. 列相对行绝对的混合 D. 行相对列绝对的混合
7. 在 Excel 工作表中，系统默认状态下，日期和时间数据会自动（ ）对齐。
A. 靠左 B. 靠右 C. 居中 D. 靠上
8. 下述（ ）不是数据清单应具有的特征。
A. 一个数据清单独占一个工作表 B. 数据清单中避免对单元格进行格式化
C. 其它信息不要放在数据清单的区域内 D. 数据清单中避免出现空行或空列
9. 下列 VB5 控件中（ ）可以自动设置滚动条。
A. 文本框 B. 图片框 C. 框架 D. 复选框
10. 若想在窗体显示一句提示信息，下列（ ）控件最适宜实现
A. 框架 B. 标签 C. 单选钮 D. 列表框
11. 下述哪个是 VB5 合法变量名？（ ）
A. Then B. If C. %h121 D. Then_1
12. 若想要建立一个学生管理的输入界面，其中要很要求选择学生的性别和选择选修课情况（共有 5 门课，供任选），应如何在窗体利用单选按钮和复选框来实现
A. 将 5 门课程用一组 5 个复选框来表示，将性别用一组两个复选框来表示
B. 将 5 门课程用一组 5 个单选框来表示，将性别用一组两个复选框来表示
C. 将 5 门课程用一组 5 个复选框来表示，将性别用一组两个单选框来表示
D. 将 5 门课程用一组 5 个单选框来表示，将性别用一组两个单选框来表示
13. 假设在当前 VB5 窗体有一个文本框 Text3,并有 Text3_keyPress 过程如下（其中变量 S1 的初始值为 0）
Private Sub Text3_keyPress (keyAscii As Integer)
Static S1
If keyAscii=13 Then '13 是回车符的 ASCII 码)
S1=S1+1
Form1.Print S1
If Text3.Text="11" Then
MsgBox "It is OK!"

```
Text3.Enabled+False
End
Else
    If S1=3 Then
        MsgBox "Sorry,invalid password!"
    End If
Else
    MsgBox "Key in password again!"
    Text3.text=""
End If
End If
End If
End Sub
```

当程序运行时用户要在文本框中输入 11，回车后窗体和屏幕上各会显示：（ ）

- A. 窗体显示“1”，屏幕上弹出一个消息框，框内显示出“Sorry,invalid password!”
- B. 窗体显示“1”，屏幕上弹出一个消息框，框内显示出“Key in password again!”
- C. 窗体显示“1”，屏幕上弹出一个消息框，框内显示出“It is OK!”
- D. 窗体显示“3”，屏幕上弹出一个消息框，框内显示出“Sorry,invalid password!”

三、判断题（每题 2 分，共 30 分）

- 1、剪贴板中只能保存最后一次剪切或复制的内容。（ ）
- 2、要关闭 Excel 的文档窗口，可以用 Alt+F4 快捷键。（ ）
- 3、Excel97 启动后自动打开一个名 book1 的工作表。（ ）
- 4、在 Excel 中，一个工作表可以包含多个工作簿。（ ）
- 5、在 Excel97 中，如果在某个单元格输入的公式中使用的是绝对地址，则复制该单元公式到另一单元时，Excel 会对后一单元的公式作相应的调整。（ ）
- 6、在 Excel 中进行数据分类汇总，既可以通过执行“数据”菜单中的“分类汇总”命令执行，也可以使用 Excel 提供的的数据透视表功能实现。前者适合依据多个字段进行分类汇总，后者适合依据一个字段进行分类汇总。（ ）
- 7、在 Excel 中，当工作表的数据发生变化时，相应图表中的数据序列也会随之变化，反之亦然（ ）
- 8、使用 Excel 默认的填充序列功能，可以自动填充出“第一课。。。第十课”的序列。（ ）
- 9、在 VB5 的标准控件中复选框控件适用于在一组按钮中显示多个选项，但用户只能而且必须选择其中一项。（ ）
- 10、在 VB 中，可能有许多事件过程不会被执行。这是因为事件过程对应的事件没有发生。（ ）
- 11、在 VB 中，变体型变量是一种没有数据类型的变量。（ ）
- 12、在 VB 中，Integer 类型的类型声明符为“! ”。（ ）
- 13、在 VB 中，局部变量只在定义它的过程内有效。（ ）
- 14、在 VB5 的过程中用 Dim 声明局部变量是动态变量，其值在过程执行完后仍保留。用 Static 声明局部变量是静态变量，其值在程序执行完后会消失。（ ）
- 15、在 VB 中，属性窗体和窗体属性中一回事。（ ）

四、多选题（提示：在下述题前有*标记的题选择是有次序的，每小题 2 分，共 20 分）

- 1、“关闭计算机的正确步骤为”（ ）
 - A. 选择“关闭系统项”
 - B. 在弹出的对话框选择“关闭计算机”并单击“确定”按钮
 - C. 单击“开始”菜单
 - D. 关闭所有已打开的窗口
- 2、下述输入数据中哪些是正确的 Excel97 的-6 左 2 分之一？（ ）
 - A. -61/2
 - B. (61/2)
 - C. -6 1/2
 - D. (6 1/2)
- 3、在 Excel97 当前工作表中，假设 C5 到 F5 单元格中已有数据型数据，求出 C5 到 F5 的总值的公式可以是（ ）

A. Sum(c5,f5) B. =Sum(c5:f5) C. =c5+d5+e5+f5 D. +Sum(c5:f5)

4、下列 () 是 Excel97 的应用范围。

A. 制作普通表格 B. 关联数据以及图表应用
C. 数据模拟运算 D. 数据库的基本操作

5、如果要使 A1 成为活动单元格, 可以 ()。

A. 单击 A1 单元格 B. 按 Home 键
C. 按 Ctrl+Home 键 D. 在编辑栏的“名称”框内输入 A1 然后回车

6、下述哪些数据不是 Excel97 中的合法的日期格式的数据。()

A. 1989 年 12 月 31 号 B. 2001/1/11 C. 04-11-89 D. 1989.12.31

7、下述哪些是 VB 中合法的注释语句? ()

A. “这是一个注释” B. '这是一个注释
C. Rem 这是一个注释 D. 这是一个注释

8、*在 VB 中将窗体的标题设置成“评估展示”的步骤为 ();

A. 将 Caption 的值改为“评估展示” B. 将当前窗体存盘
C. 将当前窗体成为焦点 D. 打开属性窗口

9、*向窗体添加控件的步骤为 ();

A. 单击工具箱中所需的控件 B. 移动鼠标到窗体中适当位置
C. 按下鼠标向另一对角拖曳直到满意释放鼠标 D. 使窗体成为焦点

71、1.文件菜单栏 (Excel 自带的)

文件就叫 Commandbars("File"), 你可以在打开命令后加个菜单, 类似的, 编辑就是 Commandbars("Edit"), 视图就是 Commandbars("View"), 这些不会因语言版本不一样而不同, 全部是一样的。

2.快捷菜单栏

好像那些打印啊、打开文件啊、居中对齐啊那些呢, 叫 Commandbars("Standard"), 那些字体大小啊, 颜色啊, 就叫 ("Formatting")。

3.右键

Excel 的右键也是可以更改的, 名叫:Commandbars("Cell")

4.自定义菜单

类似于数据透视表的那些浮动的窗体, 不用名字, 用 Commandbars.add 就可以追加了, 最简单了。如果不能运行, 记得加个 Application.在 Commandbars 的前面。

72、由于公司合同较多, 每次缴印花税时, 计算每份印花税所需要的各面额印花的张数是一件比较繁琐的事。如: 这份合同需要缴纳 23 元的印花税则需要买两张 10 元、一张 2 元、一张 1 元面额的印花。幸好有 Excel 这个财会人员的得力工具, 我编写了一段宏代码来自动帮我计算各面额印花的张数, 既快捷又方便。想到还有很多同行要做这项工作, 所以写下来与大家共享。具体操作如下:

新建一个 Excel 文件存为 stampduty.xls, 进入宏编辑器 (Alt+F11) 在 ThisWorkbook 中加入以下代码:

'印花税各面额印花张数计算程序 BY-RAY DENG

'用途: 已知印花税额, 计算各面额印花张数

'使用方法: 做好工具条后, 只要选择要计算的原始数据, 点击“印花税面额计算”工具条即可。

' (请不要选择全列, 否则无效的计算太多.)

Sub stamp_duty()

Static flagcal As Integer '计算标志, 首次计算时清空计算结果页, 否则在结果页追加计算结果

Static rowbegain As Integer '计算结果页上可用行号, 计算结果追加时从这里开始

```
Dim filename As String '打开的需要计算数据的文件名
Dim moneytype(7) As Single '定义面额
Dim money As Single
Dim billno As Integer '票额张数
filename = ActiveWorkbook.Name '记录当前打开的文件名
If filename = VBAProject.ThisWorkbook.Name Then '不要在自身这个文件中操作
MsgBox "请选择其它文件中的数据!", vbInformation + vbOKOnly
Exit Sub
End If
Application.ScreenUpdating = False '将屏幕更新关闭,可加快宏的运行速度
billno = 0 '计数清 0
moneytype(0) = 100 '印花税面额为 0.5-100,以 0.5 为舍入标准
moneytype(1) = 50
moneytype(2) = 10
moneytype(3) = 5
moneytype(4) = 2
moneytype(5) = 1
moneytype(6) = 0.5
VBAProject.ThisWorkbook.Sheets(1).Activate '打开结果页
If flagcal = 0 Then '如果为第一次计算,则清空结果页
Cells.Select
Selection.ClearContents
Range("A1").Select
End If
ActiveSheet.Cells(1, 1) = "Origin DATA" '先写上表头
For i = 0 To 6
ActiveSheet.Cells(1, i + 2) = moneytype(i)
Next i
Workbooks(filename).Activate '转到要计算的原始数据数据文件
rowno = ActiveWindow.RangeSelection.Rows.Count '所选区域有几行
rowstart = ActiveWindow.RangeSelection.Row '起始行
colstart = ActiveWindow.RangeSelection.Column '起始列
j = rowbegin '结果页中写结果的起始行
For i = 1 To rowno '开始读原始数据
origindata = Cells(i + rowstart - 1, colstart) '读原始数据
'广州印花税尾数处理方法:过 0.5 进 1,不足舍去,刚好为 0.5 及其倍数则不变 If origindata * 100 Mod 50 <> 0 Then
money = Round(origindata, 0) '以 0.5 为界限作四舍五入
Else
money = origindata '整数及尾数刚好为 0.5 不作处理
End If
VBAProject.ThisWorkbook.Sheets(1).Activate '转到结果页
ActiveSheet.Cells(i + 1 + j, 1) = origindata '在第一列写入原始数据
Workbooks(filename).Activate '转到要计算的原始数据数据文件
For k = 0 To 6 '循环计算各面额所需张数
```

```
While money >= moneytype(k)
money = money - moneytype(k)
billno = billno + 1
Wend
VBAProject.ThisWorkbook.Sheets(1).Activate
ActiveSheet.Cells(i + 1 + j, k + 2) = billno '在相应位置写入该面额张数
billno = 0 '清 0 准备计算下一面额
Workbooks(filename).Activate
Next k
rowbegain = rowbegain + 1 '结果页中写结果的起始行下移一行
Next i
flagcal = flagcal + 1 '计算次数累加
rowbegain = rowbegain + 1 '添加一空行区分不同次数的结果
Application.ScreenUpdating = True '恢复屏幕刷新
VBAProject.ThisWorkbook.Sheets(1).Activate '打开结果页
End Sub
```

编好后存盘，退出宏编辑器。开始构建一个工具栏：[工具]—[自定义]，点击[工具栏]---[新建]，工具栏名称定为“印花税”，这时会出现，点击[命令]，将“命令”列表中任意图标拖动到这个工具栏中，在刚拖过来的图标上点右键，选择[命名]，将其改为“印花税面额计算”，选择[总是只用文字]，这时会出现，在“印花税面额计算”上点一次右键，选择[指定宏]，选择“ThisWorkbook.stamp_duty”，点[确定]即可。以后要计算时，只要选好原始数据，点击该工具栏即可。

73、Sub AA()

```
Sheet1.Visible = 2 '深度隐藏
```

```
Sheet1.Visible = -1 '显示
```

```
End Sub
```

74、你在“workbook”的 Private Sub Workbook_BeforeClose(Cancel As Boolean) 中加入如下句子：myNowTime = Format(Now, "yymmdd" & "-" & "hhmmss")

```
ActiveWorkbook.SaveAs Filename:=currPath & myNowTime & ".xls", AddToMru:=False
```

`（“currPath”为你的路径及文件名称）

这样你保存文件可以以年、月、日来实现。甚至可以小时、分、秒来保存。对于在编程的来讲比较有用。如果你不需要“小时、分及秒”，可把后面的 [& "-" & "hhmmss"] 取消即可。

75、我们日常上送、下发的报表材料、通知等都要加盖公章，如果把这项工作交给 Excel 或 Word 来完成，我们的工作就轻松多了。

第一步：制作公章图案

首先我们要做出一个公章的图案，最简单的办法是把公章图案扫描到电脑中，然后处理成透明的 GIF 图像。我们也可以直接用 Excel 来制作：把绘图工具打开，选中“椭圆”工具，在按下“Shift”键的同时拖开鼠标，就可以得到一个正圆了。双击这个正圆打开“设置自选图形格式”对话框，在“颜色与线条”标签中，填充颜色选“无填充颜色”，线条颜色设为红色，选 3 磅粗的单线形（图）。公章的文字用艺术字来制作，填充颜色和线条颜色都用红色，并设成无阴影产。弧形文字和水平文字要分开来做，在做弧形文字时，把艺术字拖到圆形的上方，在艺术字工具中选“艺术字形状—细上弯弧”，按住黄色的四方块往下拉，再作适当的调整，就可以做出公章里的圆弧形文字了。公章中间还有一个红五星，用“自选图形”的星形就可以做出来了，填充颜色和线条颜色用红色。最后，按住“Shift”键把组成公章的文字、图形全部选上，执行右键菜单中的“组合”命令，一个公章就做好了。

如何把做好的公章保存出来？这里有一方法：把工作表另存为 Web 页，然后到保存目录中找到*.files 的文件夹，里面有一个 GIF 图片，这就是刚才做好的公章图案了，它的背景是透明的，我们把它改名为 gongzhang.gif 保存下来即可。

第二步：添加“盖章”按钮

接下来我们给 Excel 添加一个盖章按钮，当个工作表做好后，点击这个盖章按钮，就可以为我们盖上公章了。

先把公章图形复制出来（用来粘贴作为按钮的图标），然后打开“工具—自定义”对话框，选中“命令”标签，在“类别”栏中找到“宏”，在右边的“命令”栏里就会出现一项“自定义按钮”。用鼠标把这个笑脸图标拖出到菜单栏或工具栏上放下，在笑脸图标上击右键，在弹出的菜单中把“命名”处的文字改为“加盖公章”。接下来点击“粘贴按钮图标”这个命令，就可以用刚和复制的公章图形来代替笑脸图标了。把鼠标移下来选中“分配超级链接—插入图片”，然后在“请键入文件名称或 Web 页名称”栏里输入公章图片 gongzhang.gif 的文件名及路径，然后按“确定”返回。

好了，看到“加盖公章”这个按钮了吧，点击一下看看，呵呵，页面上就盖上一个鲜红的公章了，用鼠标可以把它拖到任意的地方。在 Word 文档中加盖公章的方法与此大同小异，大家可以自己试一试。

76、如何使某一个固定的菜单项无效

```
Application.CommandBars("File").Controls(6).Enabled = False
```

Controls(6)中的 6 就是所在菜单栏的行数

我觉得这样可能更可靠：

```
Dim i As Integer
```

```
For i = 1 To Application.CommandBars("File").Controls.Count
```

```
If Application.CommandBars("File").Controls(i).Caption = "另存为(&A)..." Then
```

```
Application.CommandBars("File").Controls(i).Enabled = False
```

```
End If
```

```
Next i
```

77、请教大家，我想在 EXCEL 文件退出时，另存为当前目录下的子目录 BAK 中，比如 EXCEL 文件在 C 盘，就另存为 C:\BAK 子目录下，但 EXCEL 文件的位置不确定，不知如何写语句？

注意先引用"Microsoft Scripting Runtime"

```
Private Sub Workbook_BeforeClose(Cancel As Boolean)
```

```
' 在本文件要關閉前，執行備份動作。
```

```
If Not ActiveWorkbook.Saved Then Exit Sub ' 若原文件未存檔，備份檔也不存。
```

```
ThisPath = ThisWorkbook.Path
```

```
' 假定備份文件夾之名稱爲"BAK"，若本文件是備份檔，則不需再備份。
```

```
If Len(Application.WorksheetFunction.Substitute(ThisPath, "BAK", "")) < Len(ThisPath) Then Exit Sub
```

```
Bak = ThisPath & "\" & "BAK"
```

```
' 檢查備份文件夾是否存在，若不存在，就建立一個。
```

```
If Len(Dir(Bak, vbDirectory)) > 0 Then
```

```
    If (GetAttr(Bak) And vbDirectory) = vbDirectory Then GoTo 3 ' 若已存在，跳到下一步。
```

```
End If
```

```
MkDir Bak ' 建立備份文件夾。
```

```
' 改變目前路徑到備份文件夾路徑，並備份之。
```

3 ChDir Bak

```
Application.EnableEvents = False ' 避免執行 BeforeSave 事件。
```

```
Application.DisplayAlerts = False ' 避免顯示是否要覆蓋原備份文件之訊息。
```

```
ActiveWorkbook.SaveAs
```

```
ChDir ThisPath ' 改變目前路徑回到原文件夾路徑。
```

```
Application.DisplayAlerts = True
```

```
Application.EnableEvents = True
```

```
End Sub
```

上述程式有個漏洞，就是在本文件做過更動，且存檔過了，但最後一次的更動未存檔，則 If Not ActiveWorkbook.Saved Then Exit Sub 這個判斷會造成不存備份檔之錯誤決定。

若想彌補上述錯誤，將觸發的事件改成 `Workbook_BeforeSave`，卻會造成 Excel 關閉的錯誤。

78、文件保存为以某一单元格中的值为文件名的宏怎么写

用命令：`ActiveWorkbook.SaveCopyAs Str(Range("Sheet1!A1")) + ".xls"`

79、Offset 属性

参阅应用于示例特性应用于 `Range` 对象的 `Offset` 属性。

返回一个 `Range` 对象，该对象代表某个指定区域以外的区域。只读。

`expression.Offset(RowOffset, ColumnOffset)`

expression 必需。该表达式返回一个 `Range` 对象。

RowOffset `Variant` 类型，可选。区域偏移的行数（正值、负值或 0（零））。正值表示向下偏移，负值表示向上偏移，默认值为 0。

ColumnOffset `Variant` 类型，可选。区域偏移的列数（正值、负值或 0（零））。正值表示向右偏移，负值表示向左偏移，默认值为 0。

应用于 `TickLabels` 对象的 `Offset` 属性。

返回或设置各级别标签之间的距离以及第一级标签与坐标轴之间的距离。其默认距离为百分之百，代表坐标轴标签与坐标轴之间的默认距离。其值可以为从 0 到 1000 之间的一个整数百分比，表示相对于坐标轴标签的字体大小。`Long` 类型，可读写。

`expression.Offset`

expression 必需。该表达式返回一个 `TickLabels` 对象。

示例

应用于 `Range` 对象。

本示例激活 `Sheet1` 上活动单元格向右偏移三列、向下偏移三行处的单元格。

`Worksheets("Sheet1").Activate`

`ActiveCell.Offset(rowOffset:=3, columnOffset:=3).Activate`

本示例假定 `Sheet1` 中包含一个具有标题行的表格。本示例先选定该表格，但并不选择标题行。运行本示例之前，活动单元格必须位于表格中。

`Set tbl = ActiveCell.CurrentRegion`

`tbl.Offset(1, 0).Resize(tbl.Rows.Count - 1, _
tbl.Columns.Count).Select`

应用于 `TickLabels` 对象。

如果偏移量小于 500，则本示例将 `Chart1` 中数值轴上标签之间的距离设置为当前距离的两倍。

`With Charts("Chart1").Axes(xlValue).TickLabels`

`If .Offset < 500 then`
`.Offset = .Offset * 2`
`End If`

`End With`

80、新建工作簿

参阅特性若要在 `Visual Basic` 中创建新的工作簿，请使用 `Add` 方法。下述过程创建了新的工作簿。Microsoft Excel 自动将该工作簿命名为“BookN”，其中“N”是下一个可用的数字。新工作簿将成为活动工作簿。

`Sub AddOne()`

`Workbooks.Add`

`End Sub`

创建新工作簿更好的方法是将其分配给一个对象变量。下例中，由 `Add` 方法返回的 `Workbook` 对象分配给了对象变量 `newBook`。然后，又设置了 `newBook` 的若干属性。使用对象变量可以很容易地控制新工作簿。

`Sub AddNew()`


```
Set NewBook = Workbooks.Add
With NewBook
    .Title = "All Sales"
    .Subject = "Sales"
    .SaveAs Filename:="Allsales.xls"
End With
End Sub
```

81-100

81、Rem:将活动单元格从 A5 移到 A6，并将 A6 单元格的数值保存到 x 变量中
range("A5 ").select 此命令就是选择 A5 单元格

```
Activecell.offset(1,0).select
X=activecell.value
```

82、用代码窗口中可用 F8 运行宏，Alt+F8 显示宏对话框，Alt+F11 可以打开 VBE 编辑器

83、Option Explicit 在程序代码前使用该命令，则变量要在声明后才能使用，否则编译程序无法识别该变量，从而产生错误信息

84、Load 语句用于加载窗体，加载后将占用内存，Hide 方法使窗体隐藏，但仍在内存中，故不再使用窗体时应使用 Uunload 语句及时卸载，将内存交还系统，Show 方法用来显示一个窗体，格式：窗体名称.show 模式，模式可取 0 或 1，为 1 时，不能到其他窗体操作，只有关闭该窗体后才能对其他窗体操作

85、控件的命名规则，通常使用 3 个字母的前缀命名控件，如 Label 的前缀为 lbl,初学者最好养成良好的命名习惯

86、按下 Ctrl+Enter 键单元格不移动。

87、按下 F2 键直接在单元格内编辑，不需动鼠标

88、命令按钮不支持双击(Dbclick)事件

89、Private Sub CheckBox1_Click()

```
If CheckBox1.Value Then
MsgBox "333"
Else
MsgBox "666"
End If
End Sub
```

本段的主要用意在于明白了 If CheckBox1.Value Then 与 If CheckBox1.Value=True Then

90、Timer 是计时器,功能是按指定时间间隔产生定时事件

91、在一个语句要分行显示的地方加一个或多个空格，加一个下划线_然后回车转入下一物理行，作用在于代码一行写不下时续行，程序代码中一行较短时可加：把多行连成一行显示

92、if a>15 then

```
b=10
else
b=100
end if
```

可改为如下 if 语句

```
b=iif(a,15,10,100)
```

93、F12 用于启动另存为对话框

94、`x=shell(calc.exe,1)`可打开计算器程序

95、`call` 语句格式: `Call<子过程名>[(<实际参数表>)]`如果过程本身没有参数,则实参和括号可省略,并报参数放在括号中,另一个调用 `Sub` 过程的方法是: `<子过程名>[<实际参数表>]`比前一个少了 `Call` 和括号,子过程调用语句的实参在数目、类型、排列上与子过程定义语句的形式参数表一致

96、自定义函数一例,求 `abc` 三个数的平均

```
Private Function Passed(By Val a As integer, By Val b As integer, By Val c As integer)
```

```
ave=(a+b+c)/3
```

```
End Sub
```

在定义时必须向函数过程名赋值,而子程序名不能赋值

97、在工作表任意单元格输入 `=Cell("filename")`可获得文件的完整路径、文件名和工作表名

98、在打开 `Excel` 文件时按住 `Shift` 键,将不运行 `VBA` 过程,可防止宏病毒,单击文件 关闭命令,在点关闭时按住 `Shift` 键将在不运行 `VBA` 过程的情况下关闭工作簿,可防止关闭时自动运行的宏病毒。

99、让一个变量得到单元格 `A1` 到 `A5` 的总和(变量设为 `X`),

`X=Sum(Range("A1:A5"))` 错

`X=Application.WorksheetFunction.Sum(Range("A1:A5"))`正确

也就是说必须通过 `Application` 的 `WorksheetFunction` 属性间接调用工作表函数

100、`Rem` 和 `'`是注释符,注释语句是非执行语句,要养成对代码注释的习惯。

101-120

101、`ActiveSheet.Next.Select` 选择活动工作表下一张工作表,

`ActiveSheet.Previous.Select` 选上一张

102、单元格 `B2`: `B10` 数值不全为空用 `IF` 语句表达

```
For Each cl in ActiveSheet.Range("B2:B10")
```

```
If cl.Value <> "" then
```

```
msgbox "有非空单元格"
```

```
Exit For
```

```
endif
```

```
next
```

103、用 `Range` 引用单元格和单元格区域

`Range("A1")` 单元格 `A1`

`Range("A1:B5")` 从单元格 `A1` 到 `B5` 区域

`Range("A1:B5 ,B1:B7")` 多块的选定区域

`Range("A:A")` `A` 列

`Range("1:1")` 第一行

`Range("A:C")` `A` 列到 `C` 列的区域

`Range("1:5")` 第 1 行到第 5 行的区域

`Range("1:1,3:3")` 第 1、3 行

`Range("A:A,C:C")` `A` 列、`C` 列

104、用 `Cells` 及编号引用单元格

`Cells(6,1)``A6` 单元格

如果对工作表用 `Cells` 属性时不指定索引,表示引用工作表上的所有单元格,下例清除活动工作簿中工作表 `Sheet1` 上所有单元格的内容

Worksheets("sheet1").Cells.ClearContents

105、可用变量代入单元格索引值，故 Cells 属性非常适用于在单元格区域中循环，如：

For counter=1 To 20

Worksheets("sheet1").Cells(counter,3).value=counter

Next counter

106、引用行或列

Rows(1) 第 1 行

Rows 所有行

Columns(1) 第 1 列

Columns("C") 第 3 列

Columns 工作表上所有列

106、可用方括号将 A1 样式的引用或命名区域的名称括起来，作为 Range 属性的快捷方式，这样就不必键入 Range 和引号，如

Worksheets("sheet1").[A1:B2].clearContents

[MyRange].Value=30

107、用 Offset 处理按相对于其他单元格的某一位置的常用办法是使用 Offset 属性，本例将活动工作表上活动单元格下一行和右边三列的单元格的内容设置为下划线，如：

ActiveCell.Offset(1,3).font.Underline=XIDouble

108、把别的工作表 Sheet2 数据，读到当前工作表的方法列举

1) [A1]=Sheet2.[A1] 把 Sheet2A1 单元格的数据，读到 A1 单元格

2) [A2:A4]=Sheet2.[B1] 把 Sheet2 单元格 B1 的数据读到 A2:到 A4 单元格

3) Range("B1")=Sheet2.Range("B1") 把 Sheet2 工作表单元格 B1 数据，读到 B1 单元格

4)Range("C1:C3")=Sheet2.Range("C1") 把 Sheet2 工作表单元格 C1 数据，读到 C1:C3

5)Cells(1,4)=Sheet2Cells(1,4) 把 Sheet2 工作表单元格 D1 数据，读到 D1 单元格

6)Range(Cells(1,5),Cells(5,5)=Sheet2.Cells(1,5) 把 sheet2 工作表单元格 E1 数据，读到 E1:E5 单元格

7)Selection.Value=Sheet2.[F1] 把 Sheet2 工作表单元格[F1]数据，读到任何你点选的单元格

109、Sub 前有个 Private 表示是私有子程序，这个子程序不会出现在“宏”对话框中

110、Sub test()

ActiveSheet.Calculate

End Sub 重算活动工作表

111、编程前应该尽可能地多了解 Excel 对象的属性、方法

112、每一个 Excel 对象的属性、方法的调用都要通过 OLE 连接的一个或多个调用，这些 OLE 调用都是需要时间的，减少使用对象引用能加快 VBA 代码的运行

113、使用 With 语句

Workbooks(1).Sheets(1).Range("A1:A1000").font.Name="Pay"

Workbooks(1).Sheets(1).Range("A1:A1000").Font.Fontstyle="Bold"...

改用 With 语句则运行速度加快

例：With Workbooks(1).Sheets(1).Range("A1:A1000").font

.Name="Pay"

.Fontstyle= Bold"

...

End With

114、用 set 设置对象变量，以减少对象的访问，如：

Set MyRange=Workbooks(1).Sheets(1)

```
Mysheet.Range("A1").Value=100
```

```
Mysheet.Range("A2").Value=200
```

比直接用 `Workbooks(1).Sheets(1). Range("A1").Value=100`

`Workbooks(1).Sheets(1). Range("A2").Value=200` 运行快

115、在循环中要尽是减少对象的访问

```
For k=1 To 100
```

```
Sheets("sheet1").select
```

```
Cells(k,1).value=Cells(1,1).Value
```

```
Next k
```

更快的代码是

```
set TheValue=Cells(1,1).Value
```

```
Sheets("sheet1").select
```

```
For k=1 To 100
```

```
Cells(k,1).value=TheValue
```

116、减少对象的激活和选择

如果你是通过录制宏的来学习 VBA 的程序里一定充满了对对象的激活和选择，如 `Workbooks(XXX).active` 、`Sheets(XXX).Select` 、`Range(XXX).Select` 等，但事实上大多数情况下这些操作不是必须的，如

```
Sheets("sheet1").Select
```

```
Range("A1").Value=100
```

```
Range("A2").Value=200
```

可改为 `With sheets("Sheet3")`

```
.Range ("A1")=100
```

```
.Range ("A2")=200
```

117、关闭屏幕更新是提高运行速度的最有效的办法，推荐使用

```
Application.ScreenUpdate=False
```

程序运行后再改回来

118、VBA 中默认的数据类型是 **Variant**，你必须选择使用何种数据类型，因为 **Variant** 数据类型占用存储空间较大(16 或 22 字节)而且它将影响程序的性能，**Vba** 必须识别 **Variant** 类型的变量中存储了何种数据类型。

119、再列一个自定义函数计算价格为 10%为运费的简单函数例子

```
Public Function Shipping(Price)
```

```
Shipping=Price*0.1
```

```
End Function
```

如还是不懂的话，将上述过程复制到模块中，然后在工作表任意单元格中输入 `=Shipping(C1)` 你就会明白

120、`ActiveWindow.DisplyGridlines=False` 此句用来关闭网格线。

121、100 个错误类型

- 1 应用程序定义或对象定义错误
- 2 应用程序定义或对象定义错误
- 3 无 `GoSub` 返回
- 4 应用程序定义或对象定义错误
- 5 无效的过程调用或参数
- 6 溢出

- 7 内存溢出
- 8 应用程序定义或对象定义错误
- 9 下标越界
- 10 该数组被固定或暂时锁定
- 11 除数为零
- 12 应用程序定义或对象定义错误
- 13 类型不匹配
- 14 溢出串空间
- 15 应用程序定义或对象定义错误
- 16 表达式太复杂
- 17 不能执行所需的操作
- 18 出现用户中断
- 19 应用程序定义或对象定义错误
- 20 无错误恢复
- 21 应用程序定义或对象定义错误
- 22 应用程序定义或对象定义错误
- 23 应用程序定义或对象定义错误
- 24 应用程序定义或对象定义错误
- 25 应用程序定义或对象定义错误
- 26 应用程序定义或对象定义错误
- 27 应用程序定义或对象定义错误
- 28 溢出堆栈空间
- 29 应用程序定义或对象定义错误
- 30 应用程序定义或对象定义错误
- 31 应用程序定义或对象定义错误
- 32 应用程序定义或对象定义错误
- 33 应用程序定义或对象定义错误
- 34 应用程序定义或对象定义错误
- 35 子过程或函数未定义
- 36 应用程序定义或对象定义错误
- 37 应用程序定义或对象定义错误
- 38 应用程序定义或对象定义错误
- 39 应用程序定义或对象定义错误
- 40 应用程序定义或对象定义错误
- 41 应用程序定义或对象定义错误
- 42 应用程序定义或对象定义错误
- 43 应用程序定义或对象定义错误
- 44 应用程序定义或对象定义错误
- 45 应用程序定义或对象定义错误
- 46 应用程序定义或对象定义错误
- 47 DLL 应用程序客户太多
- 48 加载 DLL 错误
- 49 DLL 调用约定错误
- 50 应用程序定义或对象定义错误

- 51 内部错误
- 52 文件名或文件号错误
- 53 文件未找到
- 54 文件模式错误
- 55 文件已打开
- 56 应用程序定义或对象定义错误
- 57 设备 I/O 错误
- 58 文件已存在
- 59 记录长度错误
- 60 应用程序定义或对象定义错误
- 61 磁盘已满
- 62 输入超出文件尾
- 63 记录号错误
- 64 应用程序定义或对象定义错误
- 65 应用程序定义或对象定义错误
- 66 应用程序定义或对象定义错误
- 67 文件太多
- 68 设备不可用
- 69 应用程序定义或对象定义错误
- 70 拒绝的权限
- 71 磁盘未准备好
- 72 应用程序定义或对象定义错误
- 73 应用程序定义或对象定义错误
- 74 不能更名为不同的驱动器
- 75 路径/文件访问错误
- 76 路径未找到
- 77 应用程序定义或对象定义错误
- 78 应用程序定义或对象定义错误
- 79 应用程序定义或对象定义错误
- 80 应用程序定义或对象定义错误
- 81 应用程序定义或对象定义错误
- 82 应用程序定义或对象定义错误
- 83 应用程序定义或对象定义错误
- 84 应用程序定义或对象定义错误
- 85 应用程序定义或对象定义错误
- 86 应用程序定义或对象定义错误
- 87 应用程序定义或对象定义错误
- 88 应用程序定义或对象定义错误
- 89 应用程序定义或对象定义错误
- 90 应用程序定义或对象定义错误
- 91 对象变量或 With 块变量未设置
- 92 For 循环未初始化
- 93 无效的模式串
- 94 无效使用 Null

- 95 应用程序定义或对象定义错误
- 96 由于对象已经激活了事件接受器支持的最大数目的事件，不能吸收对象的事件
- 97 不能调用对象的友元函数，该对象不是所定义类的一个实例。
- 98 属性或方法调用不能包括对私有对象的引用，不论是作为参数还是作为返回值
- 99 应用程序定义或对象定义错误
- 100 应用程序定义或对象定义错误

122、Private Sub Calendar1_Click()

ActiveCell = Me.Calendar1.Value

End Sub

123、设置日历控件字号

Private Sub UserForm_Initialize()

Me.Calendar1.GridFont.Size = 14

End Sub

124、以下是从金钢金作品里提出来的部份事件代码，做得很好，值得借鉴学习：

Private Sub Workbook_Activate()

MsgBox "工作簿被切换为作用工作簿", vbInformation, "Workbook_Activate"

End Sub

Private Sub Workbook_AddinInstall()

MsgBox "激活新的加载宏时", vbInformation, "Workbook_AddinInstall"

End Sub

Private Sub Workbook_AddinUninstall()

MsgBox "取消以前选取的加载宏时", vbInformation, "Workbook_AddinUninstall"

End Sub

Private Sub Workbook_BeforeClose(Cancel As Boolean)

MsgBox "工作簿被关闭之前", vbInformation, "Workbook_BeforeClose"

End Sub

Private Sub Workbook_BeforePrint(Cancel As Boolean)

MsgBox "工作簿打印之前", vbInformation, "Workbook_BeforePrint"

End Sub

Private Sub Workbook_BeforeSave(ByVal SaveAsUI As Boolean, Cancel As Boolean)

MsgBox "工作簿进行保存之前", vbInformation, "Workbook_BeforeSave"

End Sub

Private Sub Workbook_Deactivate()

MsgBox "工作簿切换为非作用工作簿", vbInformation, "Workbook_Deactivate"

End Sub

Private Sub Workbook_NewSheet(ByVal Sh As Object)

MsgBox "新建工作表", vbInformation, "Workbook_NewSheet"

End Sub

Private Sub Workbook_Open()

MsgBox "打开工作簿", vbInformation, "Workbook_Open"

End Sub

```
Private Sub Workbook_PivotTableCloseConnection(ByVal Target As PivotTable)
MsgBox "数据透视表关闭与其数据源的连接之后", vbInformation, "Workbook_PivotTableCloseConnection"
End Sub

Private Sub Workbook_PivotTableOpenConnection(ByVal Target As PivotTable)
MsgBox "数据透视表打开与其数据源的连接之后", vbInformation, "Workbook_PivotTableOpenConnection"
End Sub

Private Sub Workbook_SheetActivate(ByVal Sh As Object)
MsgBox "工作表" & Sh.Name & "切换为作用工作表", vbInformation, "Workbook_SheetActivate"
End Sub

Private Sub Workbook_SheetBeforeDoubleClick(ByVal Sh As Object, ByVal Target As Range, Cancel As Boolean)
MsgBox "任一单元格双击鼠标之后", vbInformation, "Workbook_SheetBeforeDoubleClick"
End Sub

Private Sub Workbook_SheetBeforeRightClick(ByVal Sh As Object, ByVal Target As Range, Cancel As Boolean)
MsgBox "任一单元格单击鼠标右键之后", vbInformation, "Workbook_SheetBeforeRightClick"
End Sub

Private Sub Workbook_SheetCalculate(ByVal Sh As Object)
MsgBox "工作表内容进行重算之后", vbInformation, "Workbook_SheetCalculate"
End Sub

Private Sub Workbook_SheetChange(ByVal Sh As Object, ByVal Target As Range)
MsgBox "在工作表内进行不同的操作", vbInformation, "Workbook_SheetChange"
End Sub

Private Sub Workbook_SheetDeactivate(ByVal Sh As Object)
MsgBox "工作表" & Sh.Name & "切换为非作用工作表", vbInformation, "Workbook_SheetDeactivate"
End Sub

Private Sub Workbook_SheetFollowHyperlink(ByVal Sh As Object, ByVal Target As Hyperlink)
MsgBox "按下 Excel 超链接之后", vbInformation, "Workbook_SheetFollowHyperlink"
End Sub

Private Sub Workbook_SheetPivotTableUpdate(ByVal Sh As Object, ByVal Target As PivotTable)
MsgBox "数据透视表更新之后", vbInformation, "Workbook_SheetPivotTableUpdate"
End Sub

Private Sub Workbook_SheetSelectionChange(ByVal Sh As Object, ByVal Target As Range)
MsgBox "工作表里选取不同单元格范围时", vbInformation, "Workbook_SheetSelectionChange"
End Sub

Private Sub Workbook_WindowActivate(ByVal Wn As Window)
MsgBox "工作簿切换为作用工作簿", vbInformation, "Workbook_WindowActivate"
End Sub

Private Sub Workbook_WindowDeactivate(ByVal Wn As Window)
MsgBox "工作簿切换为非作用工作簿", vbInformation, "Workbook_WindowDeactivate"
End Sub

Private Sub Workbook_WindowResize(ByVal Wn As Window)
MsgBox "工作簿被打开或窗口最大化、最小化之后", vbInformation, "Workbook_WindowResize"
End Sub

Private Sub Worksheet_SelectionChange(ByVal Target As Range)
MsgBox "选取不同单元格范围时", vbInformation, "Worksheet_SelectionChange"
```


End Sub

125、Rem 选定单元格的个数

Sub t()

MsgBox Selection.Cells.Count

End Sub

126、Application.OnKey "{F11}", "ccc"禁用 F11

127、'选中 A 列除 A1 外的第一个非空单元格

[a65536].End(xlUp).Offset(1, 0).Select

128、Private Sub CommandButton1_Click()

On Error GoTo ad '注意后面的 ad，这是错误处理的用法

i = Range("a1").Value

Sheets(CStr(i)).Select

ret = MsgBox("是否覆盖", vbYesNo, "")

If ret = vbYes Then

Range("A5:C7").Copy Sheets(CStr(i)).Range("A1") '重点在 copy 之后

Else

Sheets.Add.Name = i & ".2"

Range("A5:C7").Copy Sheets(CStr(i & ".2")).Range("A1")

End If

End

ad:

Sheets.Add.Name = i '工作表名的变化是可见的

Range("A5:C7").Copy Sheets(CStr(i)).Range("A1")

End Sub

129、用变量取代 Sheet1 中的“1”

For i = 1 To 3

Sheets("sheet" & i).Range("A2") = "1234"

Next

130、中英对照

English 繁體中文 簡體中文

Absolute Referencing 絕對參照 绝对引用

Active Cell 現存儲存格 活动单元格

Add-in 增益集 加载宏

Address 位址 地址

Array 陣列 数组

Array Formula 陣列公式 数组公式

Audit 稽核 审核

Bold 粗體 粗体

Bug 蟲 Bug

Cell 儲存格 单元格

Circular Reference 循環參照 循环引用

Code 程式碼 代码

Column 欄 列

Command 指令 命令

Comment	註解	注释
Condition	條件	条件
Constant	常數	常数
Cursor	浮標	指针
Cursor	游標	鼠标指针
Data	數據 / 資料	数据
Data Type	資料型態	数据类型
Debug	偵錯	调试
Debug	除蟲	调试
Declare	宣告	声明
Dependent	從屬	从属
Dialog Box	對話方塊	对话框
Double	雙精度浮點數	双精度浮点数
Double-click (on mouse)	雙按	双击
Drag	拖曳	拖曳
Drop-down Box	清單方塊	?
Edit	編輯	编辑
Electronic Mail / Email	電郵 / 電子郵件	电子邮件
Event	事件	事件
Execute	執行	执行
File	檔案	文件
Filter	篩選	筛选
Flowchart	流程圖	流程图
Footer	頁尾	页脚
Form	表單	窗体
Format	格式	格式
Function	函數	函数
General	一般	标准
Hard Copy	硬本	硬拷贝
Header	頁首	页眉
Help	說明	帮助
Hyperlink	超連結	超级链接
Immediate Window	即時運算視窗	立即窗口
Indent	縮排	缩进
Insert	插入	插入
Internet	互聯網	因特网
Italic	斜體	斜体
Label	標籤	标签
Link	連結	链接
Loop	迴圈	循环
Macro	巨集	宏
Member	成員	成员
Method	方法	方法
Microsoft	微軟	微软

Module	模組	模块
Newsgroup	新聞組	新闻组
Number Format	數字格式	数字格式
Object	物件	对象
Object Browser	瀏覽物件	对象浏览器
Office Assistant	Office 小幫手	Office 助手
Options	選項	选项
Parameter	參數	参数
Parameter	引數	参数
Path	路徑 (檔案的)	路径
Precedent	前導	先例
Program	程式	程序
Program/Subroutine	程序	过程
Project	專案	工程
Property	屬性	属性
Query	查詢	查询
Range	範圍	范围
Relative Referencing	相對參照	相对引用
Right-click (on mouse)	右按	右击
Row	列	行
Scatter Chart	X Y 散佈圖	XY 散点图
Scroll Bar	捲軸	滚动条
Select	選取	选择
Single	單精度浮點數	单精度浮点数
Single-click (on mouse)	單按	单击
Smart Tag	智慧標籤	智能标记
Soft Copy	軟本	软拷贝
Sort	排序	排序
Spinner	微調按鈕	微调按钮
Statement	陳述式	表达式
Strikethrough Line	刪除線	删除线
String	字串	字符串
Subscript	下標	上标
Superscript	上標	下标
Tab	索引標籤	Tab
Template	範本	模板
Text	文字	文本
Toolbar	工作列	工具栏
Tools	工具	工具
Transpose	轉置	转置
Trendline	趨勢線	趋势线
Underline	底線	下划线
Validation	驗證	确认
Variable	變數	变量

View	檢視	视图	
Visual Basic Editor		Visual Basic 編輯器	Visual Basic 编辑器
Watch	監看式	监视	
Wildcards (* 或 ?)		萬用字元	通配符
Window	視窗	窗口	
Wizard	精靈	向导	
Workbook	活頁簿	工作簿	
Worksheet	工作表	工作表	