

VBA自定义函数大全

目录

' 1. 函数作用: 返回 Column 英文字	9
' 2. 函数作用: 查询某一值第 num 次出现的值	9
' 3. 函数作用: 返回当个人工资薪金所得为 2000 元(起征点为 850 元)时的应纳个人所得税税额	10
' 4. 函数作用: 从形如"123545ABCDE"的字符串中取出数字	11
' 5. 函数作用: 从形如"ABCD12455EDF"的字符串中取出数字 ...	11
' 6. 函数作用: 按 SplitType 取得 RangeName 串值中的起始位置	12
' 7. 函数作用: 将金额数字转成中文大写	13
' 8. 函数作用: 计算某种税金	18
' 9. 函数作用: 人民币大、小写转换	19
' 10. 函数作用: 查汉字区位码	20
' 11. 函数作用: 把公元年转为农历	21
' 12. 函数作用: 返回指定列数的列标	42
' 13. 函数作用: 用指定字符替换某字符	43
' 14. 函数作用: 从右边开始查找指定字符在字符串中的位置 ...	43
' 15. 函数作用: 从右边开始查找指定字符在字符串中的位置 ...	44
' 16. 函数作用: 计算工龄	44
' 17. 函数作用: 计算日期差, 除去星期六、星期日	45
' 18. 函数作用: 将英文字反转的自定函数.	46

' 19. 函数作用:计算个人所得税.....	46
' 20. 函数作用:一个能计算是否有重复单元的函数.....	47
' 21. 数字金额转中文大写.....	48
' 22. 函数作用:将数字转成英文.....	49
' 23. 函数作用:人民币大小写转换.....	52
' 24. 函数作用:获取区域颜色值.....	53
' 25. 函数作用:获取活动工作表名.....	53
' 26. 函数作用:获取最后一行行数.....	54
' 27. 函数作用:判断是否连接在线.....	54
' 28. 函数作用:币种转换.....	54
' 29. 函数作用:检验工作表是否有可打印内容.....	55
' 30. 函数作用:查找一字符串 (withinstr) 在另一字符串中 (findstr1) 中某一次 (startnum) 出现时的位置, 返回零表示没找到.....	57
' 31. 函数作用:增加文件路径最后的 “\” 符号.....	58
' 32. 函数作用:计算所得税.....	58
' 33. 函数作用:从工作表第一行的标题文字以数字形式返回所在 列号.....	58
' 34. 函数作用:在多个工作表中查找一个范围内符合某个指定条 件的项目对应指定范围加总求和.....	59
' 35. 函数作用:返回 Column 英文字.....	60
' 36. 函数作用:查找指定列名的列数.....	60
' 37. 函数作用:文字格式的时间(分:秒)转化为数字格式(秒) ..	61

' 38. 函数作用:将"hh:mm:ss"格式的时分秒数转换成秒数	62
' 39. 函数作用:金额中文大写转数字	62
' 40. 函数作用:把角度转为度秒分、弧度等显示	63
' 41. 函数作用:身份证号码侦测	64
' 42. 函数作用:显示公式	65
' 43. 函数作用:方便财务人员理帐查找	66
' 44. 函数作用:数值转换为字符地址	68
' 45. 函数作用:字符地址转换为数值	69
' 46. 函数作用:等待时间（以秒计算）	69
' 47. 函数作用:得到字符串实际的长度（以单字节记）	70
' 48. 函数作用:18 位身份证最后一位有效性验证	70
' 49. 函数作用:计算符合 maturity condition 的拆解金额	72
' 50. 函数作用:对多个用同一分隔符分隔的待查找元素，逐一在表 区域首列内搜索，将返回选定单元格的值相加，	72
' 51. 函数作用:根据个人所得税（工资）反算工资数	73
' 52. 函数作用:判断表是否存在	74
' 53. 函数作用:角度转弧	74
' 54. 函数作用:比较相同的字符串	75
' 55. 函数作用:对选定的数组进行排序	76
' 56. 函数作用:取得指定月份天数	77
' 57. 函数作用:排序工作表活页簿	77
' 58. 函数作用:统计数组中非重复数据个数	78
' 59. 函数作用:摘取子字符串	79

' 60. 函数作用:计算 20000 余个汉字的笔画	79
' 61. 函数作用:删除当前工作表中的全部超连接	80
' 62. 函数作用:取得相近数据	81
' 63. 函数作用:提取定串中汉字	81
' 64. 函数作用:搜索重复数据(选定范围)	81
' 65. 函数作用:字符型转数字型	82
' 66. 函数作用:小写人民币转大写人民币	83
' 67. 函数作用:取得指定月份人星期天个数	84
' 68. 函数作用:侦测档案是否包含宏	84
' 69. 函数作用:获取循环参照单元格	85
' 70. 函数作用:创建桌面快捷方式	86
' 71. 函数作用:自动建立多级目录	86
' 72. 函数作用:统计经筛选后符合条件的记录条数	87
' 73. 函数作用:复制单元格列高与栏宽	87
' 74. 函数作用:取消隐藏工作表 (包括 vba Project 工程保护的)	88
' 75. 函数作用:删除单元格自定义名称	88
' 76. 函数作用:从文件路径中取得文件名	89
' 77. 函数作用:取得一个文件的扩展名	89
' 78. 函数作用:取得一个文件的路径	90
' 79. 函数作用:十进制转二进制	90
' 80. 函数作用:检查一个数组是否为空	90
' 81. 函数作用:字母栏名转数字栏名	91
' 82. 函数作用:数字栏名转文字栏名	91

' 83. 函数作用:判断一件活页夹中是否还有子目录	92
' 84. 函数作用:判断一个文件是否在使用中	92
' 85. 函数作用:列出档案详细摘要信息	93
' 86. 函数作用:获取菜单 ID 编号及名称列表	93
' 87. 函数作用:状态列动态显示文字	94
' 88. 函数作用:取得一个文件的路径 2	94
' 89. 函数作用:取得一个文件的路径 3	95
' 90. 函数作用:取得 Activecell 的栏名	95
' 91. 函数作用:取得单元格中指定字符前的字符	95
' 92. 函数作用:前单元格指定字符前的字符颜色改成红色	95
' 93. 函数作用:根据数字返回对应的字母列号	96
' 94. 函数作用:取工作表名字	96
' 95. 函数作用:取消所有隐藏的宏表	97
' 96. 函数作用:导出 VBA Project 代码	97
' 97. 函数作用:导入 VBA Project 代码	97
' 98. 函数作用:取得汉字拼音的第一个字母	98
' 99. 函数作用:获取两栏中相同的数据	100
' 100. 函数作用:选取当前工作表中公式出错的单元格,关返回出 错个数	101
' 101. 函数作用:将工作表中最后一列作为页脚打印在每一面页尾	101
' 102. 函数作用:获取 vbproject 引用项目	102
' 103. 函数作用:移除 Excel 工作表中的外部数据连接	103
' 104. 函数作用:将选择定单元格作成镜像图片	103

' 105. 函数作用:反选择单元格中的数	105
' 106. 函数作用:在 Excel 中加入一个量度尺(以厘米为单位) .	106
' 107. 函数作用:在 Excel 中加入一个量度尺(以寸为单位) ...	109
' 108. 函数作用:取得一个短文件名的长文件名	111
' 109. 函数作用:取得临时文件名	112
' 110. 函数作用:等用 Shell 调用的程序执行完成后再执行其它程序	112
' 111. 函数作用:将 Mouse 显示成动画	113
' 112. 函数作用:限制 Mouse 移动范围	114
' 113. 函数作用:取得当前激活窗品句柄及标题	114
' 114. 函数作用:取得屏幕分辨率	115
' 115. 函数作用:自动建立多级目录	115
' 116. 函数作用:将文件长度置零	116
' 117. 函数作用:读取 WIN9X / Me 共享文件夹密码	116
' 118. 函数作用:取得预设的打印机及设置预设的打印机	119
' 119. 函数作用:获得当前操作系统的打印机个数及检测打印是否存在	120
' 120. 函数作用:枚举打印机名称清单	120
' 121. 函数作用:读取网络服务器当前时间	122
' 122. 函数作用:下载文件到指定目录	123
' 123. 函数作用:自动映射网络驱动器	124
' 124. 函数作用:自动断开网络驱动器	125
' 125. 函数作用:连接选定单元格中的内容	125

' 126. 函数作用:获取一个单元格中有指定字体颜色部份数据 .	126
' 127. 函数作用:对指定文件加 XLS 加密	126
' 128. 函数作用:选择指定范围内使用了填充颜色的单元格 ...	127
' 129. 函数作用:在特定的区域内查找文本, 返回值是包含查找文本的单元格	127
' 130. 函数作用:返回特定区域中最大值的地址	128
' 131. 函数作用:删除表格中使用范围内的所有空白单元格 ...	129
' 132. 函数作用:返回数组中有多少个指定的字符串	129
' 133. 函数作用:返回当前工作表中引用了指定的单元的地址 .	130
' 134. 函数作用:获取 Excel 中字型列表	131
' 135. 函数作用:获取一个字符串中有多少个数字字符	131
' 136. 函数作用:在 Excel 中对多列进行填充	131
' 137. 函数作用:对选定的范围进行数据填充 (忽略单元格格式)	132
' 138. 函数作用:VBA Project 加密及解密	132
' 139. 函数作用:列出收藏夹中的网址	133
' 140. 函数作用:计算两个日期之间相隔的年份, 比如年龄, 工龄等. 可计算从 1000 年 01 月 01 日起的日期	134
' 141. 函数作用:从字符串提取纯数字	135
' 142. 函数作用:将一个数组按升序排列	136
' 143. 函数作用:将一个数组按降序排列	137
' 144. 函数作用:删除空白列	137
' 145. 函数作用:判断工作表是否为空白	138
' 146. 函数作用:将数据按类分到不同工作簿	138

' 147. 函数作用:单元格内数据排序.....	139
' 148. 函数作用:对多栏排序.....	140
' 149. 函数作用:返回计算公式的值 [, 值的计算公式].....	140
' 150. 函数作用:把第一列=某个值对应的第二列的内容连在一起, 并用、隔开.....	141
' 151. 函数作用:取得系统使用模式.....	142
' 152. 函数作用:计算机注销/关机/重启.....	142
' 153. 函数作用:更改计算机名称.....	143
' 154. 函数作用:从 n 位开始取出字符串中的汉字、英文字母、数 字.....	143
' 155. 函数作用:在指定列中寻找含有指定字符串的单元格, 并将 符合条件的单元格标为红色, 并将对应的下一列单元格赋值为 1。	144
' 156. 函数作用:清除字符串中的空格.....	145
' 157. 函数作用:查找合并单元格位置.....	145
' 158. 函数作用:阴阳历转换和阴阳历生日.....	145
' 159. 函数作用:利用数组和 Substitute 来替换某字符.....	149
' 160. 函数作用:一键创建斜线表头.....	150
' 161. 函数作用:自动获取指定月的工作日.....	151

正文

```
'#####
```

'1.函数作用：返回 Column 英文字

```
'#####
```

```
Function ColLetter(ColNumber As Integer) As String
    On Error GoTo ErrorHandler
    ColLetter = Left(Cells(1, ColNumber).Address(0, 0), 1 - (ColNumber > 26))
    Exit Function
ErrorHandler:
    MsgBox "Error encountered, please re-enter "
End Function
```

```
'#####
```

'2.函数作用：查询某一值第num次出现的值

```
' 参数说明：Value1: 查询引用的数值；
'             Range1: 查询区域；
'             num: 指定查询第几次出现；
'             Col: 返回值，相对引用区域，相对引用列的右数第 Col 列
'#####
```

```
Function MyFind(Value1, ByVal Range1 As Range, ByVal num As Integer, ByVal Col As Integer)
    If Value1 = "" Then Exit Function
    If Range1.Columns.Count > 1 Then Exit Function
    For Each D In Range1
        If D.Value = Value1 Then
            C = C + 1
        End If
    Next D
    If C = num Then
        Return Range1.Cells(1, C + Col - 1).Value
    End If
    Return ""
End Function
```

```

        If c = num Then
            v1 = D(1, Col)
            Exit For
        End If
    ElseIf IsEmpty(D) Then
        Exit For
    End If
Next
If v1 = "" Then v1 = "not"
MyFind = v1
End Function

```

'#####

'3.函数作用：返回当个人工资薪金所得为 2000 元(起征点为 850 元)时的应纳个人所得税税额

' 语 法: Grsds(bsc, mysala)
 ' 参数说明: bsc: 必选项, 为起征点, 包括税法规定的工资基数 800 元加上允许税前扣除的合理费用;

' mysala: 必选项, 为人个工资薪金所得。

' 示 例: Grsds(850, 20000) =

'#####

Function Grsds(bsc As Double, mysala As Double) As Double

'bsc 为起征点加上允许税前扣除的合理费用,mysala 为工资薪金所得

On Error GoTo Grsds_err

Select Case mysala

Case Is <= bsc

Grsds = 0

Case Is <= bsc + 500

Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.05, 2)

Case Is <= bsc + 2000

Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.1 - 25,

2)

Case Is <= bsc + 5000

Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.15 - 125,

2)

Case Is <= bsc + 20000

Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.2 - 375,

2)

Case Is <= bsc + 40000

Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.25 - 1375,

2)

Case Is <= bsc + 60000

```

        Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.3 - 3375,
2)
        Case Is <= bsc + 80000
            Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.35 - 6375,
2)
        Case Is <= bsc + 100000
            Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.4 - 10375,
2)
        Case Else
            Grsds = Application.WorksheetFunction.Round((mysala - bsc) * 0.45 -
15375, 2)
        End Select
Grsds_Exit:
    Exit Function
Grsds_err:
    MsgBox Err.Number & ":" & Err.Description
    Resume Grsds_Exit
End Function

```

'#####

'4.函数作用：从形如"123545ABCDE"的字符串中取出数字

'#####

```

Function myvalue(mystring As String) As Double
    myvalue = Val(mystring)
End Function

```

'#####

'5.函数作用：从形如"ABCD12455EDF"的字符串中取出数字

'#####

```

Function mydata(mystring As String) As Double
    Dim i As Integer
    i = 1
    Do Until Val(Mid(mystring, i, 1)) > 0
        i = i + 1
    Loop
    mydata = Val(Mid(mystring, i, Len(mystring) - i + 1))
End Function

```

```
'#####
```

'6.函数作用：按SplitType取得RangeName串值中的起始位置

```
'#####
```

'1:单元格, 2:行号, 3:列号, 4:范围

```
Public Const SINGLE_CELL = 1
Public Const ROW_NUM = 2
Public Const COL_NUM = 3
Public Const RANGE_ALL = 4

Public Function SplitRangeName(RangeName As String, SplitType As Integer) As String
    If VBA.Len(RangeName) < 3 Then
        Exit Function
    Else
        RangeName = VBA.Right(RangeName, VBA.Len(RangeName) - VBA.InStr(1,
RangeName, "!") - 1)
        If VBA.InStr(1, RangeName, ":") > 0 Then RangeName = VBA.Left(RangeName,
VBA.InStr(1, RangeName, ":") - 1)
        Select Case SplitType
            Case SINGLE_CELL
                If VBA.InStr(1, RangeName, ":") <> 0 Then
                    SplitRangeName = "$" & VBA.Left(RangeName, VBA.InStr(1,
RangeName, ":") - 1)
                Else
                    SplitRangeName = "$" & RangeName
                End If
            Case ROW_NUM
                SplitRangeName = VBA.IIf(VBA.InStr(1, RangeName, "$") > 0,
VBA.Right(RangeName, VBA.Len(RangeName) - VBA.InStr(1, RangeName, "$")),
RangeName)
                If Not IsNumeric(SplitRangeName) Then
                    SplitRangeName = ""
                    MsgBox "", vbInformation, ""
                End If
            Case COL_NUM
                If VBA.InStr(1, RangeName, "$") > 0 Then
                    SplitRangeName = VBA.Left(RangeName, VBA.InStr(1, RangeName,
"$") - 1)
                Else
                    SplitRangeName = RangeName
                End If
                If IsNumeric(SplitRangeName) Then
```

```

        SplitRangeName = ""
        MsgBox "", vbInformation, ""
    End If
    Case RANGE_ALL
        SplitRangeName = "$" & RangeName
    End Select
End If
End Function

```

```

'#####

```

'7.函数作用:将金额数字转成中文大写

```

'#####

```

```

Function Money(Number As Currency)
    Dim i, j, k, m, leng As Integer '计数器
    Dim Zero As Integer '连续零标识
    Dim Tnumber As String '储存数字字符串,计算数组长度
    Dim Num() As String '定义数组
    Dim Num1(3) As String '存储万元以下数字
    Dim Num2(1) As String '储存拆分后的数字
    Dim Cha(8), Cha1(9), Cha2(4) As String '储存转化后的汉字
    Dim Zcha As String '连接后的字符串
    Dim Flag, Flag1 As Boolean '正负标志
    Flag = True
    Flag1 = False
    Zero = 0
    '如果大于一亿,则不处理
    If (Number > 99999999) Or (Number < -99999999) Then
        MsgBox ("Sorry,数据超过一亿,暂不处理。")
        MsgBox ("顺便问一下,你真有那么多钱吗?")
        Money = "Sorry!"
    Else
        If (Number = 0) Then
            Money = "零元整"
        Else
            '*****将负数数字转化正数并更改标识*****
            If (Number < 0) Then
                Number = Number * (-1)
                Flag = False
            End If
            '*****小数点后超过两位,则截断*****
            If (((Number - Int(Number)) * 100 - Int((Number - Int(Number)) * 100))
> 0) Then

```

```

        Tnumber = CStr(Int(Number * 100) / 100)
Else
    Tnumber = CStr(Number)
End If
'*****处理四舍五入*****
If (((Number - Int(Number)) * 100 - Int((Number - Int(Number)) * 100))
>= 0.5) Then
    Tnumber = CStr((CCur(Tnumber)) + 0.01)
End If
Number = CCur(Tnumber)
'*****重新分配数组空间*****
ReDim Num(Len(Tnumber) - 1) As String
'*****将字符串分开存储至数组中*****
For i = 0 To Len(Tnumber) - 1
    Num(i) = Mid(Tnumber, i + 1, 1)
Next i
'*****定义所需字符*****
Dim M1, M2
M1 = Array("零", "壹", "贰", "叁", "肆", "伍", "陆", "柒", "捌", "玖")
M2 = Array("", "拾", "佰", "仟", "万", "亿")
'*****处理小于一元金额*****
'*****小数点后一位, 则*****
If ((Number - Int(Number) > 0) And ((Number * 100 - Int(Number) * 100)
Mod 10) = 0) Then
    i = i - 1
    Num2(0) = Num(i)
    Num(i) = ""
    i = i - 1
    Num(i) = ""
    i = i - 1
    Cha2(0) = M1(CByte(Num2(0)))
    Cha2(1) = "角"
    Cha2(2) = "整"
Else
    '*****小数点后两位则*****
    If ((Number - Int(Number) > 0)) Then
        i = i - 1
        Num2(1) = Num(i)
        Num2(0) = Num(i - 1)
        Num(i) = ""
        i = i - 1
        Num(i) = ""
        i = i - 1
        Num(i) = ""
        i = i - 1
        Cha2(0) = M1(CByte(Num2(0)))

```

```

        Cha2(1) = "角"
        Cha2(2) = M1(CByte(Num2(1)))
        Cha2(3) = "分"
    End If
End If
'*****分解大于一万的整数部分*****
If (Int(Number) > 9999) Then
    If (Cha2(0) <> "") Then
        i = i + 1
    End If
    For j = 3 To 0 Step -1
        Num1(j) = Num(i - 1)
        Num(i - 1) = ""
        i = i - 1
    Next j
Else
    If (Cha2(0) <> "") Then
        i = i + 1
    End If
    For j = 0 To i - 1
        Num1(j) = Num(j)
        Num(j) = ""
    Next j
End If
'*****转换万元以上数字*****
If (Num(0) <> "") Then
    leng = i
    j = 0
    For k = 0 To leng - 1
        If (Num(k) = "0") Then
            Zero = Zero + 1
            For m = 1 To 5
                If (Cha(j - 1) = M2(m)) Then
                    Flag1 = True
                End If
            Next m
            If ((Zero = 1) And (Flag1 = False)) Then
                Cha(j) = M1(CByte(Num(k)))
            End If
            If (Zero = 1) Then
                j = j + 1
            End If
        Else
            If (Num(k) <> "") Then
                If (Zero > 0) Then
                    Cha(j - 1) = "零"

```

```

        End If
        Cha(j) = M1(CByte(Num(k)))
    End If
    j = j + 1
End If
If (Num(k) = "0") Then
    i = i - 1
Else
    Cha(j) = M2(i - 1)
    j = j + 1
    i = i - 1
    Zero = 0
End If
Next k
Cha(j - 1) = "万"
Zero = 0
End If
'*****转换万元以下数字*****
If (Num1(0) <> "") Then
    j = 0
    Flag1 = False
    leng = 3
    While (Num1(leng) = "")
        leng = leng - 1
    Wend
    i = leng + 1
    For k = 0 To leng
        If (Num1(k) <> "") Then
            If (Num1(k) = "0") Then
                Zero = Zero + 1
                For m = 1 To 5
                    If (j <> 0) Then
                        If (Cha1(j - 1) = M2(m)) Then
                            Flag1 = True
                        End If
                    End If
                Next m
            If ((Zero = 1) And (Flag1 = False)) Then
                Cha1(j) = M1(CByte(Num1(k)))
            End If
            If (Zero = 1) Then
                j = j + 1
            End If
        Else
            If (Num1(k) <> "") Then
                If (Zero > 0) Then

```

```

        Cha1(j - 1) = "零"
    End If
    Cha1(j) = M1(CByte(Num1(k)))
End If
j = j + 1
End If
If (Num1(k) = "0") Then
    i = i - 1
Else
    Cha1(j) = M2(i - 1)
    j = j + 1
    i = i - 1
    Zero = 0
End If
End If
Next k
Cha1(j - 1) = "元"
If (Cha2(0) = "") Then
    Cha1(j) = "整"
End If
End If
'*****连接字符串*****
j = 0
While (Cha(j) <> "")
    Zcha = Zcha & Cha(j)
    j = j + 1
Wend
j = 0
While (Cha1(j) <> "")
    Zcha = Zcha & Cha1(j)
    j = j + 1
Wend
j = 0
While (Cha2(j) <> "")
    Zcha = Zcha & Cha2(j)
    j = j + 1
Wend
'*****最终显示*****
If (Flag) Then
    Money = Zcha
Else
    Money = "负" & Zcha
End If
End If
End If
End Function

```

```
'#####
```

'8.函数作用:计算某种税金

```
'#####
```

```
Public Function 税(fa)
    Dim x
    If (fa - 800) > 0 And (fa - 800) < 500 Then
        x = (fa - 800) * 0.05
        税 = x
    ElseIf (fa - 800) >= 500 And (fa - 800) < 2000 Then
        x = (fa - 800) * 0.1 - 25
        税 = x
    ElseIf (fa - 800) >= 2000 And (fa - 800) < 5000 Then
        x = (fa - 800) * 0.15 - 125
        税 = x
    ElseIf (fa - 800) >= 5000 And (fa - 800) < 20000 Then
        x = (fa - 800) * 0.2 - 375
        税 = x
    ElseIf (fa - 800) >= 20000 And (fa - 800) < 40000 Then
        x = (fa - 800) * 0.25 - 1375
        税 = x
    ElseIf (fa - 800) >= 40000 And (fa - 800) < 60000 Then
        x = (fa - 800) * 0.3 - 3375
        税 = x
    ElseIf (fa - 800) >= 60000 And (fa - 800) < 80000 Then
        x = (fa - 800) * 0.35 - 6375
        税 = x
    ElseIf (fa - 800) >= 80000 And (fa - 800) < 100000 Then
        x = (fa - 800) * 0.4 - 10375
        税 = x
    ElseIf (fa - 800) >= 100000 Then
        x = (fa - 800) * 0.45 - 15375
        税 = x
    Else
    End If
End Function
```

```
'#####
```

'9.函数作用:人民币大、小写转换

'#####

Function 小写(k)

```
Application.ScreenUpdating = False
m1 = Application.WorksheetFunction.Round(k * 100, 0)
n1 = Int(m1 / 100)
n2 = Int(m1 / 10) - n1 * 10
n3 = m1 - n1 * 100 - n2 * 10
e = Application.WorksheetFunction.Text(n1, "[DBNum1]")
f = Application.WorksheetFunction.Text(n2, "[DBNum1]")
g = Application.WorksheetFunction.Text(n3, "[DBNum1]")
If n3 = 0 Then
    小写 = "人民币大写: " & e & "元" & "整"
End If
If (n3 <> 0) And (n2 <> 0) Then
    小写 = "人民币大写: " & e & "元" & f & "角" & g & "分"
    If n1 = 0 Then
        小写 = "人民币大写: " & f & "角" & g & "分"
    End If
End If
If (n3 = 0) And n2 <> 0 Then
    小写 = "人民币大写: " & e & "元" & f & "角" & "整"
    If n1 = 0 Then
        小写 = "人民币大写: " & f & "角" & "整"
    End If
End If
If (n3 <> 0) And (n2 = 0) Then
    小写 = "人民币大写: " & e & "元" & g & "分"
    If n1 = 0 Then
        小写 = "人民币大写: " & g & "分"
    End If
End If
If k = 0 Or k = "" Then
    k = ""
End If
Application.ScreenUpdating = True
End Function
```

Function 大写(k)

```
Application.ScreenUpdating = False
m1 = Application.WorksheetFunction.Round(k * 100, 0)
n1 = Int(m1 / 100)
n2 = Int(m1 / 10) - n1 * 10
```

```

n3 = m1 - n1 * 100 - n2 * 10
e = Application.WorksheetFunction.Text(n1, "[dbnum2]")
f = Application.WorksheetFunction.Text(n2, "[dbnum2]")
g = Application.WorksheetFunction.Text(n3, "[dbnum2]")
If n3 = 0 Then
    大写 = "人民币大写: " & e & "元" & "整"
End If
If (n3 <> 0) And (n2 <> 0) Then
    大写 = "人民币大写: " & e & "元" & f & "角" & g & "分"
    If n1 = 0 Then
        大写 = "人民币大写: " & f & "角" & g & "分"
    End If
End If
If (n3 = 0) And n2 <> 0 Then
    大写 = "人民币大写: " & e & "元" & f & "角" & "整"
    If n1 = 0 Then
        大写 = "人民币大写: " & f & "角" & "整"
    End If
End If
If (n3 <> 0) And (n2 = 0) Then
    大写 = "人民币大写: " & e & "元" & f & g & "分"
    If n1 = 0 Then
        大写 = "人民币大写: " & g & "分"
    End If
End If
If k = 0 Or k = "" Then
    大写 = ""
End If
Application.ScreenUpdating = True
End Function

```

'#####

'10.函数作用:查汉字区位码

'#####

```

Public Function 区(fa$) As String
    On Error Resume Next
    Dim L1$, R1$, L$, R$, a, b$, c, d, e$
    c = Len(fa)
    For i = 1 To c
        d = Mid(fa, i, 1)
        a = Hex(Asc(d))
        L1 = CInt("&H" + Mid(a, 1, 2)) - 160
    Next i
End Function

```

```

R1 = CInt("&H" + Mid(a, 3, 2)) - 160
If Len(L1) = 1 Then
    L = "0" & L1
Else
    L = L1
End If
If Len(R1) = 1 Then
    R = "0" & R1
Else
    R = R1
End If
b = d & " " & L & R & " "
e = e & b
Next i
☒ = e
End Function

```

```

'#####

```

'11.函数作用:把公元年转为农历

```

' 函数说明:本函数利用阵列处理,以方便日后组合排列
'      IntToSimDay__$($, 0)'天干地支年
'      IntToSimDay__$($, 1)'十二生肖年
'      IntToSimDay__$($, 2)'农历月
'      IntToSimDay__$($, 3)'农历日
'      IntToSimDay__$($, 4)'24 节气
'      目前可使用至 2010 年
'      Function ChineCalender(iDate, Optional num As Integer = 0)
'      num :0~8 都可用,不输入 num 预设值为 0
'      分别介绍 0~8 的用法
'      假设 A1:2002/12/22
'      ChineCalender(A1): 壬午年[马]十一月十九冬至
'      ChineCalender(A1,0): 壬午年[马]十一月十九冬至
'      ChineCalender(A1,1): [马]十一月十九冬至
'      ChineCalender(A1,2): 十一月十九冬至
'      ChineCalender(A1,3): 十一月十九
'      ChineCalender(A1,4): 壬午年
'      ChineCalender(A1,5): [马]年
'      ChineCalender(A1,6): 十一月
'      ChineCalender(A1,7): 十九日
'      ChineCalender(A1,8): 冬至
'#####

```

```

Dim IntToSimDay__$(31, 4)

```

```

Public rgstrMonthName(11) As String
Public rgstrDayName(6) As String
Public rgiDaysInMonth(11) As String
Dim B__1__$ (11)
Dim B__2__$ (220)
Dim B__3__$ (410)
Dim B__4__$ (30)
Dim B__5__$ (12)
Dim B__6__$ (12)
Dim B__7__$ (23)
Dim iYear
Dim iMonth
Dim iDay
'IntToSimDay__$ (, 0)'天干地支年
'IntToSimDay__$ (, 1)'十二生肖年
'IntToSimDay__$ (, 2)'农历月
'IntToSimDay__$ (, 3)'农历日
'IntToSimDay__$ (, 4)'24 节气

Function ChineCalender(iDate, Optional num As Integer = 0)
    Dim iYear As Integer, iMonth As Integer, iDay As Integer
    If IsDate(iDate) Then
        iYear = Year(iDate)
        iMonth = Month(iDate)
        iDay = Day(iDate)
        Call IniLunarStr
        GetLunarDays iYear, iMonth
        '          Intyear = "民国" & Application.WorksheetFunction.Text(iYear -
1911, "[DBNum1];@") & "年"
        Select Case num
            Case 0
                ChineCalender = IntToSimDay__$ (iDay - 1, 0) & IntToSimDay__$ (iDay
- 1, 1) & IntToSimDay__$ (iDay - 1, 2) _
                    & IntToSimDay__$ (iDay - 1, 3) & IntToSimDay__$ (iDay,
4)
            Case 1
                ChineCalender = IntToSimDay__$ (iDay - 1, 1) & IntToSimDay__$ (iDay
- 1, 2) _
                    & IntToSimDay__$ (iDay - 1, 3) & IntToSimDay__$ (iDay,
4)
            Case 2
                ChineCalender = IntToSimDay__$ (iDay - 1, 2) & IntToSimDay__$ (iDay
- 1, 3) & IntToSimDay__$ (iDay, 4)
            Case 3
                ChineCalender = IntToSimDay__$ (iDay - 1, 2) & IntToSimDay__$ (iDay
- 1, 3)

```

```

    Case 4
        ChineCalender = IntToSimDay__(iDay - 1, 0)
    Case 5
        ChineCalender = IntToSimDay__(iDay - 1, 1) & "年"
    Case 6
        ChineCalender = IntToSimDay__(iDay - 1, 2)
    Case 7
        ChineCalender = IntToSimDay__(iDay - 1, 3) & "日"
    Case 8
        ChineCalender = IntToSimDay__(iDay, 4)

    Case Else
        ChineCalender = ""
    End Select
Else
    ChineCalender = ""
End If
End Function

Private Sub IniLunarStr()
    Dim i
    rgstrMonthName(0) = "一月"
    rgstrMonthName(1) = "二月"
    rgstrMonthName(2) = "三月"
    rgstrMonthName(3) = "四月"
    rgstrMonthName(4) = "五月"
    rgstrMonthName(5) = "六月"
    rgstrMonthName(6) = "七月"
    rgstrMonthName(7) = "八月"
    rgstrMonthName(8) = "九月"
    rgstrMonthName(9) = "十月"
    rgstrMonthName(10) = "十一月"
    rgstrMonthName(11) = "十二月"
    B__2__(0) = 30 '11
    B__2__(1) = 29 '12 1994 (农历月份最后一天)
    B__2__(2) = 30 '1
    B__2__(3) = 30 '2
    B__2__(4) = 30 '3
    B__2__(5) = 29 '4
    B__2__(6) = 30 '5
    B__2__(7) = 29 '6
    B__2__(8) = 30 '7
    B__2__(9) = 29 '8
    B__2__(10) = 29 '9
    B__2__(11) = 30 '10
    B__2__(12) = 29 '11

```

B__2__(13) = 30 '12 1995
 B__2__(14) = 29 '1
 B__2__(15) = 30 '2
 B__2__(16) = 30 '3
 B__2__(17) = 29 '4
 B__2__(18) = 30 '5
 B__2__(19) = 29 '6
 B__2__(20) = 30 '7
 B__2__(21) = 30 '8
 B__2__(22) = 39 'r8
 B__2__(23) = 29 '9
 B__2__(24) = 30 '10
 B__2__(25) = 29 '11
 B__2__(26) = 30 '12 1996
 B__2__(27) = 29 '1
 B__2__(28) = 30 '2
 B__2__(29) = 29 '3
 B__2__(30) = 30 '4
 B__2__(31) = 30 '5
 B__2__(32) = 29 '6
 B__2__(33) = 30 '7
 B__2__(34) = 29 '8
 B__2__(35) = 30 '9
 B__2__(36) = 30 '10
 B__2__(37) = 29 '11
 B__2__(38) = 29 '12 1997
 B__2__(39) = 30 '1
 B__2__(40) = 29 '2
 B__2__(41) = 30 '3
 B__2__(42) = 29 '4
 B__2__(43) = 30 '5
 B__2__(44) = 29 '6
 B__2__(45) = 30 '7
 B__2__(46) = 30 '8
 B__2__(47) = 29 '9
 B__2__(48) = 30 '10
 B__2__(49) = 30 '11
 B__2__(50) = 29 '12 1998
 B__2__(51) = 30 '1
 B__2__(52) = 29 '2
 B__2__(53) = 29 '3
 B__2__(54) = 30 '4
 B__2__(55) = 29 '5
 B__2__(56) = 39 'r5
 B__2__(57) = 30 '6
 B__2__(58) = 30 '7

B__2__(59) = 29 '8
B__2__(60) = 30 '9
B__2__(61) = 30 '10
B__2__(62) = 29 '11
B__2__(63) = 30 '12 1999
B__2__(64) = 30 '1
B__2__(65) = 29 '2
B__2__(66) = 29 '3
B__2__(67) = 30 '4
B__2__(68) = 29 '5
B__2__(69) = 29 '6
B__2__(70) = 30 '7
B__2__(71) = 29 '8
B__2__(72) = 30 '9
B__2__(73) = 30 '10
B__2__(74) = 30 '11
B__2__(75) = 29 '12 2000
B__2__(76) = 30 '1
B__2__(77) = 30 '2
B__2__(78) = 29 '3
B__2__(79) = 29 '4
B__2__(80) = 30 '5
B__2__(81) = 29 '6
B__2__(82) = 29 '7
B__2__(83) = 30 '8
B__2__(84) = 29 '9
B__2__(85) = 30 '10
B__2__(86) = 30 '11
B__2__(87) = 29 '12 2001
B__2__(88) = 30 '1
B__2__(89) = 30 '2
B__2__(90) = 29 '3
B__2__(91) = 30 '4
B__2__(92) = 39 'r4
B__2__(93) = 30 '5
B__2__(94) = 29 '6
B__2__(95) = 29 '7
B__2__(96) = 30 '8
B__2__(97) = 29 '9
B__2__(98) = 30 '10
B__2__(99) = 29 '11
B__2__(100) = 30 '12 2002
B__2__(101) = 30 '1
B__2__(102) = 30 '2
B__2__(103) = 29 '3
B__2__(104) = 30 '4

B__2__(105) = 29 '5
B__2__(106) = 30 '6
B__2__(107) = 29 '7
B__2__(108) = 29 '8
B__2__(109) = 30 '9
B__2__(110) = 29 '10
B__2__(111) = 30 '11
B__2__(112) = 29 '12 2003
B__2__(113) = 30 '1
B__2__(114) = 30 '2
B__2__(115) = 29 '3
B__2__(116) = 30 '4
B__2__(117) = 30 '5
B__2__(118) = 29 '6
B__2__(119) = 30 '7
B__2__(120) = 29 '8
B__2__(121) = 29 '9
B__2__(122) = 30 '10
B__2__(123) = 29 '11
B__2__(124) = 30 '12 2004
B__2__(125) = 29 '1
B__2__(126) = 30 '2
B__2__(127) = 39 'r2
B__2__(128) = 30 '3
B__2__(129) = 30 '4
B__2__(130) = 29 '5
B__2__(131) = 30 '6
B__2__(132) = 29 '7
B__2__(133) = 30 '8
B__2__(134) = 29 '9
B__2__(135) = 30 '10
B__2__(136) = 29 '11
B__2__(137) = 30 '12 2005
B__2__(138) = 29 '1 2005
B__2__(139) = 30 '2 2005
B__2__(140) = 29 '3 2005
B__2__(141) = 30 '4 2005
B__2__(142) = 29 '5 2005
B__2__(143) = 30 '6 2005
B__2__(144) = 30 '7 2005
B__2__(145) = 29 '8 2005
B__2__(146) = 30 '9 2005
B__2__(147) = 29 '10 2005
B__2__(148) = 30 '11 2005
B__2__(149) = 29 '12 2006
B__2__(150) = 30 '1 2006

B__2__(151) = 29 '2 2006
 B__2__(152) = 30 '3 2006
 B__2__(153) = 29 '4 2006
 B__2__(154) = 30 '5 2006
 B__2__(155) = 29 '6 2006
 B__2__(156) = 30 '7 2006
 B__2__(157) = 39 '7 2006 r2
 B__2__(158) = 30 '8 2006
 B__2__(159) = 30 '9 2006
 B__2__(160) = 29 '10 2006
 B__2__(161) = 30 '11 2006
 B__2__(162) = 30 '12 2006
 B__2__(163) = 29 '1 2007
 B__2__(164) = 29 '2 2007
 B__2__(165) = 30 '3 2007
 B__2__(166) = 29 '4 2007
 B__2__(167) = 29 '5 2007
 B__2__(168) = 30 '6 2007
 B__2__(169) = 29 '7 2007
 B__2__(170) = 30 '8 2007
 B__2__(171) = 30 '9 2007
 B__2__(172) = 30 '10 2007
 B__2__(173) = 29 '11 2007
 B__2__(174) = 30 '12 2007
 B__2__(175) = 30 '1 2008
 B__2__(176) = 29 '2 2008
 B__2__(177) = 29 '3 2008
 B__2__(178) = 30 '4 2008
 B__2__(179) = 29 '5 2008
 B__2__(180) = 29 '6 2008
 B__2__(181) = 30 '7 2008
 B__2__(182) = 29 '8 2008
 B__2__(183) = 30 '9 2008
 B__2__(184) = 30 '10 2008
 B__2__(185) = 29 '11 2008
 B__2__(186) = 30 '12 2008
 B__2__(187) = 30 '1 2009
 B__2__(188) = 30 '2 2009
 B__2__(189) = 29 '3 2009
 B__2__(190) = 29 '4 2009
 B__2__(191) = 30 '5 2009
 B__2__(192) = 39 'r5 2009
 B__2__(193) = 29 '6 2009
 B__2__(194) = 30 '7 2009
 B__2__(195) = 29 '8 2009
 B__2__(196) = 30 '9 2009

$B_{_2_}(197) = 29 \text{ '10 2009}$
 $B_{_2_}(198) = 30 \text{ '11 2009}$
 $B_{_2_}(199) = 30 \text{ '12 2009}$
 $B_{_2_}(200) = 30 \text{ '1 2010}$
 $B_{_2_}(201) = 29 \text{ '2 2010}$
 $B_{_2_}(202) = 30 \text{ '3 2010}$
 $B_{_2_}(203) = 29 \text{ '4 2010}$
 $B_{_2_}(204) = 30 \text{ '5 2010}$
 $B_{_2_}(205) = 29 \text{ '6 2010}$
 $B_{_2_}(206) = 29 \text{ '7 2010}$
 $B_{_2_}(207) = 30 \text{ '8 2010}$
 $B_{_2_}(208) = 29 \text{ '9 2010}$
 $B_{_2_}(209) = 29 \text{ '10 2010}$
 $B_{_2_}(210) = 30 \text{ '11 2010}$
 $B_{_2_}(211) = 30 \text{ '12 2010}$

$B_{_3_}(0) = 5 \text{ '1994}$
 $B_{_3_}(1) = 20$
 $B_{_3_}(2) = 4$
 $B_{_3_}(3) = 19$
 $B_{_3_}(4) = 6$
 $B_{_3_}(5) = 21$
 $B_{_3_}(6) = 5$
 $B_{_3_}(7) = 20$
 $B_{_3_}(8) = 6$
 $B_{_3_}(9) = 21$
 $B_{_3_}(10) = 6$
 $B_{_3_}(11) = 21$
 $B_{_3_}(12) = 7$
 $B_{_3_}(13) = 23$
 $B_{_3_}(14) = 8$
 $B_{_3_}(15) = 23$
 $B_{_3_}(16) = 8$
 $B_{_3_}(17) = 23$
 $B_{_3_}(18) = 8$
 $B_{_3_}(19) = 23$
 $B_{_3_}(20) = 7$
 $B_{_3_}(21) = 22$
 $B_{_3_}(22) = 7$
 $B_{_3_}(23) = 22$
 $B_{_3_}(24) = 6 \text{ '1995}$
 $B_{_3_}(25) = 20$
 $B_{_3_}(26) = 4$
 $B_{_3_}(27) = 19$
 $B_{_3_}(28) = 6$

B__3__(29) = 21
B__3__(30) = 5
B__3__(31) = 20
B__3__(32) = 6
B__3__(33) = 21
B__3__(34) = 6
B__3__(35) = 22
B__3__(36) = 7
B__3__(37) = 23
B__3__(38) = 8
B__3__(39) = 23
B__3__(40) = 8
B__3__(41) = 23
B__3__(42) = 9
B__3__(43) = 24
B__3__(44) = 8
B__3__(45) = 23
B__3__(46) = 7
B__3__(47) = 22
B__3__(48) = 6 '1996
B__3__(49) = 20
B__3__(50) = 4
B__3__(51) = 19
B__3__(52) = 6
B__3__(53) = 21
B__3__(54) = 5
B__3__(55) = 20
B__3__(56) = 6
B__3__(57) = 21
B__3__(58) = 6
B__3__(59) = 22
B__3__(60) = 7
B__3__(61) = 22
B__3__(62) = 7
B__3__(63) = 23
B__3__(64) = 7
B__3__(65) = 23
B__3__(66) = 8
B__3__(67) = 23
B__3__(68) = 7
B__3__(69) = 22
B__3__(70) = 7
B__3__(71) = 21
B__3__(72) = 5 '1997
B__3__(73) = 20
B__3__(74) = 4

$B_{33}(75) = 18$
 $B_{33}(76) = 5$
 $B_{33}(77) = 20$
 $B_{33}(78) = 5$
 $B_{33}(79) = 20$
 $B_{33}(80) = 5$
 $B_{33}(81) = 21$
 $B_{33}(82) = 5$
 $B_{33}(83) = 21$
 $B_{33}(84) = 7$
 $B_{33}(85) = 23$
 $B_{33}(86) = 7$
 $B_{33}(87) = 23$
 $B_{33}(88) = 7$
 $B_{33}(89) = 23$
 $B_{33}(90) = 8$
 $B_{33}(91) = 23$
 $B_{33}(92) = 7$
 $B_{33}(93) = 22$
 $B_{33}(94) = 7$
 $B_{33}(95) = 22$
 $B_{33}(96) = 5$ '1998
 $B_{33}(97) = 20$
 $B_{33}(98) = 4$
 $B_{33}(99) = 19$
 $B_{33}(100) = 6$
 $B_{33}(101) = 21$
 $B_{33}(102) = 5$
 $B_{33}(103) = 20$
 $B_{33}(104) = 6$
 $B_{33}(105) = 21$
 $B_{33}(106) = 6$
 $B_{33}(107) = 21$
 $B_{33}(108) = 7$
 $B_{33}(109) = 23$
 $B_{33}(110) = 8$
 $B_{33}(111) = 23$
 $B_{33}(112) = 8$
 $B_{33}(113) = 23$
 $B_{33}(114) = 8$
 $B_{33}(115) = 23$
 $B_{33}(116) = 7$
 $B_{33}(117) = 22$
 $B_{33}(118) = 7$
 $B_{33}(119) = 22$
 $B_{33}(120) = 6$ '1999

B__3__(121) = 20
B__3__(122) = 4
B__3__(123) = 19
B__3__(124) = 6
B__3__(125) = 21
B__3__(126) = 5
B__3__(127) = 20
B__3__(128) = 6
B__3__(129) = 21
B__3__(130) = 6
B__3__(131) = 22
B__3__(132) = 7
B__3__(133) = 23
B__3__(134) = 8
B__3__(135) = 23
B__3__(136) = 8
B__3__(137) = 23
B__3__(138) = 9
B__3__(139) = 24
B__3__(140) = 8
B__3__(141) = 23
B__3__(142) = 7
B__3__(143) = 22
B__3__(144) = 6 '2000
B__3__(145) = 21
B__3__(146) = 4
B__3__(147) = 19
B__3__(148) = 5
B__3__(149) = 20
B__3__(150) = 4
B__3__(151) = 20
B__3__(152) = 5
B__3__(153) = 21
B__3__(154) = 5
B__3__(155) = 21
B__3__(156) = 7
B__3__(157) = 22
B__3__(158) = 7
B__3__(159) = 23
B__3__(160) = 7
B__3__(161) = 23
B__3__(162) = 8
B__3__(163) = 23
B__3__(164) = 7
B__3__(165) = 22
B__3__(166) = 7

B__3__(167) = 21
B__3__(168) = 6 '2001
B__3__(169) = 20
B__3__(170) = 4
B__3__(171) = 18
B__3__(172) = 5
B__3__(173) = 20
B__3__(174) = 5
B__3__(175) = 20
B__3__(176) = 5
B__3__(177) = 21
B__3__(178) = 5
B__3__(179) = 21
B__3__(180) = 7
B__3__(181) = 23
B__3__(182) = 7
B__3__(183) = 23
B__3__(184) = 7
B__3__(185) = 23
B__3__(186) = 8
B__3__(187) = 23
B__3__(188) = 7
B__3__(189) = 22
B__3__(190) = 7
B__3__(191) = 22
B__3__(192) = 5 '2002
B__3__(193) = 20
B__3__(194) = 4
B__3__(195) = 19
B__3__(196) = 6
B__3__(197) = 21
B__3__(198) = 5
B__3__(199) = 20
B__3__(200) = 6
B__3__(201) = 21
B__3__(202) = 6
B__3__(203) = 21
B__3__(204) = 7
B__3__(205) = 23
B__3__(206) = 8
B__3__(207) = 23
B__3__(208) = 8
B__3__(209) = 23
B__3__(210) = 8
B__3__(211) = 23
B__3__(212) = 7

$B_{_3_}(213) = 22$
 $B_{_3_}(214) = 7$
 $B_{_3_}(215) = 22$
 $B_{_3_}(216) = 6$ '2003
 $B_{_3_}(217) = 20$
 $B_{_3_}(218) = 4$
 $B_{_3_}(219) = 19$
 $B_{_3_}(220) = 6$
 $B_{_3_}(221) = 21$
 $B_{_3_}(222) = 5$
 $B_{_3_}(223) = 20$
 $B_{_3_}(224) = 6$
 $B_{_3_}(225) = 21$
 $B_{_3_}(226) = 6$
 $B_{_3_}(227) = 22$
 $B_{_3_}(228) = 7$
 $B_{_3_}(229) = 23$
 $B_{_3_}(230) = 8$
 $B_{_3_}(231) = 23$
 $B_{_3_}(232) = 8$
 $B_{_3_}(233) = 23$
 $B_{_3_}(234) = 9$
 $B_{_3_}(235) = 24$
 $B_{_3_}(236) = 8$
 $B_{_3_}(237) = 23$
 $B_{_3_}(238) = 7$
 $B_{_3_}(239) = 22$
 $B_{_3_}(240) = 6$ '2004
 $B_{_3_}(241) = 21$
 $B_{_3_}(242) = 4$
 $B_{_3_}(243) = 19$
 $B_{_3_}(244) = 5$
 $B_{_3_}(245) = 20$
 $B_{_3_}(246) = 4$
 $B_{_3_}(247) = 20$
 $B_{_3_}(248) = 6$
 $B_{_3_}(249) = 21$
 $B_{_3_}(250) = 5$
 $B_{_3_}(251) = 21$
 $B_{_3_}(252) = 7$
 $B_{_3_}(253) = 22$
 $B_{_3_}(254) = 7$
 $B_{_3_}(255) = 23$
 $B_{_3_}(256) = 7$
 $B_{_3_}(257) = 23$
 $B_{_3_}(258) = 8$

$B_{_3_}(259) = 23$
 $B_{_3_}(260) = 7$
 $B_{_3_}(261) = 22$
 $B_{_3_}(262) = 7$
 $B_{_3_}(263) = 21$
 $B_{_3_}(264) = 5 \text{ '2005 } 1$
 $B_{_3_}(265) = 20 \text{ '2005 } 2$
 $B_{_3_}(266) = 4 \text{ '2005 } 3$
 $B_{_3_}(267) = 18 \text{ '2005 } 4$
 $B_{_3_}(268) = 5 \text{ '2005 } 5$
 $B_{_3_}(269) = 20 \text{ '2005 } 6$
 $B_{_3_}(270) = 5 \text{ '2005 } 7$
 $B_{_3_}(271) = 20 \text{ '2005 } 8$
 $B_{_3_}(272) = 5 \text{ '2005 } 9$
 $B_{_3_}(273) = 21 \text{ '2005 } 10$
 $B_{_3_}(274) = 5 \text{ '2005 } 11$
 $B_{_3_}(275) = 21 \text{ '2005 } 12$
 $B_{_3_}(276) = 7 \text{ '2005 } 13$
 $B_{_3_}(277) = 23 \text{ '2005 } 14$
 $B_{_3_}(278) = 7 \text{ '2005 } 15$
 $B_{_3_}(279) = 23 \text{ '2005 } 16$
 $B_{_3_}(280) = 7 \text{ '2005 } 17$
 $B_{_3_}(281) = 23 \text{ '2005 } 18$
 $B_{_3_}(282) = 8 \text{ '2005 } 19$
 $B_{_3_}(283) = 23 \text{ '2005 } 20$
 $B_{_3_}(284) = 7 \text{ '2005 } 21$
 $B_{_3_}(285) = 22 \text{ '2005 } 22$
 $B_{_3_}(286) = 7 \text{ '2005 } 23$
 $B_{_3_}(287) = 22 \text{ '2005 } 24$
 $B_{_3_}(288) = 5 \text{ '2006 } 1$
 $B_{_3_}(289) = 20 \text{ '2006 } 2$
 $B_{_3_}(290) = 4 \text{ '2006 } 3$
 $B_{_3_}(291) = 19 \text{ '2006 } 4$
 $B_{_3_}(292) = 6 \text{ '2006 } 5$
 $B_{_3_}(293) = 21 \text{ '2006 } 6$
 $B_{_3_}(294) = 5 \text{ '2006 } 7$
 $B_{_3_}(295) = 20 \text{ '2006 } 8$
 $B_{_3_}(296) = 5 \text{ '2006 } 9$
 $B_{_3_}(297) = 21 \text{ '2006 } 10$
 $B_{_3_}(298) = 6 \text{ '2006 } 11$
 $B_{_3_}(299) = 21 \text{ '2006 } 12$
 $B_{_3_}(300) = 7 \text{ '2006 } 13$
 $B_{_3_}(301) = 23 \text{ '2006 } 14$
 $B_{_3_}(302) = 7 \text{ '2006 } 15$
 $B_{_3_}(303) = 23 \text{ '2006 } 16$
 $B_{_3_}(304) = 8 \text{ '2006 } 17$

B__3__(305) = 23 '2006 18
 B__3__(306) = 8 '2006 19
 B__3__(307) = 23 '2006 20
 B__3__(308) = 7 '2006 21
 B__3__(309) = 22 '2006 22
 B__3__(310) = 7 '2006 23
 B__3__(311) = 22 '2006 24
 B__3__(312) = 6 '2007 1
 B__3__(313) = 20 '2007 2
 B__3__(314) = 4 '2007 3
 B__3__(315) = 19 '2007 4
 B__3__(316) = 6 '2007 5
 B__3__(317) = 21 '2007 6
 B__3__(318) = 5 '2007 7
 B__3__(319) = 20 '2007 8
 B__3__(320) = 6 '2007 9
 B__3__(321) = 21 '2007 10
 B__3__(322) = 6 '2007 11
 B__3__(323) = 22 '2007 12
 B__3__(324) = 7 '2007 13
 B__3__(325) = 23 '2007 14
 B__3__(326) = 8 '2007 15
 B__3__(327) = 23 '2007 16
 B__3__(328) = 8 '2007 17
 B__3__(329) = 23 '2007 18
 B__3__(330) = 8 '2007 19
 B__3__(331) = 23 '2007 20
 B__3__(332) = 8 '2007 21
 B__3__(333) = 23 '2007 22
 B__3__(334) = 7 '2007 23
 B__3__(335) = 22 '2007 24
 B__3__(336) = 6 '2008 1
 B__3__(337) = 21 '2008 2
 B__3__(338) = 4 '2008 3
 B__3__(339) = 19 '2008 4
 B__3__(340) = 5 '2008 5
 B__3__(341) = 20 '2008 6
 B__3__(342) = 4 '2008 7
 B__3__(343) = 20 '2008 8
 B__3__(344) = 5 '2008 9
 B__3__(345) = 21 '2008 10
 B__3__(346) = 5 '2008 11
 B__3__(347) = 21 '2008 12
 B__3__(348) = 7 '2008 13
 B__3__(349) = 22 '2008 14
 B__3__(350) = 7 '2008 15

B__3__(351) = 23 '2008 16
B__3__(352) = 7 '2008 17
B__3__(353) = 22 '2008 18
B__3__(354) = 8 '2008 19
B__3__(355) = 23 '2008 20
B__3__(356) = 7 '2008 21
B__3__(357) = 22 '2008 22
B__3__(358) = 7 '2008 23
B__3__(359) = 21 '2008 24
B__3__(360) = 5 '2009 1
B__3__(361) = 20 '2009 2
B__3__(362) = 4 '2009 3
B__3__(363) = 18 '2009 4
B__3__(364) = 5 '2009 5
B__3__(365) = 20 '2009 6
B__3__(366) = 4 '2009 7
B__3__(367) = 20 '2009 8
B__3__(368) = 5 '2009 9
B__3__(369) = 21 '2009 10
B__3__(370) = 5 '2009 11
B__3__(371) = 21 '2009 12
B__3__(372) = 7 '2009 13
B__3__(373) = 23 '2009 14
B__3__(374) = 7 '2009 15
B__3__(375) = 23 '2009 16
B__3__(376) = 7 '2009 17
B__3__(377) = 23 '2009 18
B__3__(378) = 8 '2009 19
B__3__(379) = 23 '2009 20
B__3__(380) = 7 '2009 21
B__3__(381) = 22 '2009 22
B__3__(382) = 7 '2009 23
B__3__(383) = 22 '2009 24
B__3__(384) = 5 '2010 1
B__3__(385) = 20 '2010 2
B__3__(386) = 4 '2010 3
B__3__(387) = 18 '2010 4
B__3__(388) = 5 '2010 5
B__3__(389) = 20 '2010 6
B__3__(390) = 4 '2010 7
B__3__(391) = 20 '2010 8
B__3__(392) = 5 '2010 9
B__3__(393) = 21 '2010 10
B__3__(394) = 5 '2010 11
B__3__(395) = 21 '2010 12
B__3__(396) = 7 '2010 13

```

B__3__(397) = 23 '2010 14
B__3__(398) = 7 '2010 15
B__3__(399) = 23 '2010 16
B__3__(400) = 7 '2010 17
B__3__(401) = 23 '2010 18
B__3__(402) = 8 '2010 19
B__3__(403) = 23 '2010 20
B__3__(404) = 7 '2010 21
B__3__(405) = 22 '2010 22
B__3__(406) = 7 '2010 23
B__3__(407) = 22 '2010 24
' t-birdlo:19950401 - Add for Kumi Speed Up Module
B__4_$(1) = "初一"
B__4_$(2) = "初二"
B__4_$(3) = "初三"
B__4_$(4) = "初四"
B__4_$(5) = "初五"
B__4_$(6) = "初六"
B__4_$(7) = "初七"
B__4_$(8) = "初八"
B__4_$(9) = "初九"
B__4_$(10) = "初十"
B__4_$(11) = "十一"
B__4_$(12) = "十二"
B__4_$(13) = "十三"
B__4_$(14) = "十四"
B__4_$(15) = "十五"
B__4_$(16) = "十六"
B__4_$(17) = "十七"
B__4_$(18) = "十八"
B__4_$(19) = "十九"
B__4_$(20) = "二十"
B__4_$(21) = "廿一"
B__4_$(22) = "廿二"
B__4_$(23) = "廿三"
B__4_$(24) = "廿四"
B__4_$(25) = "廿五"
B__4_$(26) = "廿六"
B__4_$(27) = "廿七"
B__4_$(28) = "廿八"
B__4_$(29) = "廿九"
B__4_$(30) = "三十"
B__5_$(1) = "闰一月"
B__5_$(2) = "闰二月"
B__5_$(3) = "闰三月"
B__5_$(4) = "闰四月"

```

```

B__5__$ (5) = "闰五月"
B__5__$ (6) = "闰六月"
B__5__$ (7) = "闰七月"
B__5__$ (8) = "闰八月"
B__5__$ (9) = "闰九月"
B__5__$ (10) = "闰十月"
B__5__$ (11) = "闰十一"
B__5__$ (12) = "闰十二"
B__6__$ (1) = "正月"
For i = 2 To 12
    B__6__$ (i) = rgstrMonthName(i - 1)
Next i
B__7__$ (0) = "小寒"
B__7__$ (1) = "大寒"
B__7__$ (2) = "立春"
B__7__$ (3) = "雨水"
B__7__$ (4) = "惊蛰"
B__7__$ (5) = "春分"
B__7__$ (6) = "清明"
B__7__$ (7) = "谷雨"
B__7__$ (8) = "立夏"
B__7__$ (9) = "小满"
B__7__$ (10) = "芒种"
B__7__$ (11) = "夏至"
B__7__$ (12) = "小暑"
B__7__$ (13) = "大暑"
B__7__$ (14) = "立秋"
B__7__$ (15) = "处暑"
B__7__$ (16) = "白露"
B__7__$ (17) = "秋分"
B__7__$ (18) = "寒露"
B__7__$ (19) = "霜降"
B__7__$ (20) = "立冬"
B__7__$ (21) = "小雪"
B__7__$ (22) = "大雪"
B__7__$ (23) = "冬至"
End Sub

```

```

Private Sub GetLunarDays(iYear, iMonth)
    Dim StartOf1994Month
    Dim StartOf1994Day
    Dim iDS1994
    Dim iDSAsk
    Dim iFrom1994
    Dim iTotalSim

```

```

Dim iSMName
Dim iSimMonth
Dim fDBMonth
Dim k
Dim iStartSim
Dim i
StartOf1994Month = 11
StartOf1994Day = 20
iDS1994 = DateSerial(1994, 1, 1)
iDSAsk = DateSerial(iYear, iMonth, 1)
iFrom1994 = iDSAsk - iDS1994
iTotalSim = 0
iSMName = StartOf1994Month
While iTotalSim < (iFrom1994 - StartOf1994Day)
    If B__2__(iSimMonth) > 30 Then
        fDBMonth = 1
        iTotalSim = iTotalSim + B__2__(iSimMonth) - 10 'B_2_(0)=30
    Else
        fDBMonth = 0
        iTotalSim = iTotalSim + B__2__(iSimMonth)
        iSMName = iSMName + 1
        If iSMName > 12 Then iSMName = 1
    End If
    iSimMonth = iSimMonth + 1
Wend
If B__2__(iSimMonth) > 30 Then
    k = B__2__(iSimMonth) - 10
    iSMName = iSMName - 1
Else
    k = B__2__(iSimMonth)
End If
iStartSim = StartOf1994Day + (iFrom1994 - iTotalSim)
If iStartSim > k Then
    iStartSim = iStartSim Mod k
    iSimMonth = iSimMonth + 1
    If B__2__(iSimMonth) > 30 Then
        fDBMonth = 1
        k = B__2__(iSimMonth) - 10
    Else
        fDBMonth = 0
        k = B__2__(iSimMonth)
        If B__2__(iSimMonth) < 31 Then iSMName = iSMName + 1
        If iSMName > 12 Then iSMName = 1
    End If
End If
lunYeay = iYear

```

```

For i = 0 To 30
    If iStartSim = 1 Then
        If fDBMonth = 1 Then
            IntToSimDay__(i, 2) = B__5__(iSMName)
            IntToSimDay__(i, 3) = B__4__(iStartSim)
            IntToSimDay__(i, 4) = GetLunarSections(i, iYear, iMonth)
            IntToSimDay__(i, 0) = lunCalYear(i, iYear, iMonth)
            IntToSimDay__(i, 1) = TwelveAnimals(i, iYear, iMonth)
        Else
            IntToSimDay__(i, 2) = B__6__(iSMName)
            IntToSimDay__(i, 3) = B__4__(iStartSim)
            IntToSimDay__(i, 4) = GetLunarSections(i, iYear, iMonth)
            IntToSimDay__(i, 0) = lunCalYear(i, iYear, iMonth)
            IntToSimDay__(i, 1) = TwelveAnimals(i, iYear, iMonth)

        End If
        iStartSim = iStartSim + 1
    Else
        If iStartSim > k Then
            iSimMonth = iSimMonth + 1
            iStartSim = 1
            i = i - 1
            If B__2__(iSimMonth) > 30 Then
                k = B__2__(iSimMonth) - 10
                fDBMonth = 1
            Else
                k = B__2__(iSimMonth)
                fDBMonth = 0
                iSMName = iSMName + 1
                If iSMName > 12 Then iSMName = 1
            End If
            If iSMName = 1 Then
                lunYeay = iYear + 1
            Else
                lunYeay = iYear
            End If
        Else
            IntToSimDay__(i, 2) = B__6__(iSMName)
            IntToSimDay__(i, 3) = B__4__(iStartSim)
            IntToSimDay__(i, 4) = GetLunarSections(i, iYear, iMonth)
            IntToSimDay__(i, 0) = lunCalYear(i, iYear, iMonth)
            IntToSimDay__(i, 1) = TwelveAnimals(i, iYear, iMonth)

            iStartSim = iStartSim + 1
        End If
    End If

```

```

        End If
    Next
    '    GetLunarSections iYear, iMonth
End Sub

```

```

Function GetLunarSections(i, iYear, iMonth)
    Dim iSimSection
    Dim j
    iSimSection = (iYear - 1994) * 24 + (iMonth - 1) * 2
    j = B__3__(iSimSection)
    If i = j Then
        GetLunarSections = B__7__$((iMonth - 1) * 2)
        Exit Function
    Else
        GetLunarSections = ""
    End If
    j = B__3__(iSimSection + 1)
    If j = i Then
        GetLunarSections = B__7__$((iMonth - 1) * 2 + 1)
        Exit Function
    Else
        GetLunarSections = ""
    End If
End Function

```

```

Function lunCalYear(i, iYear, iMonth) 'lunarCalendarYear(天干地支年)
    Dim Gan()
    Dim Zhi()
    Gan = Array("甲", "乙", "丙", "丁", "戊", "己", "庚", "辛", "壬", "癸")
    Zhi = Array("子", "丑", "寅", "卯", "辰", "巳", "午", "未", "申", "酉", "戌",
"亥")
    Y = iYear
    If iMonth = 1 Or iMonth = 2 Then
        If IntToSimDay__(i, 2) = "十一月" Or IntToSimDay__(i, 2) = "十二月" Then
            Y = Y - 1
        End If
    End If
    While (Y - 1904) >= 10 '天干
        Y = Y - 10
    Wend
    rGan = Gan(Y - 1904)

    Y = iYear
    If iMonth = 1 Or iMonth = 2 Then

```

```

        If IntToSimDay__(i, 2) = "十一月" Or IntToSimDay__(i, 2) = "十二月" Then
            Y = Y - 1
        End If
    End If
    While (Y - 1900) >= 12 '地支
        Y = Y - 12
    Wend
    rZhi = Zhi(Y - 1900)
    lunCalYear = rGan & rZhi & "年"
End Function

```

```

Function TwelveAnimals(i, iYear, iMonth) '十二生肖年
    Dim Ani()
    Ani = Array("鼠", "牛", "虎", "兔", "龙", "蛇", "马", "羊", "猴", "鸡", "狗",
"猪")
    Y = iYear
    If iMonth = 1 Or iMonth = 2 Then
        If IntToSimDay__(i, 2) = "十一月" Or IntToSimDay__(i, 2) = "十二月" Then
            Y = Y - 1
        End If
    End If
    While (Y - 1900) >= 12
        Y = Y - 12
    Wend

    TwelveAnimals = "[" & Ani(Y - 1900) & "]"
End Function

```

'#####

'12.函数作用:返回指定列数的列标

' 参数说明: pureNum 为 1-256 之间的整数

'#####

```

Public Function NumToChr(PureNum As Integer) As String
    If PureNum Mod 26 = 0 Then
        NumToChr = VBA.IIf(PureNum \ 26 = 1, "", VBA.Chr(PureNum \ 26 + 63)) & "Z"
    Else
        If PureNum <= 256 Then
            NumToChr = VBA.IIf(PureNum \ 26 = 0, "", Chr(PureNum \ 26 + 64)) &
Chr(PureNum Mod 26 + 64)
        Else
            NumToChr = "超出范围"
        End If
    End If
End Function

```

```
MsgBox "当前 EXCEL 版本只有 256 列,你输入的列数不存在.请在 1-256 之间选取数字"
```

```
End If
End If
End Function
```

```
'#####
```

```
'13.函数作用:用指定字符替换某字符
```

```
'#####
```

```
Public Function ReplaceIt(OriginalStr As String, SearchStr As String, ToBeReplaced As String) As String
```

```
Dim FoundPos As Integer
```

```
Do While VBA.InStr(1, OriginalStr, SearchStr) <> 0
```

```
FoundPos = VBA.InStr(1, OriginalStr, SearchStr)
```

```
OriginalStr = VBA.Left(OriginalStr, FoundPos - 1) & ToBeReplaced &
```

```
VBA.Mid(OriginalStr, (FoundPos + VBA.Len(SearchStr)))
```

```
Loop
```

```
ReplaceIt = OriginalStr
```

```
End Function
```

```
'#####
```

```
'14.函数作用:从右边开始查找指定字符在字符串中的位置
```

```
'#####
```

```
Public Function MyInStrRev(MainStr As String, SubStr As String) As Integer
```

```
Dim Counter As Integer
```

```
Dim Success As Boolean
```

```
If VBA.Len(MainStr) < VBA.Len(SubStr) Then
```

```
MyInStrRev = 0
```

```
Else
```

```
For Counter = VBA.Len(SubStr) To VBA.Len(MainStr)
```

```
If VBA.Left(VBA.Right(MainStr, Counter), VBA.Len(SubStr)) = SubStr
```

```
Then
```

```
Success = True
```

```
Exit For
```

```
End If
```

```
Next Counter
```

```
If Success Then
```

```

        MyInStrRev = VBA.Len(MainStr) - Counter + 1
    Else
        MyInStrRev = 0
    End If
End If
End Function

```

```

'#####

```

'15.函数作用:从右边开始查找指定字符在字符串中的位置

```

'#####

```

```

Public Function MyInStrRev(MainStr As String, SubStr As String) As Integer
    Dim Counter As Integer
    Dim Success As Boolean
    If VBA.Len(MainStr) < VBA.Len(SubStr) Then
        MyInStrRev = 0
    Else
        For Counter = VBA.Len(SubStr) To VBA.Len(MainStr)
            If VBA.Left(VBA.Right(MainStr, Counter), VBA.Len(SubStr)) = SubStr
Then
                Success = True
                Exit For
            End If
        Next Counter
        If Success Then
            MyInStrRev = VBA.Len(MainStr) - Counter + 1
        Else
            MyInStrRev = 0
        End If
    End If
End Function

```

```

'#####

```

'16.函数作用:计算工龄

```

'#####

```

```

Function Elapsed(StartDate As Date, EndDate As Date, ReturnType As Integer)
    Dim StartYear As Integer '定义变量用以参数中开始日期的计算
    Dim StartMonth As Integer

```

```

Dim StartDay As Integer
Dim EndYear As Integer '定义变量用以参数中结束日期的计算
Dim EndMonth As Integer
Dim EndDay As Integer
StartYear = Year(StartDate) '从参数中获取开始日期和结束日期的年数，月数，天数
StartMonth = Month(StartDate)
StartDay = Day(StartDate)
EndYear = Year(EndDate)
EndMonth = Month(EndDate)
EndDay = Day(EndDate)
If EndDay < StartDay Then '如果结束日期参数的天数小于开始日期中的天数，则...
    EndDay = EndDay + (DateSerial(EndYear, EndMonth + 1, EndDay) -
DateSerial(EndYear, EndMonth, EndDay))
    EndMonth = EndMonth - 1 '...从月数中借 1 后再进行减运算，从而得到相关天数
End If
If EndMonth < StartMonth Then '如果结束日期参数的月数小于开始日期参数中的月数，
    EndMonth = EndMonth + 12
    EndYear = EndYear - 1 '从年数中借 1 后再进行减运算，从而得到相差月数
End If
Select Case ReturnType '如果没有以上特殊情况，则直接进行相减的运算
    Case 1 '返回年数
        Elapsed = EndYear - StartYear
    Case 2 '返回月数
        Elapsed = EndMonth - StartMonth
    Case 3 '返回天数
        Elapsed = EndDay - StartDay
End Select
End Function

```

'#####

'17.函数作用:计算日期差，除去星期六、星期日

'#####

```

Function daydif(x As Range, y As Range)

```

```

    Dim date1, date2 As Date
    date1 = x
    date2 = y

    dif = 0
    Do
        If (date1 >= date2) Then
            Exit Do

```

```

        End If

        date1 = date1 + 1
        t1 = Weekday(date1)
        If (t1 < 7 And t1 > 1) Then
            dif = dif + 1
        End If
    Loop
    daydif = dif

End Function

```

```

'#####

```

'18.函数作用:将英文字反转的自定函数.

```

'#####

```

```

Function TextReverse(sSource As String) As String
    Dim iCounter As Integer
    Dim sText As String
    For iCounter = Len(sSource) To 1 Step -1
        sText = sText & Mid(sSource, iCounter, 1)
    Next
    TextReverse = sText
End Function

```

```

'#####

```

'19.函数作用:计算个人所得税

' 参数说明:q:应纳税所得额
 ' w:为扣除额,可自定义,如 800
 ' 使用说明:如 a1 为应纳税所得额,直接在单元格输入“=sds(a1,800)”,也可以是
 “=sds(a1,b1))”

' 如果扣除额不是 800,可自己改数字,也可以是单元格
 '#####

```

Public Function sds(q, w)

    je = q - w
    If q < w Then
        'msgbox("应纳税所得额必须大于或等于扣除额!")
    End If

```

```

sds = 0
ElseIf je <= 500 Then
    sds = je * 0.05
ElseIf je > 500 And je <= 2000 Then
    sds = je * 0.1 - 25
ElseIf je > 2000 And je <= 5000 Then
    sds = je * 0.15 - 125
ElseIf je > 5000 And je <= 20000 Then
    sds = je * 0.2 - 375
ElseIf je > 20000 And je <= 40000 Then
    sds = je * 0.25 - 1375
ElseIf je > 40000 And je <= 60000 Then
    sds = je * 0.3 - 3375
ElseIf je > 60000 And je <= 80000 Then
    sds = je * 0.35 - 6375
ElseIf je > 80000 And je <= 100000 Then
    sds = je * 0.4 - 10375
Else
    sds = je * 0.45 - 15375
End If

```

End Function

'#####

'20.函数作用:一个能计算是否有重复单元的函数

'#####

Function IsRepeate(c As Range) As Boolean

```

Dim cell As Range
Dim SumC As Integer
Dim CountBlank As Integer
SumC = 0
CountBlank = 0
For Each cell In c
    If VBA.IsEmpty(cell) Then
        CountBlank = CountBlank + 1
    Else
        SumC = SumC + 1 / WorksheetFunction.CountIf(c, cell)
    End If
Next cell
If SumC = c.Count - CountBlank And c.Count > CountBlank Then '不重复的话就返
回 FALSE

```

```

        IsRepeate = False

Else '重复的话就返回 TRUE
    IsRepeate = True

End If

End Function

'#####

'21.数字金额转中文大写

'#####

Function DaXie(ByVal Num)
    Application.Volatile True
    Place = "分角元拾佰仟万拾佰仟亿拾佰仟万"
    Dn = "壹贰叁肆伍陆柒捌玖"
    D1 = "整零元零零零万零零零亿零零零万"
    Num = Format(Abs(Num), "###0.00") * 100
    If Num > 999999999999999# Then
        DaXie = "数字超出转换范围!!"
        Exit Function
    End If
    If Num = 0 Then
        DaXie = "零元零分"
        Exit Function
    End If
    If Num < 0 Then FuHao = "（负）"
    NumA = Trim(Str(Num))
    NumLen = Len(NumA)
    For J = NumLen To 1 Step -1 ' 数字转换过程
        Temp = Val(Mid(NumA, NumLen - J + 1, 1))
        If Temp <> 0 Then ' 非零数字转换
            NumC = NumC & Mid(Dn, Temp, 1) & Mid(Place, J, 1)
        Else ' 数字零的转换
            If Right(NumC, 1) <> "零" Then
                NumC = NumC & Mid(D1, J, 1)
            Else
                Select Case J ' 特殊数位转换
                    Case 1
                        NumC = Left(NumC, Len(NumC) - 1) & Mid(D1, J, 1)
                    Case 3, 11
                        NumC = Left(NumC, Len(NumC) - 1) & Mid(D1, J, 1) & "零"
                End Select
            End If
        End If
    Next J
    If FuHao <> "" Then NumC = FuHao & NumC
    If NumC <> "" Then
        If Right(NumC, 1) = "元" Then
            If Right(NumC, 2) = "零元" Then
                NumC = Left(NumC, Len(NumC) - 2)
            End If
        End If
    End If
    DaXie = NumC
End Function

```

```

        Case 7
            If Mid(NumC, Len(NumC) - 1, 1) <> "亿" Then
                NumC = Left(NumC, Len(NumC) - 1) & Mid(D1, J, 1) & "
零"

            End If
        Case Else
        End Select
    End If
End If
Next
DaXie = FuHao & Trim(NumC)
End Function

```

```

'#####

```

'22.函数作用:将数字转成英文

```

'#####

```

```

'*****' Main Function *'*****

```

```

Function SpellNumber(ByVal MyNumber)
    Dim Dollars, Cents, Temp
    Dim DecimalPlace, Count
    ReDim Place(9) As String
    Application.Volatile True
    Place(2) = " Thousand "
    Place(3) = " Million "
    Place(4) = " Billion "
    Place(5) = " Trillion " ' String representation of amount
    MyNumber = Trim(Str(MyNumber)) ' Position of decimal place 0 if none
    DecimalPlace = InStr(MyNumber, ".")
    'Convert cents and set MyNumber to dollar amount
    If DecimalPlace > 0 Then
        Cents = GetTens(Left(Mid(MyNumber, DecimalPlace + 1) & "00", 2))
        MyNumber = Trim(Left(MyNumber, DecimalPlace - 1))
    End If
    Count = 1
    Do While MyNumber <> ""
        Temp = GetHundreds(Right(MyNumber, 3))
        If Temp <> "" Then Dollars = Temp & Place(Count) & Dollars
        If Len(MyNumber) > 3 Then
            MyNumber = Left(MyNumber, Len(MyNumber) - 3)
        Else
            MyNumber = ""

```

```

        End If
        Count = Count + 1
    Loop
    Select Case Dollars
        Case ""
            Dollars = "No Dollars"
        Case "One"
            Dollars = "One Dollar"
        Case Else
            Dollars = Dollars & " Dollars"
    End Select
    Select Case Cents
        Case ""
            Cents = " and No Cents"
        Case "One"
            Cents = " and One Cent"
        Case Else
            Cents = " and " & Cents & " Cents"
    End Select
    SpellNumber = Dollars & Cents
End Function

'*****
' Converts a number from 100-999 into text *
'*****

Function GetHundreds(ByVal MyNumber)
    Dim Result As String
    If Val(MyNumber) = 0 Then Exit Function
    MyNumber = Right("000" & MyNumber, 3) 'Convert the hundreds place
    If Mid(MyNumber, 1, 1) <> "0" Then
        Result = GetDigit(Mid(MyNumber, 1, 1)) & " Hundred "
    End If
    'Convert the tens and ones place
    If Mid(MyNumber, 2, 1) <> "0" Then
        Result = Result & GetTens(Mid(MyNumber, 2))
    Else
        Result = Result & GetDigit(Mid(MyNumber, 3))
    End If
    GetHundreds = Result
End Function

'*****
' Converts a number from 10 to 99 into text. *
'*****

```

```

Function GetTens(TensText)
    Dim Result As String
    Result = "" 'null out the temporary function value
    If Val(Left(TensText, 1)) = 1 Then ' If value between 10-19
        Select Case Val(TensText)
            Case 10
                Result = "Ten"
            Case 11
                Result = "Eleven"
            Case 12
                Result = "Twelve"
            Case 13
                Result = "Thirteen"
            Case 14
                Result = "Fourteen"
            Case 15
                Result = "Fifteen"
            Case 16
                Result = "Sixteen"
            Case 17
                Result = "Seventeen"
            Case 18
                Result = "Eighteen"
            Case 19
                Result = "Nineteen"
            Case Else
        End Select
    Else ' If value between 20-99
        Select Case Val(Left(TensText, 1))
            Case 2
                Result = "Twenty "
            Case 3
                Result = "Thirty "
            Case 4
                Result = "Forty "
            Case 5
                Result = "Fifty "
            Case 6
                Result = "Sixty "
            Case 7
                Result = "Seventy "
            Case 8
                Result = "Eighty "
            Case 9
                Result = "Ninety "
            Case Else

```

```

        End Select
        Result = Result & GetDigit _
            (Right(TensText, 1)) 'Retrieve ones place
    End If
    GetTens = Result
End Function

```

```

'*****
' Converts a number from 1 to 9 into text. *
'*****

```

```

Function GetDigit(Digit)
    Select Case Val(Digit)
        Case 1
            GetDigit = "One"
        Case 2
            GetDigit = "Two"
        Case 3
            GetDigit = "Three"
        Case 4
            GetDigit = "Four"
        Case 5
            GetDigit = "Five"
        Case 6
            GetDigit = "Six"
        Case 7
            GetDigit = "Seven"
        Case 8
            GetDigit = "Eight"
        Case 9
            GetDigit = "Nine"
        Case Else
            GetDigit = ""
    End Select
End Function

```

```

'#####

```

'23.函数作用:人民币大小写转换

```

'#####

```

```

Function NtoC(ByVal n) As String 'n as Currency
    Const cNum As String = "零壹贰叁肆伍陆柒捌玖-万仟佰拾亿仟佰拾万仟佰拾元角分"

```

```

Const cCha As String = "零仟零佰零拾零零零零零亿零万零元亿万零角零分零整-零零零
零零亿万元亿零整整"
Dim sNum As String
Dim i As Long

If (n <> 0) And (Abs(n) < 10000000000000#) Then
    sNum = Trim(Str(Int(Abs(n) * 100)))
    For i = 1 To Len(sNum) '逐位转换
        NtoC = NtoC + Mid(cNum, (Mid(sNum, i, 1)) + 1, 1) + Mid(cNum, 26 - Len(sNum)
+ i, 1)
    Next
    For i = 0 To 11 '去掉多余的零
        NtoC = Replace(NtoC, Mid(cCha, i * 2 + 1, 2), Mid(cCha, i + 26, 1))
    Next
    If n < 0 Then NtoC = "(负)" + NtoC
Else
    NtoC = IIf(n = 0, "零元", "溢出")
End If
End Function

```

```

'#####

```

'24.函数作用:获取区域颜色值

```

'#####

```

```

Function ColorID(ReColor As Range) As Integer
    Application.Volatile
    ColorID = ReColor.Interior.ColorIndex
End Function

```

```

'#####

```

'25.函数作用:获取活动工作表名

```

'#####

```

```

Public Function sh_name() As String
    sh_name = ActiveSheet.Name
End Function

```

```

'#####

```

'26.函数作用:获取最后一行行数

'#####

```
Function Myrange()  
    Myrange = Worksheets("数据表").[B65536].End(xlUp).Row  
End Function
```

'#####

'27.函数作用:判断是否连接在线

'#####

```
Public Declare Function InternetGetConnectedState _  
    Lib "wininet.dll" (lpdwFlags As Long, _  
    ByVal dwReserved As Long) As Boolean  
  
Function IsConnected() As Boolean  
    Dim Stat As Long  
    IsConnected = (InternetGetConnectedState(Stat, 0) <> 0)  
End Function
```

```
Sub Test()  
    ' Randy Birch  
    If IsConnected = True Then  
        MsgBox "Copy your mail code here"  
    Else  
        MsgBox "You can't use this subroutine because you are not online"  
    End If  
End Sub
```

'#####

'28.函数作用:币种转换

'#####

```
Function curr(curr_code, price)  
    If curr_code = "RMB" Then  
        curr = Application.WorksheetFunction.Round(price / 1.06, 2)  
    ElseIf curr_code = "HKD$" Or curr_code = "HKD" Then
```

```

        curr = price
    ElseIf curr_code = "USD$" Or curr_code = "USD" Then
        curr = Application.WorksheetFunction.Round(price * 7.8, 2)
    ElseIf curr_code = "YEN$" Or curr_code = "YEN" Then
        curr = Application.WorksheetFunction.Round(price * 0.065, 2)
    End If

End Function

'#####

'29.函数作用:检验工作表是否有可打印内容

'#####

Function IsSheetEmpty(sheet As Worksheet) As Boolean

    Dim cellTxt As String
    Dim lastCellRange As Range
    Dim tempCellRange As Range
    Dim lastCol As Long
    Dim icol As Long
    Dim nCells As Long
    Dim falseVariant As Variant
    Dim mergeAreaVariant As Variant
    Dim retVal As Boolean
    Dim colorIndx As Variant
    Dim shapeCount As Long
    Dim printableShapes As Boolean
    Dim ishape As Long

    falseVariant = False
    IsSheetEmpty = False
    retVal = False

    '    Set lastCellRange = sheet.Cells.HPageecialCells(xlCellTypeLastCell)

    If ((sheet.HPageBreaks.Count <> 0) Or (sheet.VPageBreaks.Count <> 0)) Then
        Exit Function
    End If
    If (sheet.PageSetup.PrintArea <> "") Then
        Exit Function
    End If

    nCells = sheet.UsedRange.Cells.Count

```

```

Set lastCellRange = sheet.UsedRange.Cells(nCells)
lastCol = lastCellRange.Column
cellTxt = lastCellRange.Text
If (Len(cellTxt) = 0) Then
    For icol = 1 To lastCol
        Set tempCellRange = sheet.Cells(1, icol)
        cellTxt = tempCellRange.Text
        If (Len(cellTxt) = 0) Then
            cellTxt = tempCellRange.End(xlDown).Text
            If (Len(cellTxt) <> 0) Then
                Exit Function
            End If
        Else
            Exit Function
        End If
    Next icol
Else
    Exit Function
End If

'maddy
shapeCount = sheet.Shapes.Count
printableShapes = False
If (shapeCount) Then
    For ishape = 1 To shapeCount
        If (sheet.Shapes(ishape).Type = msoComment) Then
            If ((sheet.PageSetup.PrintComments <> xlPrintNoComments) And (Not
((Not (sheet.Shapes(ishape).Visible)) And (sheet.PageSetup.PrintComments =
xlPrintInPlace)))) Then
                printableShapes = True
            Exit For
        End If
    Else
        If (sheet.Shapes(ishape).ControlFormat.PrintObject) Then
            printableShapes = True
        End If
    End If
    Next ishape
End If
'if the sheet has merged cells, then it is non-empty
If (Not printableShapes) Then
    mergeAreaVariant = sheet.Cells.MergeCells
    If (mergeAreaVariant = falseVariant) Then
        retVal = True
    End If
End If

```

```

'if the sheet has colored cells then it is non-empty
If (retVal = True) Then
    colorIndx = sheet.UsedRange.Interior.ColorIndex
    If (IsNull(colorIndx) Or (Not (colorIndx = xlColorIndexNone))) Then
        retVal = False
    End If
End If

IsSheetEmpty = retVal
End Function

```

```

'#####

```

'30.函数作用:查找一字符串(withinstr)在另一字符串中(findstr1)中某一次(startnum)出现

时的位置, 返回零表示没找到。

```

' 使用方法: 如 Findstr("IloveVBA VeryMuch,VBAisMylove","VBA",1),返回结果为 6;
'           Findstr("IloveVBAVeryMuch,VBAisMylove","VBA",2),返回结果为 18。
'#####

```

```

Public Function findstr(ByVal findstr1 As String, withinstr As String, startnum
As Integer) As Integer

```

```

    Dim i As Integer
    Dim finded As Integer

    finded = 0
    For i = 1 To Len(findstr1) - Len(withinstr) + 1
        If Mid(findstr1, i, Len(withinstr)) = withinstr Then
            finded = finded + 1
            If finded = startnum Then
                findstr = i
                Exit Function
            End If
        Else
            If i = startnum Then
                findstr = 0
                Exit Function
            End If
        End If
    Next i

```

End Function

'#####

'31.函数作用:增加文件路径最后的“\”符号

'#####

```
Private Function EndPath(sInstring As String) As String
    'Make sure that the path of file is end with a "\"
    If Right(sInstring, 1) <> "\" Then sInstring = sInstring & "\"
    EndPath = sInstring
End Function
```

'#####

'32.函数作用:计算所得税

' 使用说明:直接填在单元格里就可以用了
' 收入填到 A2 中,起征金额填到 B1 中(为方便复制,已做绝对引用)

= ROUND(If(A2<\$B\$1, 0, If(A2 - \$B\$1<500, (A2 - \$B\$1) * 0.05, If(A2 - \$B\$1<2000, (A2 - \$B\$1) * 0.1 -25, If(A2 - \$B\$1<5000, (A2 - \$B\$1) * 0.15 -125, If(A2 - \$B\$1<20000, (A2 - \$B\$1) * 0.2 -375, "太累了, 自己看着加吧"))))), 2)

'#####

'33.函数作用:从工作表第一行的标题文字以数字形式返回所在列号

' 使用示例:姓名 col = 从列标题名称获取列号数("人事档案", "姓名")
' 如果是"人事档案"为当前工作表,上式可写成:
' 姓名 col = 从列标题名称获取列号数("", "姓名")

Private Function 从列标题名称获取列号数(thisSheetName\$, thisTitle\$) As Long

'约定标题在第一列, A 1 起, 无间断

```
Dim c As Integer
Dim tf As Boolean
```

```

从列标题名称获取列号数 = 0
For c = 1 To 255
    If thisSheetName$ = "" Then
        tf = Cells(1, c) = thisTitle$
    Else
        tf = Sheets(thisSheetName$).Cells(1, c) = thisTitle$
    End If
    If tf Then
        从列标题名称获取列号数 = c
        Exit For
    End If
Next c

If 从列标题名称获取列号数 = 0 Then
    MsgBox "在工作表“" & thisSheetName$ & "”中没有找到标题为[" & thisTitle$ & "]"
    的列，程序终止", vbokly + vbCritical
End
End If

End Function

```

'#####

'34.函数作用:在多个工作表中查找一个范围内符合某个指定条件的项目对应指定范围加总求和

```

' 参数说明:Rang:要查找的范围
'          critreia:符合条件的标准
'          sum_range:要加总的范围
'#####

```

```

Function SumIfAllSheets(rang As Range, Criteria As Variant, sum_range As Range)
    Dim wSheet As Worksheet
    Dim vSum
    On Error Resume Next

    For Each wSheet In ActiveWorkbook.Worksheets
        With wSheet
            Set rang = .Range(rang.Address)
            Set sum_range = .Range(sum_range.Address)
            vSum = vSum + WorksheetFunction.SumIf(rang, Criteria, sum_range)
        End With
    Next wSheet

    Set rang = Nothing

```

```

    Set sum_range = Nothing
    SumIfAllSheets = vSum
End Function

```

```

'#####

```

```

'35.函数作用:返回 Column 英文字

```

```

'#####

```

```

Function ColIntToLetter(intCol As Integer) As String
    ,

    Dim intPart As Integer
    Dim intRemainder As Integer

    If intCol > 255 Or intCol <= 0 Then
        MsgBox ("The Wrong Column Number: " & CStr(intCol))
        Exit Function
    End If

    intPart = intCol \ 26
    intRemainder = intCol Mod 26

    If intPart = 0 Then
        ColIntToLetter = Chr(intCol + 64)
    ElseIf intPart = 1 And intRemainder = 0 Then
        ColIntToLetter = "Z"
    ElseIf intRemainder = 0 Then
        ColIntToLetter = Chr(intPart - 1 + 64) & "Z"
    Else
        ColIntToLetter = Chr(intPart + 64) & Chr(intRemainder + 64)
    End If

End Function

```

```

'#####

```

```

'36.函数作用:查找指定列名的列数

```

```

'#####

```

```

Function FindColumnNumber(strTmp As String, strsheet As String) As Integer
    ' strSheet is the name of the sheet

```

```

' strTmp is the name of this column
Dim Tmp As String

Sheets(strsheet).Select
strTmp = LCase(strTmp)
FindColumnNumber = 0

For j = 1 To 255
    Tmp = Sheets(strsheet).Cells(1, j).Value
    Tmp = LCase(Tmp)
    If Tmp = strTmp Then
        FindColumnNumber = j
        Exit For
    End If
Next j

If FindColumnNumber = 0 Then
    MsgBox ("Can't find this column: " & strTmp)
End
End If
End Function

```

```

'#####

```

'37.函数作用:文字格式的时间(分:秒)转化为数字格式(秒)

```

'#####

```

```

Function TxtSecondToNumber(strTxt As String)
    ' The format of strTxt is mm:ss.??
    Dim iFirst As Integer
    strTxt = Trim(strTxt)
    iFirst = InStr(1, strTxt, ":")
    If iFirst > 0 Then
        TxtSecondToNumber = Val(Left(strTxt, iFirst - 1)) * 60 + _
            Val(Mid(strTxt, iFirst + 1))
    Else
        TxtSecondToNumber = Val(strTxt)
    End If
End Function

```

```

'#####

```

'38.函数作用:将"hh:mm:ss"格式的时分秒数转换成秒数

'#####

```
Public Function tom (str1 As String) As Single
    Dim f1 As Integer, f2 As Integer
    Dim tom1 As Single, tom2 As Single
    Tom1 = 0
    tom2 = 0
    Str1 = Trim (str1)
    f1 = InStr (1, str1, ":")
    f2 = InStr(f1 + 1, str1, ":")
    tom1 = Val(str1) * 3600
    tom2 = Val(Mid(str1, f1 + 1, f2 - 1)) * 60 + Val(Mid(str1, f2 + 1))
    tom = tom1 + tom2
End Function
```

'#####

'39.函数作用:金额中文大写转数字

'#####

```
Function SuZi(A As String) ' 人民币中文大写转数字函数
    Application.Volatile True
    Hsf = "分角元拾佰仟万 亿"
    Hs = "零壹贰叁肆伍陆柒捌玖 "
    JH = 1
    A = Replace(A, "整", "")
    A = Replace(A, "亿", ")亿")
    A = Replace(A, "万", ")万")
    If A <> "" Then
        Mylen = Len(A$)
        For m = 1 To Mylen
            If Mid(A, m, 1) = "万" And JH = 1 Then
                A = "(" & A
                JH = 0
            End If
            If Mid(A, m, 1) = "亿" Then
                A = "(" & A
                JH = 0
            End If
            For K = m + 3 To Mylen + 2
                If Mid(A$, K, 1) = "万" Then
```

```

        A = Replace(A, "亿", "亿(")
    Exit For
End If
Next
Exit For
End If
Next
For i = 0 To 10
    A = Replace(A, Mid(Hs, i + 1, 1), i)
    A = Replace(A, Mid(Hsf, i + 1, 1), "*" & (10 ^ (i - 2)) & "+")
Next
A = Replace(A, "+)", ")")
A = Replace(A, "+*", "*")
Mylen = Len(A)
A = Left(A, Mylen - 1)
SuZi = Evaluate(A)
End If
End Function

```

'#####

'40.函数作用:把角度转为度秒分、弧度等显示

'#####

```

Function degtodms(my_degree)
    Dim a1 As Double
    Dim dms As Integer
    Dim Minute, Second As Double
    a1 = my_degree
    dms = Fix(a1)
    Minute = Fix((a1 - dms) * 60)
    Second = Round((a1 - dms - (Minute / 60)) * 3600)
    degtodms = (dms + Minute / 100 + Second / 10000)
End Function

```

```

Function pitodms(hudu)
    Dim deg As Double
    deg = hudu * 180 / pi()
    pitodms = degtodms(deg)
End Function

```

```

Function pi()
    pi = 3.14159265358979
End Function

```

```
Function distance(x1, Y1, x2, y2)
    distance = Sqr((x2 - x1) * (x2 - x1) + (y2 - Y1) * (y2 - Y1))
End Function
```

```
Function dmstopi(dms)
    Dim a, Minute, Second As Double
    Dim deg As Integer
    a = (dms)
    deg = Fix(a)
    Minute = Fix((a - deg) * 100)
    Second = ((a - deg) * 10000 - Minute * 100)
    dmstopi = (deg + Minute / 60 + Second / 3600) * pi() / 180
End Function
```

```
'#####
```

'41.函数作用:身份证号码侦测

```
'#####
```

```
Public Function xfz(sid, xb) ' As Currency
    '1、身份证不满 15 位, 2、性别与身份证不符, 3、出生月份出错(不在 1-12)
    '4、出生日期出错(不在 1-31 范围内), 5、18 位校验位出错, 6、18 位身份证年份出借
    On Error Resume Next
    Dim s1, s2, jym, x
    If xb = 1 Then x = 1
    If xb = "男" Then x = 1
    If xb = 2 Then x = 0
    If xb = "女" Then x = 0
    s1 = " 7 910 5 8 4 2 1 6 3 7 910 5 8 4 2"
    s2 = "10x98765432"
    If Len(sid) <> 15 And Len(sid) <> 18 Then
        xfz = "身份证位数错误"
        '测试 15 位身份证的信息
    ElseIf Len(sid) = 15 And Val(Mid(sid, 7, 2)) < 10 Then
        xfz = "年龄好大, 请多多保重!"
    ElseIf Len(sid) = 15 And Val(Mid(sid, 9, 2)) > 12 Then
        xfz = "出生月份错误!"
    ElseIf Len(sid) = 15 And Val(Mid(sid, 11, 2)) > 31 Then
        xfz = "出生日期错误!"
    ElseIf Len(sid) = 15 And Mid(sid, 15, 1) Mod 2 <> x Then
        xfz = "性别错误!"
    ElseIf Len(sid) = 15 Then
        newid = Left(sid, 6) + "19" + Right(sid, 9)
```

```

jym = 0
For i = 1 To 17
    jym = jym + Val(Mid(s1, i * 2 - 1, 2)) * Val(Mid(newid, i, 1))
Next i
xfz = newid + Mid(s2, jym Mod 11 + 1, 1)
'测试 18 位身份证的信息
ElseIf Len(sid) = 18 And Val(Mid(sid, 7, 2)) <> 19 Then
    xfz = "出生年错误!"
ElseIf Len(sid) = 18 And Val(Mid(sid, 9, 2)) < 10 Then
    xfz = "年龄好大, 请多多保重!"
ElseIf Len(sid) = 18 And Val(Mid(sid, 11, 2)) > 12 Then
    xfz = "出生月份错误!"
ElseIf Len(sid) = 18 And Val(Mid(sid, 13, 2)) > 31 Then
    xfz = "出生日期错误!"
ElseIf Len(sid) = 18 And Mid(sid, 17, 1) Mod 2 <> x Then
    xfz = "性别错误!"
Else
    newid = Left(sid, 17)
    jym = 0
    For i = 1 To 17
        jym = jym + Val(Mid(s1, i * 2 - 1, 2)) * Val(Mid(newid, i, 1))
    Next i
    If Mid(s2, jym Mod 11 + 1, 1) <> Mid(sid, 18, 1) Then
        xfz = "识别码错, 应为: " & Mid(s2, jym Mod 11 + 1, 1)
    Else
        xfz = ""
    End If
End If
End Function

```

'#####

'42.函数作用:显示公式

' 说明:假如 A1 的公式为 = B1 + C1,
 ' 在 A2 输入公式 = xsGs(A1, True), 显示值为 = B1 + C1
 ' 在 A2 输入公式 = xsGs(A1, False), 显示值为 = RC[1] + RC[2]
 '#####

```

Function xsGs(Vcell As Range, TrueOrFalse As String)
    If Left(Vcell.FormulaR1C1, 1) = "=" Then
        If TrueOrFalse = "True" Then
            xsGs = Vcell.Formula Else
            xsGs = Vcell.FormulaR1C1
        End If
    End If

```

```

Else
    xsgs = "nothing"
End If
End Function

```

```

'#####

```

'43.函数作用:方便财务人员理帐查找

' 说明:searchit (金额, 构对 参数[比如是款项性质, 对方单位, 某某人的, 可以多项, 自己增加])

' 返回的是查找到的行次。

```

'#####

```

```

Function searchit(need_search As Range, overit As Range, p_1 As Range, p_2 As Range,
p_3 As Range, p_4 As Range) As Integer
    Dim money, dm, jfn, jyq As Integer
    Dim coll1, row1, coll11, coll12, coll13 As Integer
    Dim old_p1, old_p2, old_p3, old_p4 As Integer
    Dim c As Range
    Dim over_col As Integer

    coll1 = need_search.Column
    row1 = need_search.Row
    money = need_search.Value

    coll11 = p_1.Column
    coll12 = p_2.Column
    coll13 = p_3.Column
    old_p1 = Cells(row1, coll11)
    old_p2 = Cells(row1, coll12)
    old_p3 = Cells(row1, coll13)
    old_p4 = Cells(row1, coll13)

    over_col = overit.Column

    With Range(Cells(1, over_col), Cells(row1 - 1, over_col))
        Set c = .Find(row1, LookIn
        = xlValues)
        If Not c Is Nothing Then
            firstaddress = c.Address
            Do
                If c.Value2 = row1 Then
                    searchit = c.Row
                    GoTo over
                End If
            Loop
        End With
    End Function

```

```

        End If
        Set c = .FindNext(c)
    Loop While Not c Is Nothing And c.Address <> firstaddress
End If
End With

With Range(Cells(row1, col1), Cells(10000, col1))
    Set c = .Find( - money, LookIn
    = xlValues)
    If Not c Is Nothing Then
        firstaddress = c.Address
        Do
            If old_p1 = c.Offset(0, (p_1.Column - col1)).Value And _
                old_p2 = c.Offset(0, (p_2.Column - col1)).Value And
-
                old_p3 = c.Offset(0, (p_3.Column - col1)).Value And
-
                old_p4 = c.Offset(0, (p_4.Column - col1)).Value And
-
                money = - (c.Value) Then
                searchhit = c.row
                GoTo over
            End If
            Set c = .FindNext(c)
        Loop While Not c Is Nothing And c.Address <> firstaddress
    End If
End With
searchhit = 0
over:

End Function

```

'这个函数是辅助的，作用是将 excel 表格列数换成字母 (b - >2)

```

Function transtoi(abc As String) As Integer
    Dim le As Integer
    Dim a, b As String
    Dim i, j As Integer
    If Len(abc) = 1 Then
        transtoi = Asc(LCase(abc)) - Asc("a") + 1
    Else
        a = Left(abc, 1)
        b = Right(abc, 1)
        i = Asc(LCase(a)) - Asc("a") + 1
        j = Asc(LCase(b)) - Asc("a") + 1
        transtoi = i * 26 + j
    End If
End Function

```

```
End If
End Function
```

'这个是调用的格式

```
Sub Test()
    Dim sea_i, i As Integer
    Dim coli, col1, col2, col3, col4 As Integer

    coli = transtoi(Application.InputBox("请输入列", "要查找列数", , , , , 2))
    col1 = transtoi(Application.InputBox("请输入行", "要查找匹配列数", , , , , 2))
    col2 = transtoi(Application.InputBox("请输入行", "要查找匹配列数", , , , , 2))
    col3 = transtoi(Application.InputBox("请输入行", "要查找匹配列数", , , , , 2))
    col4 = transtoi(Application.InputBox("请输入行", "要查找匹配列数", , , , , 2))
    col_end = transtoi(Application.InputBox("请结果行", "要结果的行数", , , , ,
2))
    i = 2
    Do
        sea_i = searchit(Cells(i, coli), Range(Cells(1, col_end), Cells(10000,
col_end)) _
            , Range(Cells(1, col1), Cells(10000, col1)) _
            , Range(Cells(1, col2), Cells(10000, col2)), Range(Cells(1, col3),
Cells(10000, col3)) _
            , Range(Cells(1, col4), Cells(10000, col4)))
        Cells(i, col_end) = sea_i
        i = i + 1
    Loop While Cells(i, coli) <> 0
    MsgBox (i)
End Sub
```

'#####

'44.函数作用:数值转换为字符地址

'#####

```
Public Function NtoC(Numbers As Integer) As String
    Dim S As String, E As String
    If Numbers <= 26 Then
        NtoC = Chr$(Numbers + 64)
    Else
        S = Chr$(Int((Numbers - 1) / 26) + 64)
        If Numbers Mod 26 = 0 Then
            E = "Z"
        Else

```

```

        E = Chr$(Numbers Mod 26 + 64)
    End If
    NtoC = S & E
End If
End Function

```

```

'#####

```

'45.函数作用:字符地址转换为数值

```

'#####

```

```

Public Function CtoN(Strings As String) As Integer
    Dim S1 As Long, S1 As String, S2 As String
    Strings = UCase(Strings)
    S1 = Len(Strings)
    If S1 = 0 Then
        CtoN = 0
    ElseIf S1 = 1 Then
        CtoN = Asc(Strings) - 64
    ElseIf S1 > 1 Then
        S1 = Mid(Strings, 1, 1)
        S2 = Mid(Strings, 2, 1)
        CtoN = (Asc(S1) - 64) * 26 + Asc(S2) - 64
    End If
End Function

```

```

'#####

```

'46.函数作用:等待时间（以秒计算）

```

'#####

```

```

Public Sub WaitTime(ByVal SpecSecond As Integer)
    Dim S1 As Date, S2 As Date, S As Long
    If SpecSecond <= 0 And SpecSecond > 60 Then Exit Sub
    S = 0
    S1 = Time()
    Do
        S2 = Time()
        If Second(S2) < Second(S1) Then
            '如果转到下一分钟,则以上一分钟的差加上下一分钟已过秒为间差
            S = (60 - Second(S1)) + Second(S2)
        End If
    Loop

```

```

Else
    '如果在相同分钟内，则直接相减即可
    S = Second(S2 - S1)
End If
DoEvents
Loop While S < SpecSecond
End Sub

```

'#####

'47.函数作用:得到字符串实际的长度（以单字节记）

'#####

```

Function LenTrue(SourceStr)
    Dim L, S, LenIs, GetStr
    S = 0
    L = 0
    Do
        S = S + 1
        '是双字节，跳到下一个字符
        GetStr = Mid(SourceStr, S, 1)
        If GetStr <> "" Then
            '是双字节
            If Asc(GetStr) < 0 Then
                LenIs = 2
            Else
                LenIs = 1
            End If
            L = L + LenIs
        End If
    Loop While GetStr <> ""
    LenTrue = L
End Function

```

'#####

'48.函数作用:18位身份证最后一位有效性验证

'#####

```

Function isTrue(bCode As String) As String
    Dim wi(1 To 17) As Integer

```

```

Dim ai(1 To 11) As String
wi(1) = 7
wi(2) = 9
wi(3) = 10
wi(4) = 5
wi(5) = 8
wi(6) = 4
wi(7) = 2
wi(8) = 1
wi(9) = 6
wi(10) = 3
wi(11) = 7
wi(12) = 9
wi(13) = 10
wi(14) = 5
wi(15) = 8
wi(16) = 4
wi(17) = 2
ai(1) = "1"
ai(2) = "0"
ai(3) = "X"
ai(4) = "9"
ai(5) = "8"
ai(6) = "7"
ai(7) = "6"
ai(8) = "5"
ai(9) = "4"
ai(10) = "3"
ai(11) = "2"
For i = 1 To 17
    b = Mid(bCode, i, 1)
    w = wi(i)
    sigma = sigma + (b * w)
Next
Number = Int(sigma Mod 11)
If LCase(Right(bCode, 1)) = LCase(ai(Number + 1)) Then
    isTrue = "合法"
Else
    isTrue = "不合法"
End If
End Function

'#####

```

'49.函数作用:计算符合maturity condition的拆解金额

```
'#####
```

```
Public Function LiabToHo(dReportDate As Date, dbUSD2CNY As Double, _ dbHKD2CNY As Double) As Double
```

```
    Dim EntryNum As Integer
    ' EntryNum is the number of contracts
    EntryNum = ThisWorkbook.Worksheets("DataPool").Range
_("Contract").Rows.Count
    LiabToHo = 0
    Dim dValuedate As Date
    Dim dMatuDate As Date
    Dim sCcy As String
    For i = 1 To EntryNum
        dValuedate =
ThisWorkbook.Worksheets("DataPool").Range("ValueDate").Cells(i).Value
        dMatuDate =
ThisWorkbook.Worksheets("DataPool").Range("MatuDate").Cells(i).Value
        sCcy = ThisWorkbook.Worksheets("DataPool").Range("Ccy").Cells(i)
        If dMatuDate - dValuedate <= 365 Then
            If dReportDate > dValuedate Then
                Select Case sCcy
                    Case Is = "USD"
                        LiabToHo = LiabToHo +
ThisWorkbook.Worksheets("DataPool").Range("Amt").Cells(i).Value * dbUSD2CNY
                    Case Is = "HKD"
                        LiabToHo = LiabToHo +
ThisWorkbook.Worksheets("DataPool").Range("Amt").Cells(i).Value * dbHKD2CNY
                End Select
            End If
        End If
    Next i
End Function
```

```
'#####
```

'50.函数作用:对多个用同一分隔符分隔的待查找元素，逐一在表区域首列内搜索，将返回选定单

元格的值相加，

' 相当于多个 vlookup 函数相加，对于查找不到的元素在批注中添加，以提醒用户。

```

'#####

Function vlookupmore(lookup_value, delimiter, data_type, table_array,
col_index_num)
    With Application.Caller
        If Not .Comment Is Nothing Then .Comment.Delete
        kmarr = Split(lookup_value, delimiter)
        For Each perkm In kmarr
            If data_type = 1 Then
                kmdata = "=vlookup(" & Chr(34) & perkm & Chr(34) & "," &
table_array.Address(1, 1, 1, 1) & "," & col_index_num & ",false)"
            Else
                kmdata = "=vlookup(" & perkm & "," & table_array.Address(1, 1, 1,
1) & "," & col_index_num & ",false)"
            End If
            If IsError(Evaluate(kmdata)) = False Then
                vlookupmore = vlookupmore + Evaluate(kmdata)
            Else
                If .Comment Is Nothing Then
                    .AddComment perkm & "NotFound"
                Else
                    .Comment.Text perkm & "NotFound" & Chr(10), 1, False
                End If
            End If
        Next
    End With
End Function

```

'#####

'51.函数作用:根据个人所得税（工资）反算工资数

'#####

```

Function gz(Deduction As Double, tax As Double)

```

'本函数为计算根据个人所得税 计算工资
'Deduction 为扣除标准，北京现为 1200
'Gz 为当月应发工资总额

```

Select Case tax
    Case Is < 0
        gz = 0
    Case Is <= 25
        gz = Round((tax / 0.05 + Deduction), 2)

```

```

Case 25 To 175
    gz = Round(((tax - 25) / 0.1 + Deduction + 500), 2)
Case 175 To 625
    gz = Round(((tax - 175) / 0.15 + Deduction + 2000), 2)
Case 625 To 3625
    gz = Round(((tax - 625) / 0.2 + Deduction + 5000), 2)
Case 3625 To 8625
    gz = Round(((tax - 3625) / 0.25 + Deduction + 20000), 2)
Case 8625 To 14625
    gz = Round(((tax - 8625) / 0.3 + Deduction + 40000), 2)
Case 14625 To 21625
    gz = Round(((tax - 14625) / 0.35 + Deduction + 60000), 2)
Case 21625 To 29625
    gz = Round(((tax - 21625) / 0.4 + Deduction + 80000), 2)
Case Is >= 29625
    gz = Round(((tax - 29625) / 0.45 + Deduction + 100000), 2)
End Select
End Function

```

'#####

'52.函数作用:判断表是否存在

'#####

```

Public Fnction IsSheetExist(wb As WorkBook, sht As String) As Boolean
On Error GoTo ErrISE
Dim s As String

s = wb.worksheets(sht).Name
IsSheetExist = True
ErrISE:
IsSheetExist = False
End Function

```

'#####

'53.函数作用:角度转弧

'#####

```

Public Const pi = 3.1415926535

```

```

Public Function hd(dfm As Single) As Double
    Dim d As Integer
    Dim f As Single
    Dim m As Single
    '分别取出输入度数的度、分、秒
    d = Fix(dfm)
    f = Fix((dfm - d) * 100)
    m = ((dfm - d) * 100 - f)
    If f >= 60 Or m >= 60 Then
        MsgBox ("度、分、秒输入有误，请重新输入！")
        Exit Function
    End If
    '将它转换成十进制的度
    dfm = d + f / 60 + m / 36
    '将它转换成弧度
    hd = dfm * pi / 180
    '将弧度保留 6 位小数
    hd = Format(hd, "#0.000000")
End Function

```

```

'#####

```

```

'54.函数作用:比较相同的字符串

```

```

'#####

```

```

Function FindExistCount(rngSource As Range, rngTarget As Range) As Long
    Dim lngCount As Long
    Dim rg As Range
    Dim rngFind As Range
    For Each rg In rngTarget
        Set rngFind = rngSource.Find(rg.Text)
        If Not rngFind Is Nothing Then
            rg.Interior.Color = vbYellow
            lngCount = lngCount + 1
        End If
    Next rg
    FindExistCount = lngCount
    Set rg = Nothing
    Set rngFind = Nothing
End Function

```

```

'#####

```

'55.函数作用:对选定的数组进行排序

'#####

```
Sub SORTX()  
    Dim XX() As Variant  
    Dim Address As Excel.Range  
    Dim Record As Long  
    Addre = ActiveWindow.RangeSelection.Address  
    With Range(Addre)  
        SRow = .Row '数组起始列  
        CRow = .Rows.Count '数组总列数  
        TRow = SRow + CRow - 1 '数组结束列  
        Scolumn = .Column '数组起始栏  
        CColumn = .Columns.Count '数组总栏数  
        TColumn = Scolumn + CColumn - 1 '数组结束栏  
    End With  
    Record = CRow * CColumn '数组记录数  
    ReDim XX(Record, 2) As Variant  
    For Cx = Scolumn To TColumn  
        For Rx = SRow To TRow  
            Data = Trim(Cells(Rx, Cx).Value)  
            Cells(Rx, Cx).Value = ""  
            If Data <> vbNullString Then  
                I = I + 1  
                XX(I, 1) = Left(Data, 1)  
                XX(I, 2) = Val(Right(Data, (Len(Data) - 1)))  
            End If  
        Next Rx  
    Next Cx  
    For Cx = 1 To Record - 1  
        For Rx = Cx + 1 To Record  
            If XX(Cx, 2) > XX(Rx, 2) Then  
                TOD = XX(Cx, 2)  
                XX(Cx, 2) = XX(Rx, 2)  
                XX(Rx, 2) = TOD  
            End If  
            TOD = XX(Cx, 1)  
            XX(Cx, 1) = XX(Rx, 1)  
            XX(Rx, 1) = TOD  
        Next Rx  
    Next Cx  
    I = 0  
    For Rx = SRow To TRow  
        For Cx = Scolumn To TColumn
```

Lin:

```
        I = I + 1
    If I > UBound(XX) Then Exit Sub
    Data = Trim(XX(I, 1) & XX(I, 2))
    If Data <> vbNullString Then
        Cells(Rx, Cx) = Data
    Else
        GoTo Lin
    End If
Next Cx
Next Rx
End Sub
```

'#####

'56.函数作用:取得指定月份天数

'#####

```
Public Function MDay(Optional XDate As Variant = 0) As Integer
    If IsDate(XDate) Then
        MDay = Day(DateSerial(Year(XDate), Month(XDate) + 1, 0))
    Else
        MDay = 0
    End If
End Function
```

'#####

'57.函数作用:排序工作表活页簿

'#####

```
Private Function Sort_Sheets()
    Dim sCount As Integer, I As Integer, R As Integer
    ReDim Na(0) As String
    sCount = Sheets.Count

    For I = 1 To sCount
        ReDim Preserve Na(I) As String
        Na(I) = Sheets(I).Name
    Next
```

```

For I = 1 To sCount - 1
    For R = I + 1 To sCount
        If Na(R) < Na(I) Then
            JH = Na(I)
            Na(I) = Na(R)
            Na(R) = JH
        End If
    Next
Next

For I = 1 To sCount
    Sheets(Na(I)).Move After
    = Sheets(I)
Next
End Function

'#####

'58.函数作用:统计数组中非重复数据个数

'#####

Public Function NumberCount() As Long
    Dim SeRange As Range
    Dim Nx As Range
    Dim No As Double

    Set SeRange = Range(Selection.Address)
    For Each Nx In SeRange
        No = WorksheetFunction.CountIf(SeRange, Nx)
        If No > 0 And No < 1 Then
            NumberCount = NumberCount + (1 / No)
        ElseIf No <> 0 Then
            NumberCount = NumberCount + 1
        End If
    Next
    Set SeRange = Nothing
End Function

'#####

```

'59.函数作用:摘取子字符串

' 说 明:第一参数: StrR 为引用单元格, 第二参数 StrH 为分割字符, 第三参数 I 为摘取第几个子字符串

'#####

```
Function Ssplit(StrR As Range, StrH As String, I As Integer) As String
    Ssplit = Split(Application.Trim(StrR), StrH, -1)(I - 1)
End Function
```

'#####

'60.函数作用:计算 20000 余个汉字的笔画

'#####

```
Function STROCK(CHNCHR As String) STR1 = "与之及矢扌 3, 允乏以及巨 4, 卂夂卂印回夂 5, 𠂇似吸𠂇 6, 𠂇巫𠂇𠂇 7, 𠂇𠂇𠂇𠂇𠂇𠂇 8, 𠂇𠂇𠂇𠂇𠂇𠂇 9, 𠂇𠂇𠂇𠂇𠂇𠂇 10, 彪彪将𠂇𠂇𠂇𠂇𠂇𠂇 11, "
    STR2 = "晚𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 12, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 13, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 14, "
    STR3 = "𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 15, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 16, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 17, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 18, "
    STR3 = "𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 19, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 20, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 21, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 22, "
    STR4 = "𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 23, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 24, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 25, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 26, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 27, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 28, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 29, 𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇𠂇 30, "
    STR1234 = STR1 + STR2 + STR3 + STR4
    On Error Resume Next
    N = WorksheetFunction.Find(CHNCHR, STR1234)
    If N > 0 Then
        CN = "0"
        For i = N To Len(STR1234)
            CHAR0 = Mid(STR1234, i, 1)
            If CHAR0 <> "," Then
                If Asc(CHAR0) <= 57 And Asc(CHAR0) > 47 Then
                    CN = CVar(CN) * 10 + CVar(CHAR0)
                End If
            Else
                STROCK = CInt(CN)
                Exit Function
            End If
        Next i
    End If
```


'62.函数作用:取得相近数据

'#####

```
Sub tet()  
    Dim temp As String  
    Dim MyArray(11)  
    For I = 0 To 11  
        MyArray(I) = I  
    Next  
    hh = "9"  
    temp = MyArray(0)  
    For I = 1 To 11  
        If Abs(hh - MyArray(I)) < Abs(hh - temp) Then temp = MyArray(I)  
    Next  
    MsgBox temp  
End Sub
```

'#####

'63.函数作用:提取定串中汉字

'#####

```
Public Function HZGet(ByVal strscr As String) As String  
    Dim i As Integer  
    For i = 1 To Len(strscr)  
        '汉字小于 ASC 值 0，否则在 0-127 之间  
        If Asc(Mid(strscr, i, 1)) < 0 Then  
            HZGet = HZGet & Mid(strscr, i, 1)  
        End If  
    Next i  
    HZGet = HZGet  
End Function
```

'#####

'64.函数作用:搜索重复数据(选定范围)

'#####

```
Public Function DataCheck()
```

```

Dim SelRange As Range
Dim Tx1 As Range
Set SelRange = Range(Selection.Address)

For Each Tx1 In SelRange
    If WorksheetFunction.CountIf(SelRange, Tx1) > 1 Then
        Tx1.Font.ColorIndex = 3
    End If
Next
Set SelRange = Nothing
End Function

'#####

'65.函数作用:字符型转数字型

'#####

Private Function TxtCData()
    Dim Sel As Range
    Dim TRow As Long, BRow As Long
    Dim LCou As Long, RCou As Long

    Set Sel = Range(Selection.Address)

    TRow = Sel.Row
    BRow = TRow + Sel.Rows.Count - 1

    LCou = Sel.Column
    RCou = LCou + Sel.Columns.Count - 1

    For C = LCou To RCou
        For R = TRow To BRow
            If Cells(R, C).NumberFormatLocal = "@" And IsNumeric(Cells(R, C).Value)
= True Then
                Cells(R, C).NumberFormatLocal = "G/通用格式"
                If Cells(R, C).Value <> vbNullString Then _
                    Cells(R, C).Value = Val(Cells(R, C).Value)
            End If
        Next
    Next
    Set Sel = Nothing
End Function

'#####

```

'66.函数作用:小写人民币转大写人民币

'#####

```
Function DXRMB(ByVal num As String) As String
    Dim NumV
    Dim HzStr As String, Nums As String

    NumV = Val(num) '
    If NumV < 0 Then ZfBz = "(负)" '正负数标志
    NumV = Abs(NumV) '转换为绝对值
    If NumV = 0 Then
        DXRMB = "零元"
        Exit Function
    End If
    If NumV >= 10000000000000# Then
        DXRMB = "#金额超出范围!"
        Exit Function
    End If

    DxSt = Split("零-壹-贰-叁-肆-伍-陆-柒-捌-玖", "-") '预设大写字符数组
    HzStr = "万仟佰拾亿仟佰拾万仟佰拾元角分" '预设人民币字符

    Nums = Trim(Str(NumV * 100)) '将数字乘 100 转换为整数
    If InStr(1, Nums, ".") > 0 Then Nums = Left(Nums, InStr(1, Nums, ".") - 1)
    NumCount = Len(Nums) '计算数字转换后的字符数
    HzStr = Right(HzStr, NumCount) '提取与数字字符数相同的人民币字符

    For i = 1 To NumCount
        StrID = Val(Mid(Nums, i, 1)) '从数字字符各数值计算出提取大写字符数组号
        RmbStr = RmbStr & DxSt(StrID) & Mid(HzStr, i, 1) '提取大写字符及人民币字符
    Next

    StrA = Split("零仟零佰零拾零万-零仟-零佰-零拾-零零-零零-零亿-零万-零零-零零-零元-零角零分-零角-零分", "-") '被替换的字符
    StrB = Split("零^零^零^零^零^零^亿零^万零^零^零^元^整^零^整", "^") '要替换的新字符

    For i = 0 To UBound(StrA)
        If InStr(1, RmbStr, StrA(i)) > 1 Then RmbStr = Replace(RmbStr, StrA(i), StrB(i)) '开始替换
    Next i

    DXRMB = ZfBz & RmbStr '取得函数值
```

End Function

'#####

'67.函数作用:取得指定月份人星期天个数

'#####

Public Function CWDay(XDate As Variant) As Integer

 If IsDate(XDate) Then

 Dim CDay As Integer, Cweek As Integer

 CDay = Day(DateSerial(Year(XDate), Month(XDate) + 1, 0))

 Cweek = Weekday(DateSerial(Year(XDate), Month(XDate), 1), 2)

 'Int((CDay - 1 + Cweek) / 7)

 CWDay = Int((CDay - 1 + Cweek) / 7)

 Else

 CWDay = 0

 End If

End Function

'#####

'68.函数作用:侦测档案是否包含宏

'#####

Sub CheckMacro()

 Dim vaItem

 Dim VBC As Object

 Dim HasCode As Boolean

 Dim wb As Workbook

 Application.EnableEvents = False

 Application.ScreenUpdating = False

 Set wb = Workbooks.Open("F:\REPORT\S.XLS", ReadOnly
 = True)

 HasCode = False

 If ActiveWorkbook.VBProject.Protection = 1 Then GoTo Eo

```

For Each VBC In wb.VBProject.VBComponents
    If VBC.Type <> 100 Then
        HasCode = True
        Exit For
    ElseIf VBC.CodeModule.CountOfDeclarationLines <
VBC.CodeModule.CountOfLines Then
        HasCode = True
        Exit For
    End If
Next
Eo:
If HasCode = True Then
    MsgBox "档案有宏"
Else
    MsgBox "档案无宏"
End If

wb.Close 0
Application.EnableEvents = True
Application.ScreenUpdating = True
End Sub

```

'#####

'69.函数作用:获取循环参照单元格

'#####

```

Sub CheckIntersect()
    Dim rng As Range
    Dim sht As Worksheet
    Dim fd As Range

    For Each sht In ThisWorkbook.Worksheets
        For Each rng In sht.Cells.SpecialCells(xlCellTypeFormulas) '包含工式的单
元格
            On Error Resume Next
            Set fd = rng.Precedents '前导参照
            If Not fd Is Nothing Then
                If Not Application.Intersect(fd, rng) Is Nothing Then '检查重迭范
围
                    MsgBox rng.Worksheet.Name & "!" & rng.Address
                End If
                Set fd = Nothing
            End If
        Next
    Next
End Sub

```

```
Next
Next
End Sub
```

```
'#####
```

'70.函数作用:创建桌面快捷方式

```
'#####
```

```
Sub CreatShortCut()
    Dim WSHShell
    Set WSHShell = CreateObject("WScript.Shell")
    Dim MyShortcut, MyDesktop, DesktopPath
    DesktopPath = WSHShell.SpecialFolders("Desktop")
    Set MyShortcut = WSHShell.CreateShortcut(DesktopPath & "\记事本快捷方式.lnk")
    MyShortcut.TargetPath =
    WSHShell.ExpandEnvironmentStrings("%windir%\notepad.exe")
    MyShortcut.WorkingDirectory = WSHShell.ExpandEnvironmentStrings("%windir%")
    MyShortcut.WindowStyle = 4
    MyShortcut.IconLocation =
    WSHShell.ExpandEnvironmentStrings("%windir%\notepad.exe, 0")
    MyShortcut.Save
End Sub
```

```
'#####
```

'71.函数作用:自动建立多级目录

```
'#####
```

```
Public Function M_Number(Field As String) As String
    Field_Len = Len(Field)
    Start = 4
    Number = InStr(Start, Field, "\")
    Do While Number > 0 Or Start < Field_Len
        If Number > 0 Then
            Text = Left(Field, Number - 1)
            Start = Number + 1
        Else
            Text = Field
            Start = Field_Len
        End If
```

```

        Number = InStr(Start, Field, "\")
        If Dir(Text, 30) = "" Then
            Mkdir Text
        Else
            If (GetAttr(Text) And vbDirectory) <> vbDirectory Then Mkdir Text
        End If
    Loop
End Function

```

```

'#####

```

'72.函数作用:统计经筛选后符合条件的记录条数

```

'#####

```

```

Public Function CuntRecord()As Long
    Dim uRange As Range
    Set uRange = ActiveSheet.UsedRange
    CuntRecord = (uRange.SpecialCells(xlCellTypeVisible).Count /
uRange.Columns.Count) - 1
    Set uRange = Nothing
End Function

```

```

'#####

```

'73.函数作用:复制单元格列高与栏宽

```

'#####

```

```

Sub CopyFormat()
    If Application.CutCopyMode = False Then
        ThisWorkbook.Keywords = "[" & ActiveWorkbook.Name & "]" & ActiveSheet.Name
& "!" & Selection.Address
        Range(Selection.Address).Copy
    Else
        Dim cRange As Range
        Dim Rng As Range

        Set cRange = Range(ThisWorkbook.Keywords)

        For Each Rng In cRange.EntireColumn
            ActiveCell.ColumnWidth = Rng.ColumnWidth
            ActiveCell.Offset(0, 1).Select
        Next Rng
    End If
End Sub

```

```

Next

For Each Rng In cRange.EntireRow
    ActiveCell.RowHeight = Rng.RowHeight
    ActiveCell.Offset(1, 0).Select
Next Rng
Cells(cRange.Row, cRange.Column).Select
ThisWorkbook.Keywords = vbNullString
Application.CutCopyMode = False
End If
End Sub

```

'#####

'74.函数作用:取消隐藏工作表(包括vba Project工程保护的)

'#####

```

Sub ShowSheet()
    Dim I As Worksheet
    For Each I In ActiveWorkbook.Sheets
        If I.Visible > -1 Then _
            I.Visible = -1
    Next
End Sub

```

'#####

'75.函数作用:删除单元格自定义名称

'#####

```

Sub DeleteName()
    For Each I In ActiveWorkbook.Names
        ActiveWorkbook.Names(I.Name).Delete
    Next
End Sub

```

'#####

'76.函数作用:从文件路径中取得文件名

'#####

```
Function FileName(FullName As Variant) As String
    Dim X%
    FileName$ = FullName
    X% = InStr(FullName, "\")
    Do While X%
        Ct% = X%
        X% = InStr(Ct% + 1, FullName, "\")
    Loop
    If Ct% > 0 Then FileName$ = Mid$(FullName, Ct% + 1)
End Function
```

'#####

'77.函数作用:取得一个文件的扩展名

'#####

```
Function Extension(FullName As Variant) As String
    Dim X%
    Extension$ = FullName
    X% = InStr(FullName, "\")
    Do While X%
        Ct% = X%
        X% = InStr(Ct% + 1, FullName, "\")
    Loop
    If Ct% > 0 Then Extension = Mid$(FullName, Ct% + 1)

    X% = InStr(Extension, ".")
    If X% > 0 Then
        Extension = Mid$(Extension, X% + 1)
    Else
        Extension = vbNullString
    End If
End Function
```

'#####

'78.函数作用:取得一个文件的路径

'#####

```
Function FilePath(FullName As Variant) As String
    Dim X%, Ct%
    FilePath$ = FullName
    X% = InStr(FullName, "\")
    Do While X%
        If X% > 0 Then FilePath$ = Left$(FullName, X%)
        X% = InStr(X% + 1, FullName, "\")
    Loop
End Function
```

'#####

'79.函数作用:十进制转二进制

'#####

```
Public Function dec2bin(mynum As Variant) As String
    Dim loopcounter As Integer
    If mynum >= 2 ^ 31 Then
        dec2bin = "Too big"
        Exit Function
    End If
    Do
        If (mynum And 2 ^ loopcounter) = 2 ^ loopcounter Then
            dec2bin = "1" & dec2bin
        Else
            dec2bin = "0" & dec2bin
        End If
        loopcounter = loopcounter + 1
    Loop Until 2 ^ loopcounter > mynum
End Function
```

'#####

'80.函数作用:检查一个数组是否为空

'#####

```

Public Function CheckArray(ArrayName As Variant, Optional Com As Integer = 0) As
Variant
    On Error GoTo Er
    Select Case Com
        Case 0
            Do
                Ne = Ne + 1
                XT = UBound(ArrayName, Ne)
            Loop
        Case Else
            CheckArray = UBound(ArrayName, Com)
    End Select
    Exit Function
Er:
    If Com = 0 Then CheckArray = Ne - 1 Else CheckArray = -1
End Function

```

'#####

'81.函数作用:字母栏名转数字栏名

'#####

```

Function ColumnN(abc As String) As Long
    abc = UCase(abc)
    Select Case Len(abc)
        Case 1
            ColumnN = Asc(abc) - 64
        Case 2
            ColumnN = (Asc(Left(abc, 1)) - 64) * 26 + Asc(Right(abc, 1)) - 64
    End Select
End Function

```

'#####

'82.函数作用:数字栏名转文字栏名

'#####

```

Function ColumnT(Colum As Integer) As String
    Select Case Colum
        Case 1 To 26
            ColumTex = Chr(64 + Colum)

```

```

        Case 27 To 256
            ColumTex = Chr(64 + (Colum \ 26)) & Chr(64 + (Colum Mod 26))
        End Select
    End Function

```

```

'#####

```

'83.函数作用:判断一件活页夹中是否还有子目录

```

'#####

```

```

Function CheckDirectory(sPath As String) As Boolean
    If Right(sPath, 1) <> "\" Then sPath = sPath & "\"
    Dim sDir As String
    sDir = Dir(sPath & ".*", vbDirectory)
    While sDir <> ""
        If GetAttr(sPath & sDir) And vbDirectory Then
            CheckDirectory = True
            sDir = ""
        Else
            sDir = Dir()
        End If
    Wend
End Function

```

```

'#####

```

'84.函数作用:判断一个文件是否在使用中

```

'#####

```

```

Function IsOpen(sFile As String) As Boolean
    Dim fFile As Integer
    fFile = FreeFile()
    On Error GoTo ErrOpen
    Open sFile For Binary Lock Read Write As fFile
    Close fFile
    Exit Function
ErrOpen:
    If Err.Number <> 70 Then
        Msg = "Error # " & Str(Err.Number) & " was generated by " & Err.Source &
Chr(13) & Err.Description
        MsgBox Msg, , "Error", Err.HelpFile, Err.HelpContext
    End If
End Function

```

```

Else
    IsOpen = True
End If
End Function

```

```

'#####

```

'85.函数作用:列出档案详细摘要信息

```

'#####

```

```

Sub GetDetails()
    Set objshell = CreateObject("Shell.Application") '引用 Shell.Application 物件
    '取得档案
    FileName = Application.GetOpenFilename(FileFilter
    = "档案(*.*),*.*", Title
    = "请选取档案")
    If FileName = False Then Exit Sub
    '取得路径
    rPath = WorksheetFunction.Substitute(FileName, Dir(FileName, vbDirectory),
    "")
    Set ofolder = objshell.Namespace(rPath) '引用指定数据夹
    For i = 0 To 35
        Cells(1, i + 1) = ofolder.GetDetailsOf(ofolder.Items, i)
    Next
    Set oFile = ofolder.Items.Item(Dir(FileName)) '引用指定档案
    '列出档案详细摘要信息
    For i = 0 To 35
        Cells(2, i + 1) = ofolder.GetDetailsOf(oFile, i)
    Next i
    Set ofolder = Nothing
    Set objshell = Nothing
End Sub

```

```

'#####

```

'86.函数作用:获取菜单ID编号及名称列表

```

'#####

```

```

Sub MenuList()
    On Error Resume Next
    Dim Nx As CommandBar

```

```

Dim I As Integer
For Each Nx In Application.CommandBars
    I = I + 1
    Range("A" & I).Value = Nx.Name
    Range("C" & I).Value = Nx.NameLocal
    For Each X In Application.CommandBars(Nx.Name).Controls
        I = I + 1
        Range("B" & I).Value = X.Id
        Range("C" & I).Value = X.Caption
        Range("D" & I).Value = X.FaceId
    Next
Next
End Sub

```

'#####

'87.函数作用:状态列动态显示文字

'#####

```

Public Function Message_List()
    Move_Tx = String(152, " ") & "Excel 精英俱乐部"
    If Len(Move_Tx) - Gx = 0 Then Gx = 0
    Move_Tx = Right(Move_Tx, Len(Move_Tx) - Gx)
    Gx = Gx + 2
    Application.StatusBar = Move_Tx
    Application.OnTime Now + TimeValue("00:00:01"), "Message_List"
End Function

```

'#####

'88.函数作用:取得一个文件的路径 2

'#####

```

Function getPath(fullName As String) As String
    Dim varVar As Variant
    varVar = Split(fullName, "\")
    varVar(UBound(varVar)) = ""
    getPath = Join(varVar, "\")
End Function

```

'#####

'89.函数作用:取得一个文件的路径 3

'#####

```
Function thePath(fullName As String) As String
    thePath = Replace(fullName, Dir(fullName), "")
End Function
```

'#####

'90.函数作用:取得Activecell的栏名

'#####

```
Function chrCol(myCell As Range) As String
    chrCol = Split(Split(myCell.Address, ":")(0), "$")(1)
End Function
```

'#####

'91.函数作用:取得单元格中指定字符前的字符

'#####

```
Public Function xLeft(Reg As Range, Space As String) As Variant
    Dim X As Integer
    X = InStr(Reg.Value, Space)
    If X <> 0 Then
        xLeft = Left(Reg.Value, X - 1)
    Else
        xLeft = Reg.Value
    End If
End Function
```

'#####

'92.函数作用:前单元格指定字符前的字符颜色改成红色

'#####

```

Public Function tColor(Reg As Range, Space As String) As Variant
    Dim X As Integer
    X = InStr(Reg.Value, Space)
    If X <> 0 Then
        Reg.Characters(start
            = 1, Length
            = X).Font.ColorIndex = 3
    Else
        xLeft = Reg.Value
    End If
End Function

```

'#####

'93.函数作用:根据数字返回对应的字母列号

'#####

'n 必须介于 1 到 256 之间

```

Function num2letter(n As Integer) As String
    If n >= 1 And n <= 256 Then
        num2letter = IIf(n < 26, Mid(Cells(1, n).Address, 2, 1), Mid(Cells(1,
n).Address, 2, 2))
    Else
        num2letter = ""
    End If
End Function

```

'#####

'94.函数作用:取工作表名字

'#####

```

Function SN(I As Integer) As String
    SN = Sheets(I).Name
End Function

```

'#####

'95.函数作用:取消所有隐藏的宏表

'#####

```
Sub ListMacroSheet()  
    For Each I In ThisWorkbook.Excel4MacroSheets  
        I.Visible = True  
    Next  
End Sub
```

'#####

'96.函数作用:导出VBA Project代码

'#####

```
Public Function ExportCode()  
    For Each theMod In ThisWorkbook.VBProject.VBComponents  
        theMod.Export "the" & theMod.Name & ".bas"  
    Next  
End Function
```

'#####

'97.函数作用:导入VBA Project代码

'#####

```
Function ImportCode1()  
    'Dim theMod As VBIDE.VBComponent  
    For Each theMod In ThisWorkbook.VBProject.VBComponents  
        With theMod.CodeModule  
            '                .AddFromFile "c:\windows\desktop\index_Y.txt"  
            .AddFromFile "the" & .Parent.Name & ".bas"  
        End With  
    Next  
End Function
```

'#####

'98.函数作用:取得汉字拼音的第一个字母

'#####

```
Private Function GetPYChar(a1 As String) As String
    Dim t1 As String
    If Asc(a1) < 0 Then
        t1 = Left(a1, 1)
        If Asc(t1) < Asc("啊") Then
            GetPYChar = " "
            Exit Function
        End If
        If Asc(t1) >= Asc("啊") And Asc(t1) < Asc("芭") Then
            GetPYChar = "A"
            Exit Function
        End If
        If Asc(t1) >= Asc("芭") And Asc(t1) < Asc("擦") Then
            GetPYChar = "B"
            Exit Function
        End If
        If Asc(t1) >= Asc("擦") And Asc(t1) < Asc("搭") Then
            GetPYChar = "C"
            Exit Function
        End If
        If Asc(t1) >= Asc("搭") And Asc(t1) < Asc("蛾") Then
            GetPYChar = "D"
            Exit Function
        End If
        If Asc(t1) >= Asc("蛾") And Asc(t1) < Asc("发") Then
            GetPYChar = "E"
            Exit Function
        End If
        If Asc(t1) >= Asc("发") And Asc(t1) < Asc("噶") Then
            GetPYChar = "F"
            Exit Function
        End If
        If Asc(t1) >= Asc("噶") And Asc(t1) < Asc("哈") Then
            GetPYChar = "G"
            Exit Function
        End If
        If Asc(t1) >= Asc("哈") And Asc(t1) < Asc("击") Then
            GetPYChar = "H"
            Exit Function
        End If
        If Asc(t1) >= Asc("击") And Asc(t1) < Asc("喀") Then
```

```

        GetPYChar = "J"
        Exit Function
    End If
    If Asc(t1) >= Asc("喀") And Asc(t1) < Asc("垃") Then
        GetPYChar = "K"
        Exit Function
    End If
    If Asc(t1) >= Asc("垃") And Asc(t1) < Asc("妈") Then
        GetPYChar = "L"
        Exit Function
    End If
    If Asc(t1) >= Asc("妈") And Asc(t1) < Asc("拿") Then
        GetPYChar = "M"
        Exit Function
    End If
    If Asc(t1) >= Asc("拿") And Asc(t1) < Asc("哦") Then
        GetPYChar = "N"
        Exit Function
    End If
    If Asc(t1) >= Asc("哦") And Asc(t1) < Asc("啪") Then
        GetPYChar = "O"
        Exit Function
    End If
    If Asc(t1) >= Asc("啪") And Asc(t1) < Asc("期") Then
        GetPYChar = "P"
        Exit Function
    End If
    If Asc(t1) >= Asc("期") And Asc(t1) < Asc("然") Then
        GetPYChar = "Q"
        Exit Function
    End If
    If Asc(t1) >= Asc("然") And Asc(t1) < Asc("撒") Then
        GetPYChar = "R"
        Exit Function
    End If
    If Asc(t1) >= Asc("撒") And Asc(t1) < Asc("塌") Then
        GetPYChar = "S"
        Exit Function
    End If
    If Asc(t1) >= Asc("塌") And Asc(t1) < Asc("挖") Then
        GetPYChar = "T"
        Exit Function
    End If
    If Asc(t1) >= Asc("挖") And Asc(t1) < Asc("昔") Then
        GetPYChar = "W"
        Exit Function

```

```

End If
If Asc(t1) >= Asc("昔") And Asc(t1) < Asc("压") Then
    GetPYChar = "X"
    Exit Function
End If
If Asc(t1) >= Asc("压") And Asc(t1) < Asc("匣") Then
    GetPYChar = "Y"
    Exit Function
End If
If Asc(t1) >= Asc("匣") Then
    GetPYChar = "Z"
    Exit Function
End If
Else
    If UCase(a1) <= "Z" And UCase(a1) >= "A" Then
        GetPYChar = UCase(Left(a1, 1))
    Else
        GetPYChar = " "
    End If
End If
End Function

```

```

Private Function GetPYStr(ByVal S As String) As String
    Dim l As Long
    Dim sOut As String

    If S <> "" Then
        For l = 1 To Len(S)
            sOut = sOut & GetPYChar(Mid(S, l, 1))
        Next l
        GetPYStr = sOut
    End If
End Function

```

```
'#####
```

'99.函数作用:获取两栏中相同的数据

```
'#####
```

```

Function Wsame(x As Variant, y As Variant, z As Integer)
    Dim I As Long
On Error GoTo Er:
    Application.ScreenUpdating = False
    For Each Mr1 In x

```

```

        D = WorksheetFunction.Match(Mr1, y, 0)
        If D > 0 Then
            I = I + 1
            If I = z Then Wsame = Mr1
        End If
    Next
    Application.Volatile
    Exit Function
Er:
    D = 0
    Resume Next
End Function

'#####

'100.函数作用:选取当前工作表中公式出错的单元格,关返回出错个数

'#####

Public Function FormulaErrors() As Long
    If MsgBox("Do you want select cells with an error in their formula ?", _
        vbQuestion + vbOKCancel, AT) = vbCancel Then Exit Function
On Error GoTo Er:
    Cells.SpecialCells(xlCellTypeFormulas, 16).Select
    MsgBox "Cells with an error are selected.", vbInformation, "Formula"
    FormulaErrors = Selection.Count
    Exit Function
Er:
    MsgBox "There are no cells with an error on this sheet.", vbInformation,
    "Formula"
    FormulaErrors = 0
End Function

'#####

'101.函数作用:将工作表中最后一列作为页脚打印在每一面页尾

'#####

Public Sub Prin()
    '获取总页数
    If ExecuteExcel4Macro("Get.Document(50)") > 1 Then
        '获取每页行数

```

```

I = Application.ExecuteExcel4Macro("INDEX(GET.DOCUMENT(64),1)") - 2
X = I + 1
L = Range("A65536").End(xlUp).Row '总行数

For T = 2 To Application.WorksheetFunction.RoundUp(L / (I + 1), 0)
    Rows(L).Copy
    Rows(X).Insert Shift
    = xlDown
    Application.CutCopyMode = False
    X = X + I
    L = L + 1
Next T
Else
    ActiveSheet.PrintOut
Exit Sub
End If
ActiveSheet.PrintOut
For D = T - 1 To 2 Step -1
    X = X - I
    Rows(X).Delete Shift
    = xlUp
Next D
End Sub

```

```

'#####

```

'102.函数作用:获取vbproject引用项目

```

'#####

```

```

Sub ListReferences()
    For Each Ref In ThisWorkbook.VBProject.References
        i = i + 1
        Cells(i, 1) = Ref.Name
        Cells(i, 2) = Ref.GUID
        Cells(i, 3) = Ref.Major
        Cells(i, 4) = Ref.Minor
        Cells(i, 5) = Ref.FullPath
        Cells(i, 6) = Ref.Description
    Next Ref
End Sub

```

```

'#####

```

'103.函数作用:移除Excel工作表中的外部数据连接

```
'#####
```

```
Sub RemoveExternalLinks()  
    Dim intnroflinks As Integer  
    arlink = ActiveWorkbook.LinkSources()  
  
    On Error GoTo Continue  
    If arlink = 0 Then 'Empty  
        MsgBox "在这个工作表中未发现连接...", vbInformation, "提示"  
        Exit Sub  
    End If  
    On Error GoTo 0  
Continue:  
    intnroflinks = UBound(arlink)  
    If MsgBox("目前工作表连接到 " & intnroflinks & " 个文件,是否要把连接转成数据?",  
32 + vbYesNo, "提示") = vbYes Then  
        'varResponse = MsgBox("Do you want to delete this link: " &  
arLink(intCounter), vbYesNoCancel + vbQuestion + vbDefaultButton2, "")  
        Call LinksToData  
    End If  
End Sub
```

```
Function LinksToData()  
    On Error Resume Next  
    Dim rng As Range  
    Dim sht As Worksheet  
    For Each sht In ActiveWorkbook.Worksheets  
        For Each rng In sht.Cells.SpecialCells(xlCellTypeFormulas)  
            If InStr(rng.Formula, "[") <> 0 Then rng = rng.Value  
        Next  
    Next  
End Function
```

```
'#####
```

'104.函数作用:将选定单元格作成镜像图片

```
'#####
```

```
Sub Test()  
    'ExportRangeAsImage "d:\a.gif", "GIF"
```

```

        'ExportRangeAsImage "d:\a.JPG", "JPG"
End Sub

Sub ExportRangeAsImage(varFileName As Variant, ImageFilter As String)
    Dim objChart As ChartObject
    Dim chtChart As Chart
    Dim picPicture As Picture
    Dim sglWidth As Single
    Dim sglHeight As Single
    Dim rngSelection As Range
    Dim blnRet As Boolean
    On Error GoTo ExportRangeError

    Set rngSelection = Selection
    With Application
        .StatusBar = "Exporting range..."
        .ScreenUpdating = False
    End With
    rngSelection.CopyPicture Appearance
    = xlScreen, Format
    = xlPicture
    Set objChart = ActiveSheet.ChartObjects.Add(0, 0, 5000, 5000)
    Set chtChart = objChart.Chart

    objChart.Activate
    With chtChart
        .ChartArea.Select
        .Paste
        Set picPicture = .Pictures(1)
    End With

    With picPicture
        sglWidth = .Width + 7
        sglHeight = .Height + 7
        .Left = 0
        .Top = 0
    End With

    With objChart
        .Border.LineStyle = xlNone
        .Width = sglWidth
        .Height = sglHeight
    End With

    blnRet = chtChart.Export(FileName
    = varFileName, Filtername

```

```

    = ImageFilter, Interactive
    = False)
    objChart.Delete
    Set objChart = Nothing
    Application.StatusBar = False
    If Not blnRet Then
        MsgBox "Sorry, the export failed: please verify that you " & vbCrLf & _
            "have the appropriate filter installed on your PC.", vbExclamation, AT
    & " - Export range as image"
    Else
    End If
Continue:
    With Application
        .StatusBar = False
        .ScreenUpdating = True
    End With
    Exit Sub
ExportRangeError:
    MsgBox "Sorry, the export failed: please verify that you " & vbCrLf & _
        "have the appropriate filter installed on your PC." & vbCrLf & _
        "Error nr. " & Err.Number & ": " & Err.Description, vbExclamation, AT & "
- Export range as image"
    If Not objChart Is Nothing Then objChart.Delete
    Resume Continue
End Sub

```

'#####

'105.函数作用:反选择单元格中的数

' 示 例:A1 = 1,B1 = 2,C1 = 3;执行结果:C1 = 1,B1 = 2,A1 = 3

'#####

```

Function ReverseSelection()
    Application.ScreenUpdating = False
    Application.StatusBar = True
    Application.EnableEvents = False

    Set rngCel = Selection
    Rw = Selection.Rows.Count
    Cl = Selection.Columns.Count
    If Rw > 1 And Cl > 1 Then
        MsgBox "你选择的范围只能是一栏或一列...", 32, "提示"
        GoTo EndMacro
    End If
End Function

```

```

End If

If rngCel.Cells.Count = ActiveCell.EntireColumn.Cells.Count Then
    MsgBox "你选择的范围不能是一个整栏...", 32, "提示"
    GoTo EndMacro
End If

If Rw > 1 Then
    ReDim Arr(Rw)
Else
    ReDim Arr(C1)
End If

Rw = 0
For Each c In rngCel
    Arr(Rw) = c.Formula
    Rw = Rw + 1
Next c
Rw = Rw - 1

For Each c In rngCel
    c.Formula = Arr(Rw)
    Rw = Rw - 1
Next c

EndMacro:
    Application.ScreenUpdating = True
    Application.StatusBar = False
    Application.EnableEvents = True
End Function

'#####

'106.函数作用:在Excel中加入一个量度尺(以厘米为单位)

'#####

Sub MakeRuler_cm()'以厘米为单位
    'Define the size of a new ruler.
    Const Ruler_Width As Double = 10 'Width 16 cm
    Const Ruler_Height As Double = 10 'Height 14 cm

    'The setting size on the screen and the actual size on the printer.
    Const Screen_Width As Double = 16
    Const Screen_Height As Double = 14

```

```

Const Printer_Width As Double = 16
Const Printer_Height As Double = 14

Dim i As Long
Dim l As Long
Dim x As Long
Dim y As Long
Dim ws As Worksheet
Dim x2 As Double
Dim y2 As Double

x = Ruler_Width * 10
y = Ruler_Height * 10

Application.ScreenUpdating = False

Set ws = ActiveSheet
Worksheets.Add
ActiveSheet.Move
ActiveSheet.Lines.Add 0, 0, 3 * x, 0
For i = 1 To x
    If i Mod 10 = 0 Then
        l = 5 Else
        If i Mod 5 = 0 Then l = 4 Else l = 3
    End If
    ActiveSheet.Lines.Add 3 * i, 0, 3 * i, 3 * l
Next
ActiveSheet.Lines.Add 0, 0, 0, 3 * y
For i = 1 To y
    If i Mod 10 = 0 Then
        l = 5 Else
        If i Mod 5 = 0 Then l = 4 Else l = 3
    End If
    ActiveSheet.Lines.Add 0, 3 * i, 3 * l, 3 * i
Next
ActiveSheet.Lines.Border.ColorIndex = 55

For i = 10 To x - 1 Step 10
    With ActiveSheet.TextBoxes.Add(3 * i - 9, 3 * 5, 18, 12)
        .Text = Format(i \ 10, "!@@")
    End With
Next
For i = 10 To y - 1 Step 10
    With ActiveSheet.TextBoxes.Add(3 * 5, 3 * i - 9, 12, 18)
        .Orientation = xlDownward
        .Text = Format(i \ 10, "!@@")
    End With
Next

```

```

        End With
    Next
With ActiveSheet.TextBoxes
    .Font.Size = 9
    .Font.ColorIndex = 55
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter
    .Border.ColorIndex = xlNone
    .Interior.ColorIndex = xlNone
End With

With ActiveSheet.DrawingObjects.Group
    .Placement = xlFreeFloating
    .Width = Application.CentimetersToPoints(x / 10)
    .Height = Application.CentimetersToPoints(y / 10)
    .CopyPicture xlScreen, xlPicture
    ActiveSheet.Paste
    x2 = (Selection.Width - .Width) / 3
    y2 = (Selection.Height - .Height) / 3
    Selection.Delete
    .CopyPicture xlPrinter, xlPicture
    ActiveSheet.Paste
    .Width = .Width * .Width / (Selection.Width - x2 * 2) * Screen_Width /
Printer_Width
    .Height = .Height * .Height / (Selection.Height - y2 * 2) * Screen_Height
/ Printer_Height
    Selection.Delete
    If Val(Application.Version) >= 9 Then
        .Copy
        ActiveSheet.PasteSpecial 'Format:="Picture (PNG)"
        With Selection.ShapeRange.PictureFormat
            .CropLeft = x2
            .CropTop = y2
            .CropRight = x2
            .CropBottom = y2
        End With
        Selection.Copy
        ws.Activate
        ws.PasteSpecial 'Format:="Picture (PNG)"
        Selection.Placement = xlFreeFloating
        .Parent.Parent.Close False
    End If
End With
Application.ScreenUpdating = True
End Sub

```

```
'#####
```

'107.函数作用:在Excel中加入一个量度尺(以寸为单位)

```
'#####
```

```
Sub MakeRuler_inch() '以寸为单位
    'Define the size of a new ruler.
    Const Ruler_Width As Double = 6 'Width 6 inch
    Const Ruler_Height As Double = 5 'Height 5 inch

    'The setting size on the screen and the actual size on the printer.
    Const Screen_Width As Double = 6
    Const Screen_Height As Double = 5
    Const Printer_Width As Double = 6
    Const Printer_Height As Double = 5

    Dim i As Long
    Dim l As Double
    Dim x As Long
    Dim y As Long
    Dim ws As Worksheet
    Dim a(0 To 15) As Double
    Dim x2 As Double
    Dim y2 As Double

    x = Ruler_Width * 16
    y = Ruler_Height * 16
    a(0) = 3.6
    a(1) = 1
    a(2) = 2
    a(3) = 1
    a(4) = 2
    a(5) = 1
    a(6) = 2
    a(7) = 1
    a(8) = 3
    a(9) = 1
    a(10) = 2
    a(11) = 1
    a(12) = 2
    a(13) = 1
    a(14) = 2
    a(15) = 1
    Application.ScreenUpdating = False
```

```

Set ws = ActiveSheet
Worksheets.Add
ActiveSheet.Move
ActiveSheet.Lines.Add 0, 0, 3 * x, 0
For i = 1 To x
    l = a(i Mod 16)
    ActiveSheet.Lines.Add 3 * i, 0, 3 * i, 3 * l
Next
ActiveSheet.Lines.Add 0, 0, 0, 3 * y
For i = 1 To y
    l = a(i Mod 16)
    ActiveSheet.Lines.Add 0, 3 * i, 3 * l, 3 * i
Next
ActiveSheet.Lines.Border.ColorIndex = 55

For i = 16 To x - 1 Step 16
    With ActiveSheet.TextBoxes.Add(3 * i - 9, 3 * 3.6, 18, 12)
        .Text = Format(i \ 16, "!@@")
    End With
Next
For i = 16 To y - 1 Step 16
    With ActiveSheet.TextBoxes.Add(3 * 3.6, 3 * i - 9, 12, 18)
        .Orientation = xlDownward
        .Text = Format(i \ 16, "!@@")
    End With
Next
With ActiveSheet.TextBoxes
    .Font.Size = 9
    .Font.ColorIndex = 55
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter
    .Border.ColorIndex = xlNone
    .Interior.ColorIndex = xlNone
End With

With ActiveSheet.DrawingObjects.Group
    .Placement = xlFreeFloating
    .Width = Application.InchesToPoints(x / 16)
    .Height = Application.InchesToPoints(y / 16)
    .CopyPicture xlScreen, xlPicture
ActiveSheet.Paste
x2 = (Selection.Width - .Width) / 3
y2 = (Selection.Height - .Height) / 3
Selection.Delete
.CopyPicture xlPrinter, xlPicture

```

```

ActiveSheet.Paste
.Width = .Width * .Width / (Selection.Width - x2 * 2) * Screen_Width /
Printer_Width
.Height = .Height * .Height / (Selection.Height - y2 * 2) * Screen_Height
/ Printer_Height
Selection.Delete
If Val(Application.Version) >= 9 Then
    .Copy
    ActiveSheet.PasteSpecial 'Format:="Picture (PNG)"
    With Selection.ShapeRange.PictureFormat
        .CropLeft = x2
        .CropTop = y2
        .CropRight = x2
        .CropBottom = y2
    End With
    Selection.Copy
    ws.Activate
    ws.PasteSpecial 'Format:="Picture (PNG)"
    Selection.Placement = xlFreeFloating
    .Parent.Parent.Close False
End If
End With
Application.ScreenUpdating = True
End Sub

```

'#####

'108.函数作用:取得一个短文件名的长文件名

'#####

```

Public Function GetLongFilename (ByVal sShortName As String) As String
    Dim sLongName As String
    Dim sTemp As String
    Dim iSlashPos As Integer
    sShortName = sShortName & "\"
    iSlashPos = InStr(4, sShortName, "\")
    While iSlashPos
        sTemp = Dir(Left$(sShortName, iSlashPos - 1), vbNormal + vbHidden + vbSystem
+ vbDirectory)
        If sTemp = "" Then
            GetLongFilename = ""
            Exit Function
        End If
        sLongName = sLongName & "\" & sTemp
    End While
    GetLongFilename = sLongName & sShortName
End Function

```

```

        iSlashPos = InStr(iSlashPos + 1, sShortName, "\")
    Wend
    GetLongFilename = Left$(sShortName, 2) & sLongName
End Function

```

```

'#####

```

'109.函数作用:取得临时文件名

```

'#####

```

```

Public Const MAX_PATH = 260
Public Declare Function GetTempFileName Lib "kernel32" Alias "GetTempFileNameA"
    (ByVal lpzPath As String, ByVal lpPrefixString As String, ByVal wUnique As Long,
    ByVal lpTempFileName As String) As Long

```

```

Public Declare Function GetTempPath Lib "kernel32" Alias "GetTempPathA" (ByVal
nBufferLength As Long, ByVal lpBuffer As String) As Long

```

```

Public Function GetTempFile() As String
    Dim lngRet As Long
    Dim strBuffer As String, strTempPath As String
    strBuffer = String$(MAX_PATH, 0)
    lngRet = GetTempPath(Len(strBuffer), strBuffer)
    If lngRet = 0 Then Exit Function
    strTempPath = Left$(strBuffer, lngRet)
    strBuffer = String$(MAX_PATH, 0)
    lngRet = GetTempFileName(strTempPath, "tmp", 0&, strBuffer)
    If lngRet = 0 Then Exit Function
    lngRet = InStr(1, strBuffer, Chr(0))
    If lngRet > 0 Then
        GetTempFile = Left$(strBuffer, lngRet - 1)
    Else
        GetTempFile = strBuffer
    End If
End Function

```

```

'#####

```

'110.函数作用:等用Shell调用的程序执行完成后再执行其它程序

```

'#####

```

```

Public Declare Function WaitForSingleObject Lib "kernel32" (ByVal hObject As Long,
ByVal dwMilliseconds As Long) As Long
Public Declare Function CloseHandle Lib "kernel32" (ByVal hObject As Long) As Long
Public Declare Function OpenProcess Lib "kernel32" (ByVal dwDesiredAccess As Long,
ByVal bInheritHandle As Long, ByVal dwProcessId As Long) As Long
Public Const INFINITE = -1&
Public Const SYNCHRONIZE = &H100000

```

```

Private Sub Command1_Click()
    Dim i As Long
    Dim r As Long
    Dim p As Long
    i = Shell("NOTEPAD.EXE", vbNormalFocus)
    p = OpenProcess(SYNCHRONIZE, False, i)
    r = WaitForSingleObject(p, INFINITE)
    r = CloseHandle(p)
    MsgBox "Program Close"
End Sub

```

```

'#####

```

'111.函数作用:将Mouse显示成动画

```

'#####

```

```

Option Explicit
Const OCR_NORMAL = 32512
Const IDC_ARROW = 32512&
Private Declare Function LoadCursorFromFile Lib "user32" Alias
"LoadCursorFromFileA" (ByVal lpFileName As String) As Long
Private Declare Function LoadCursor Lib "user32" Alias "LoadCursorA" (ByVal
hInstance As Long, lpCursorName As Any) As Long ' modified
Private Declare Function SetSystemCursor Lib "user32" (ByVal hcur As Long, ByVal
id As Long) As Long

```

```

Sub ttt()
    Dim hCursor As Long

    hCursor = LoadCursorFromFile '(欲显示的.ani 或.cur 文件名称)
    Call SetSystemCursor(hCursor, OCR_NORMAL)
End Sub

```

```

Sub rest()
    '还原 Mouse 状态
    hCursor = LoadCursor(0&, ByVal IDC_ARROW)

```

```
Call SetSystemCursor(hCursor, OCR_NORMAL)
End Sub
```

```
'#####
```

```
'112.函数作用:限制Mouse移动范围
```

```
'#####
```

```
Public Declare Function ClipCursor Lib "User32" (lpRect As Any) As Long
```

```
Public Type RECT
    Left As Long
    Top As Long
    Right As Long
    Bottom As Long
End Type
```

```
Private Sub IE_Timer1_Timer()
    Dim z As RECT
    z.Bottom = 560 '下边界
    z.Top = 0 '上边界
    z.Left = 220 '左边界
    z.Right = 800 '右边界
    ClipCursor z
End Sub
```

```
'#####
```

```
'113.函数作用:取得当前激活窗品句柄及标题
```

```
'#####
```

```
Declare Function GetActiveWindow Lib "user32" () As Long
Declare Function GetWindowText Lib "user32" Alias "GetWindowTextA" (ByVal Hwnd As Long, ByVal lpString As String, ByVal cch As Long) As Long
```

```
Function Ac_Caption() As String
    Dim ACaption As String
    Dim Leng As Long
    ACaption = String$(255, vbNullChar)
    Leng = Len(ACaption)
```

```

        If GetWindowText(GetActiveWindow, ACaption, Leng) > 0 Then Ac_Caption =
ACaption
End Function

```

```

'#####

```

```

'114.函数作用:取得屏幕分辨率

```

```

'#####

```

```

Declare Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long
Const SM_CXSCREEN As Long = 0
Const SM_CYSCREEN As Long = 1

```

```

Public Function Display()
    Y = GetSystemMetrics(SM_CYSCREEN)
    X = GetSystemMetrics(SM_CXSCREEN)
End Function

```

```

'#####

```

```

'115.函数作用:自动建立多级目录

```

```

'#####

```

```

Public Function M_Number(Field As String) As String
    Field_Len = Len(Field)
    Start = 4
    Number = InStr(Start, Field, "\")
    Do While Number > 0 Or Start < Field_Len
        If Number > 0 Then
            Text = Left(Field, Number - 1)
            Start = Number + 1
        Else
            Text = Field
            Start = Field_Len
        End If
        Number = InStr(Start, Field, "\")
        If Dir(Text, 30) = "" Then
            Mkdir Text
        Else
            If (GetAttr(Text) And vbDirectory) <> vbDirectory Then Mkdir Text
        End If
    End While
End Function

```

```
Loop
End Function
```

```
'#####
```

'116.函数作用:将文件长度置零

```
'#####
```

```
Declare Function lcreat Lib "kernel32" Alias "_lcreat" (ByVal lpPathName As String,
ByVal iAttribute As Long) As Long
Declare Function lclose Lib "kernel32" Alias "_lclose" (ByVal hFile As Long) As
Long
```

```
Public Function auto_open()
    Dim ID As Long, FileName As String

    If MsgBox("是否要置为 0 长度?", 32 + vbYesNo, "提示") = vbNo Then Exit Function
    FileName = ThisWorkbook.FullName
    If GetAttr(FileName) And vbReadOnly <> 0 Then SetAttr FileName, vbNormal
    ID = lcreat(ThisWorkbook.FullName, 1)
    lclose ID
End Function
```

```
'#####
```

'117.函数作用:读取WIN9X / Me共享文件夹密码

```
'#####
```

```
Public Declare Function RegEnumKeyEx Lib "advapi32.dll" Alias "RegEnumKeyExA"
(ByVal hKey As Long, ByVal dwIndex As Long, ByVal lpName As String, lpcbName As
Long, ByVal lpReserved As Long, ByVal lpClass As String, lpcbClass As Long,
lpftLastWriteTime As FILETIME) As Long
Public Declare Function RegOpenKeyEx Lib "advapi32.dll" Alias "RegOpenKeyExA"
(ByVal hKey As Long, ByVal lpSubKey As String, ByVal ulOptions As Long, ByVal
samDesired As Long, phkResult As Long) As Long
Public Declare Function RegQueryValueEx Lib "advapi32.dll" Alias
"RegQueryValueExA" (ByVal hKey As Long, ByVal lpValueName As String, ByVal
lpReserved As Long, lpType As Long, lpData As Any, lpcbData As Long) As Long
Public Declare Function RegSetValueEx Lib "advapi32.dll" Alias "RegSetValueExA"
(ByVal hKey As Long, ByVal lpValueName As String, ByVal Reserved As Long, ByVal
dwType As Long, lpData As Any, ByVal cbData As Long) As Long
```

```

Public Const HKEY_LOCAL_MACHINE = &H80000002
Public Const READ_CONTROL = &H20000
Public Const STANDARD_RIGHTS_READ = (READ_CONTROL)
Public Const KEY_QUERY_VALUE = &H1
Public Const KEY_ENUMERATE_SUB_KEYS = &H8
Public Const KEY_NOTIFY = &H10
Public Const SYNCHRONIZE = &H100000
Public Const KEY_READ = ((STANDARD_RIGHTS_READ Or KEY_QUERY_VALUE Or
KEY_ENUMERATE_SUB_KEYS Or KEY_NOTIFY) And (Not SYNCHRONIZE))
Public Const ERROR_SUCCESS = 0&
Public Const ERROR_NO_MORE_ITEMS = 259&

```

Type DateTime

```

    dwLowDateTime As Long
    dwHighDateTime As Long

```

End Type

'建立共享密码打印文件

```

Public Function ShareFolderPasswordList(CreateFileName As String)
    Dim ret As Long
    Dim hKey As Long
    sKey = "Software\Microsoft\Windows\CurrentVersion\Network\LanMan"
    ret = RegOpenKeyEx(HKEY_LOCAL_MACHINE, sKey, 0&, KEY_READ, hKey)
    If ret <> ERROR_SUCCESS Then
        MsgBox "Get Register Date Find Error", vbOKOnly + vbCritical, "Error"
        Exit Function
    End If
    Dim lngIndex As Long
    Dim sName As String
    Dim lngcbName As Long
    Dim ftLastWriteTime As DateTime
    Dim FileID As Integer
    lngIndex = 0
    FileID = FreeFile
    Open CreateFileName For Output As FileID
    ToNextSubKey:
        sName = String(13, 0)
        lngcbName = 13
        ret = RegEnumKeyEx(hKey, lngIndex, sName, lngcbName, 0&, vbNullString, 0&,
ftLastWriteTime)
        If ret = ERROR_NO_MORE_ITEMS Then GoTo ToContinue
        If ret <> ERROR_SUCCESS Then
            MsgBox "Get Register Date Find Error", vbOKOnly + vbCritical, "Error"
            Exit Function
        End If

```

```

Dim s As String
s = Left(sName, LngcbName)
'>>>Read Only Password
Dim sPassword1 As String
sPassword1 = ""
sPassword1 = GetSharePassword(s, "Parm2enc")
'>>>All Password
Dim sPassword2 As String
sPassword2 = ""
sPassword2 = GetSharePassword(s, "Parm1enc")
Write #FileID, s, sPassword1, sPassword2
LngIndex = LngIndex + 1
GoTo ToNextSubKey
ToContinue:
    Close FileID
End Function

'读取指定共享源文件密码

Public Function GetSharePassword(ByVal sName As String, ByVal sValueName As String)
As String
    Dim ret As Long
    Dim hKey As Long
    Dim sKey As String
    sKey = "Software\Microsoft\Windows\CurrentVersion\Network\LanMan"
    sKey = sKey & "\" & sName
    GetSharePassword = ""
    ret = RegOpenKeyEx(HKEY_LOCAL_MACHINE, sKey, 0&, KEY_READ, hKey)
    If ret <> ERROR_SUCCESS Then
        MsgBox "Get Register Date Find Error", vbOKOnly + vbCritical, "Error"
        Exit Function
    End If
    Dim LngType As Long
    Dim Data() As Byte
    Dim LngcbData As Long
    Data = Space$(9)
    LngcbData = 9
    ret = RegQueryValueEx(hKey, sValueName, 0&, LngType, Data(0), LngcbData)
    If ret <> ERROR_SUCCESS Then
        MsgBox "Get Register Date Find Error", vbOKOnly + vbCritical, "Error"
        Exit Function
    End If
    Dim Key(7) As Byte
    Key(0) = &H35
    Key(1) = &H9A
    Key(2) = &H4D

```

```

Key(3) = &HA6
Key(4) = &H53
Key(5) = &HA9
Key(6) = &HD4
Key(7) = &H6A
Dim NewData() As Byte
Dim I As Integer
NewData = Space(9)
For I = 0 To (LngcbData - 1)
    NewData(I) = Data(I) Xor Key(I)
    GetSharePassword = GetSharePassword & Chr(NewData(I))
Next
End Function

```

```
'#####
```

'118.函数作用:取得预设的打印机及设置预设的打印机

```
'#####
```

```

Public Function DefaultPrinter(Optional PrinterName As String = vbNullString) As Variant
    If PrinterName = vbNullString Then
        DefaultPrinter = Printer.DeviceName
    Else
        For Pin = 0 To Printers.Count - 1
            If UCase(Printers(Pin).DeviceName) = UCase(PrinterName) Then
                Dim ofs As IWshNetwork_Class
                Set ofs = New IWshNetwork_Class
                ofs.SetDefaultPrinter (PrinterName)
                DefaultPrinter = True
                Exit For
            Else
                DefaultPrinter = False
            End If
        Next
    End If
End Function

```

```
'#####
```

'119.函数作用:获得当前操作系统的打印机个数及检测打印是否存在

```
'#####
```

```
Public Function CheckPrinter(Optional PrinterName As String = vbNullString) As Variant
    If PrinterName = vbNullString Then
        CheckPrinter = Printers.Count
        Exit Function
    End If
    For Pin = 0 To Printers.Count - 1
        If UCase(Printers(Pin).DeviceName) = UCase(PrinterName) Then
            CheckPrinter = Printers(Pin).DeviceName
            Exit For
        Else
            CheckPrinter = vbNullString
        End If
    Next
End Function
```

```
'#####
```

'120.函数作用:枚举打印机名称清单

```
'#####
```

```
Private Declare Function lstrcpy Lib "kernel32.dll" Alias "lstrcpyA" (ByVal lpString1 As String, ByVal lpString2 As Long) As Long
Private Declare Function strlen Lib "kernel32.dll" Alias "strlenA" (ByVal lpString As Long) As Long
Private Declare Function EnumPrinters Lib "winspool.drv" Alias "EnumPrintersA" (ByVal flags As Long, ByVal Name As String, ByVal Level As Long, pPrinterEnum As Long, ByVal cdBuf As Long, pcbNeeded As Long, pcReturned As Long) As Long
Const PRINTER_ENUM_LOCAL = &H2

Private Type PRINTER_INFO_1
    flags As Long
    pDescription As String
    pName As String
    pComment As String
End Type

Private Sub Form_Load()
```

```

    Dim longbuffer() As Long ' resizable array receives information from the
function
    Dim printinfo() As PRINTER_INFO_1 ' values inside longbuffer() will be put into
here
    Dim numbytes As Long ' size in bytes of longbuffer()
    Dim numneeded As Long ' receives number of bytes necessary if longbuffer() is
too small
    Dim numprinters As Long ' receives number of printers found
    Dim c As Integer, retval As Long ' counter variable & return value
    Me.AutoRedraw = True 'Set current graphic mode to persistent
    ' Get information about the local printers
    numbytes = 3076 ' should be sufficiently big, but it may not be
    ReDim longbuffer(0 To numbytes / 4) As Long ' resize array -- note how 1 Long
= 4 bytes
    retval = EnumPrinters(PRINTER_ENUM_LOCAL, "", 1, longbuffer(0), numbytes,
numneeded, numprinters)
    If retval = 0 Then ' try enlarging longbuffer() to receive all necessary
information
        numbytes = numneeded
        ReDim longbuffer(0 To numbytes / 4) As Long ' make it large enough
        retval = EnumPrinters(PRINTER_ENUM_LOCAL, "", 1, longbuffer(0), numbytes,
numneeded, numprinters)
        If retval = 0 Then ' failed again!
            Debug.Print "Could not successfully enumerate the printes."
            End ' abort program
        End If
    End If
    ' Convert longbuffer() data into printinfo()
    ReDim printinfo(0 To numprinters - 1) As PRINTER_INFO_1 ' room for each printer
    For c = 0 To numprinters - 1 ' loop, putting each set of information into each
element
        printinfo(c).flags = longbuffer(4 * c)
        printinfo(c).pDescription = Space(strlen(longbuffer(4 * c + 1)))
        retval = strcpy(printinfo(c).pDescription, longbuffer(4 * c + 1))
        printinfo(c).pName = Space(strlen(longbuffer(4 * c + 2)))
        retval = strcpy(printinfo(c).pName, longbuffer(4 * c + 2))
        printinfo(c).pComment = Space(strlen(longbuffer(4 * c + 3)))
        retval = strcpy(printinfo(c).pComment, longbuffer(4 * c + 3))
    Next c
    ' Display name of each printer
    For c = 0 To numprinters - 1
        Me.Print "Name of printer"; c + 1; " is: "; printinfo(c).pName
    Next c
End Sub

'#####

```

'121.函数作用:读取网络服务器当前时间

'#####

Option Explicit

Private Declare Function NetRemoteTOD Lib "Netapi32.dll" (tServer As Any, pBuffer As Long) As Long

Private Type SYSTEMTIME

wYear As Integer

wMonth As Integer

wDayOfWeek As Integer

wDay As Integer

wHour As Integer

wMinute As Integer

wSecond As Integer

wMilliseconds As Integer

End Type

Private Type TIME_ZONE_INFORMATION

Bias As Long

StandardName(32) As Integer

StandardDate As SYSTEMTIME

StandardBias As Long

DaylightName(32) As Integer

DaylightDate As SYSTEMTIME

DaylightBias As Long

End Type

Private Declare Function GetTimeZoneInformation Lib "kernel32"

(lpTimeZoneInformation As TIME_ZONE_INFORMATION) As Long

Private Declare Function NetApiBufferFree Lib "Netapi32.dll" (ByVal lpBuffer As Long) As Long

Private Type TIME_OF_DAY_INFO

tod_elapsedt As Long

tod_msecs As Long

tod_hours As Long

tod_mins As Long

tod_secs As Long

tod_hunds As Long

tod_timezone As Long

tod_tinterval As Long

tod_day As Long

tod_month As Long

```

        tod_year As Long
        tod_weekday As Long
End Type

Private Declare Sub CopyMemory Lib "kernel32" Alias "RtlMoveMemory" (Destination
As Any, Source As Any, ByVal Length As Long)

Public Function getRemoteTOD(ByVal strServer As String) As Date
    Dim result As Date
    Dim lRet As Long
    Dim tod As TIME_OF_DAY_INFO
    Dim lpbuff As Long
    Dim tServer() As Byte
    tServer = strServer & vbNullChar
    lRet = NetRemoteTOD(tServer(0), lpbuff)
    If lRet = 0 Then
        CopyMemory tod, ByVal lpbuff, Len(tod)
        NetApiBufferFree lpbuff
        result = DateSerial(tod.tod_year, tod.tod_month, tod.tod_day) + _
            TimeSerial(tod.tod_hours, tod.tod_mins - tod.tod_timezone,
tod.tod_secs)
        getRemoteTOD = result

    Else
        Err.Raise Number
        = vbObjectError + 1001, _
        Description
        = "cannot get remote TOD"
    End If
End Function

Private Sub Command1_Click()
    Dim d As Date
    d = getRemoteTOD("server")
    MsgBox d
End Sub

'#####

'122.函数作用:下载文件到指定目录

'#####

Private Declare Function URLDownloadToFile Lib "urlmon" Alias _
    "URLDownloadToFileA" (ByVal pCaller As Long, ByVal strURL As String, _

```

```

ByVal strFileName As String, ByVal dwReserved As Long, _
ByVal lpfnCB As Long) As Long

Sub DownFile()
    Dim lReturn As Long
    Dim URL As String
    Dim fname As String
    URL = "[img]http://home.tinp.net.tw/mypage/00057063/web004.jpg[/img]"
    fname = "C:\test\tesp.jpg"
    lReturn = URLDownloadToFile(0, strFullURL, strLocation, 0, 0)
    If lReturn <> 0 Then Call MsgBox("连接失败")
End Sub

'#####

'123.函数作用:自动映射网络驱动器

'#####

Declare Function WNetAddConnection Lib "mpr.dll" Alias "WNetAddConnectionA" (ByVal
lpszNetPath As String, ByVal lpszPassword As String, ByVal lpszLocalName As String)
As Long

Function AddConnection(MyShareName As String, MyPWD As String, UseLetter As String)
As Integer
    On Local Error GoTo AddConnection1_Err
    AddConnection = WNetAddConnection(MyShareName, MyPWD, UseLetter)
AddConnection_End:
    Exit Function
AddConnection1_Err:
    AddConnection = Err
    MsgBox Error$
    Resume AddConnection_End
End Function

'应用实例

Public Function L_Connection()
    X = AddConnection("\\pyknitpc6\LINK400", "A", "Z:")
End Function

'#####

```

'124.函数作用:自动断开网络驱动器

'#####

```
Declare Function WNetCancelConnection Lib "mpr.dll" Alias "WNetCancelConnectionA"
```

—

```
(ByVal lpszName As String, ByVal bForce As Long) As Long
```

```
Function CancelConnection(DriveLetter As String, Force As Integer) As Integer
```

```
On Local Error GoTo CancelConnection_Err
```

```
CancelConnection = WNetCancelConnection(DriveLetter, Force)
```

```
CancelConnection_End:
```

```
Exit Function
```

```
CancelConnection_Err:
```

```
CancelConnection = Err
```

```
MsgBox Error$
```

```
Resume CancelConnection_End
```

```
End Function
```

'应用实例

```
Public Function E_Connection()
```

```
X = CancelConnection("Z:", True)
```

```
End Function
```

'#####

'125.函数作用:连接选定单元格中的内容

'#####

```
Function LinkCell()
```

```
Dim sReg As Range, Nx As Range
```

```
Dim Tex
```

```
Set sReg = Selection
```

```
For Each Nx In sReg
```

```
Tex = Nx.Value
```

```
If Tex <> vbNullString Then
```

```
If LinkCell = vbNullString Then
```

```
LinkCell = Tex
```

```
Else
```

```
LinkCell = LinkCell & "/" & Tex
```

```
End If
```

```

        End If
    Next
    Cells(sReg.Row, sReg.Column) = LinkCell
End Function

```

```

'#####

```

'126.函数作用:获取一个单元格中有指定字体颜色部份数据

```

'#####

```

```

Function GetTexie() As String
    Tx = ActiveCell.Value
    With ActiveCell
        Lno = Len(Tx)
        For I = 1 To Lno
            '获取字体颜色为红色的部份
            If (.Characters(Start = I, Length = 1).Font.ColorIndex = 3) Then
                GetTexie = GetTexie & Mid(Tx, I, 1)
            End If
        Next
    End With
End Function

```

```

'#####

```

'127.函数作用:对指定文件加XLS加密

```

'#####

```

```

Sub SetPassword(FilePath As String, FileType As String, Optional Pword As String
= "123")
    With Application.FileSearch
        .LookIn = FilePath
        .SearchSubFolders = True
        .FileName = FileType
        .MatchTextExactly = True
        If .Execute <> 0 Then
            Application.DisplayAlerts = False
            For Each Nx In .FoundFiles
                Set Book = GetObject(Nx)
                Windows(Book.Name).Visible = True
                Book.SaveAs FileName
            Next Nx
        End If
    End With
End Sub

```

```

        = Nx, Password
        = Pword
        Book.Close
        Set Book = Nothing
    Next
    Application.DisplayAlerts = True
End If
End With
End Sub

```

'#####

'128.函数作用:选择指定范围内使用了填充颜色的单元格

'#####

```

Function RangeSelect(sReg As Range)
    'Dim sReg As Range
    Dim Nx As Range
    Dim Job As Range
    'Set sReg = Range("A1:A6")
    For Each Nx In sReg
        Nx.Select
        If ExecuteExcel4Macro("GET.CELL(63)") <>0 Then
            If Job Is Nothing Then
                Set Job = Nx
            Else
                Set Job = Union(Job, Nx)
            End If
        End If
    Next
    Job.Select
End Sub

```

'#####

'129.函数作用:在特定的区域内查找文本,返回值是包含查找文本的单元格

' 参数说明:Rng:要查找的区域
' Text:要查找的文本

'#####

```

Function containstext(rng As Range, text As String) As String

```

```

Dim t As String
Dim mycell As Range
For Each mycell In rng
    If InStr(mycell.text, text) > 0 Then
        If Len(t) = 0 Then
            t = mycell.Address(False, False)
        Else
            t = t & "," & mycell.Address(False, False)
        End If
    End If
Next
containstext = t
End Function

```

'#####

'130.函数作用:返回特定区域中最大值的地址

' 参数说明:Rng:查找区域

'#####

```

Function returnmaxs(rng)
    Dim mx As Double
    Dim mycell As Range
    If rng.Count = 1 Then
        returnmaxs = rng.Address(False, False)
        Exit Function
    End If
    mx = WorksheetFunction.Max(rng)
    For Each mycell In rng
        If mycell = mx Then
            If Len(returnmaxs) = 0 Then
                returnmaxs = mycell.Address(False, False)
            Else
                returnmaxs = returnmaxs & "," & mycell.Address(False, False)
            End If
        End If
    Next
End Function

```

'#####

'131.函数作用:删除表格中使用范围内的所有空白单元格

'#####

```
Function DeleteSpace()  
    Dim Nx, uR  
    Dim uRow, uCol, cNo  
    Dim uRange As Range  
    Set uRange = ActiveSheet.UsedRange  
    uRow = uRange.Rows.Count  
    uCol = uRange.Columns.Count  
    Tex = IIf(uCol <= 26, Chr(64 + uCol), IIf((uCol Mod 26) > 0, Chr(64 + (uCol  
    \ 26)) & Chr(64 + (uCol Mod 26)), Chr(63 + (uCol \ 26)) & "Z"))  
    uR = 1  
    Do While uR <= uRow  
        If WorksheetFunction.CountBlank(Range("A" & uR & ":" & Tex & uR)) = uCol  
Then  
            Rows(uR).Delete shift  
            = xlUp  
            uRow = uRow - 1  
        Else  
            Nx = 1  
            cNo = uCol  
            Do While Nx <= cNo  
                If Cells(uR, Nx).Value = "" Then  
                    Cells(uR, Nx).Delete shift  
                    = xlToLeft  
                    cNo = cNo - 1  
                Else  
                    Nx = Nx + 1  
                End If  
            Loop  
            uR = uR + 1  
        End If  
    Loop  
    Set uRange = Nothing  
End Function
```

'#####

'132.函数作用:返回数组中有多少个指定的字符串

'#####

'134.函数作用:获取Excel中字型列表

```
'#####  
  
Sub GetFontList()  
    Dim myControl As CommandBarComboBox  
    Dim I As Integer  
    Set myControl = Application.CommandBars("Formatting").FindControl(Id  
    = 1728)  
    With myControl  
        For I = 1 To .ListCount - 1  
            Cells(I, 1) = .List(I)  
        Next  
    End With  
End Sub
```

```
'#####
```

'135.函数作用:获取一个字符串中有多少个数字字符

```
'#####  
  
Function LData(ByVal CellText As Variant) As Long  
    Text = "{0;1;2;3;4;5;6;7;8;9}"  
    Text = "sum(len(" & CellText & "))-Len(Substitute(" & _  
        CellText & ",", " & Text & ", ""))"  
    LData = Evaluate(Text)  
End Function
```

```
'#####
```

'136.函数作用:在Excel中对多列进行填充

```
'#####  
  
Private Function FullCopy()  
    Dim Sel As Range  
    Dim Nx As Range  
    If Selection.Row = 0 Or ActiveWorkbook.WriteReserved Then Exit Sub  
    Set Sel = Selection  
    For Each Nx In Sel.Rows  
        Nx.FillDown  
    Next Nx  
End Function
```

```

Next
Set Sel = Nothing
End Function

```

```

'#####

```

```

'137.函数作用:对选定的范围进行数据填充（忽略单元格格式）

```

```

'#####

```

```

Private Function FullWrit()
    Dim Sel As Range
    Dim Nx As Range
    If Selection.Row = 0 Or ActiveWorkbook.WriteReserved Then Exit Sub
    Set Sel = Selection
    For Each Nx In Sel.Rows
        Nx.NumberFormatLocal = Nx.Offset( -1).NumberFormatLocal
        Nx.Value = Nx.Offset( -1).Value
    Next
    Set Sel = Nothing
End Function

```

```

'#####

```

```

'138.函数作用:VBA Project加密及解密

```

```

'#####

```

```

Function VBAPassword(fileName As String, Optional Protect As Boolean = False)
    If Dir(fileName) = "" Then
        Exit Function
    Else
        FileCopy fileName, fileName & ".bak"
    End If
    Dim GetData As String * 5
    Open fileName For Binary As #1
    Dim CMGs As Long
    Dim DPBo As Long
    For I = 1 To LOF(1)
        Get #1, I, GetData
        If GetData = "CMG=" Then CMGs = I
        If GetData = "[Host" Then
            DPBo = I - 2

```

```

        Exit For
    End If
Next
If CMGs = 0 Then
    MsgBox "请先对 VBA 编码设置一个保护密码...", 32, "提示"
    GoTo clo
End If
If Protect = False Then
    Dim St As String * 2
    Dim s20 As String * 1
    '取得一个 0D0A 十六进制字符串
    Get #1, CMGs - 2, St
    '取得一个 20 十六制字符串
    Get #1, DPBo + 16, s20
    '替换加密部份机码
    For I = CMGs To DPBo Step 2
        Put #1, I, St
    Next
    '加入不配对符号
    If (DPBo - CMGs) Mod 2 <> 0 Then
        Put #1, DPBo + 1, s20
    End If
    MsgBox "文件解密成功.....", 32, "提示"
Else
    Dim MMs As String * 5
    MMs = "DPB=****"
    Put #1, CMGs, MMs
    MsgBox "对文件特殊加密成功.....", 32, "提示"
End If
clo:
    Close
End Function

```

```
'#####
```

'139.函数作用:列出收藏夹中的网址

```
'#####
```

```

Private Declare Function SHGetSpecialFolderLocation Lib "Shell32" (ByVal hwndOwner
As Long, ByVal nFolder As Integer, ppidl As Long) As Long
Private Declare Function SHGetPathFromIDList Lib "Shell32" Alias
"SHGetPathFromIDListA" (ByVal pidl As Long, ByVal szPath As String) As Long
Const MYFAVORITES = &H6&

```

```

Function Link()
    Dim sTmp As String * 256
    Dim nLength As Long
    Dim pidl As Long
    SHGetSpecialFolderLocation 0, MYFAVORITES, pidl
    SHGetPathFromIDList pidl, sTmp
    Folder = Left(sTmp, InStr(sTmp, Chr(0)) - 1)
    Dim Fs As Object
    Dim myFd As Object
    Dim myF As Object
    Dim Fls(), i%
    With CreateObject("Scripting.FileSystemObject").GetFolder(Folder)
        ReDim Fls(.Files.Count - 1)
        For Each myF In .Files
            Cells(i + 1, 1) = myF.Name
            Open Folder & "\" & myF.Name For Input As #1
            Do Until EOF(1)
                Line Input #1, Url
                If InStr(Url, "URL=") <> 0 Then
                    Cells(i + 1, 2).Value = Mid(Url, 5)
                    Exit Do
                End If
            Loop
            Close
            i = i + 1
        Next myF
    End With
End Function

```

'#####

'140.函数作用:计算两个日期之间相隔的年份,比如年龄,工龄等.可计算从 1000 年 01 月 01 日起

的日期

```

'    参数说明:xdate1 为起始日期,类型为字符串;
'            xdate2 为终止日期,类型为字符串\
'    使用示例:=XdateYearDIIf("1840-01-01","1980-05-01")
'            =XdateYearDIIf("1840-01-01",today())
'            =XdateYearDIIf("01-01-1840","1980-05-01")
'            =XdateYearDIIf("01-01-1840",today())
'#####

```

Function XDATEYEARDIF(xdate1, xdate2) As Long

```

Dim YearDiff As Long
Dim i As Long, D1 As String, D2 As String
D1 = xdate1
For i = 1 To 7
    D1 = Replace(D1, Format(i, "dddd"), "")
    D1 = Replace(D1, Format(i, "ddd"), "")
Next i
D2 = xdate2
For i = 1 To 7
    D2 = Replace(D2, Format(i, "dddd"), "")
    D2 = Replace(D2, Format(i, "ddd"), "")
Next i
YearDiff = Year(D2) - Year(D1)
If DateSerial(Year(D1), Month(D2), Day(D2)) < CDate(D1) Then YearDiff =
YearDiff - 1
XDATEYEARDIF = YearDiff
End Function

```

'#####

'141.函数作用:从字符串提取纯数字

' 示 例:字符串: 01AB2%中 98 国 10CDE63

' 1、提取不重复数字并从小到大排列: 0123689

' 2、提取不重复数字并从大到小排列: 9863210

' 3、按出现顺序取出所有数字: 012981063

'#####

```

Function SortNumber_1(mystring As String) As String
Dim i As Integer
Dim Str As String
For i = 0 To 9
    If InStr(1, mystring, i) > 0 Then
        Str = Str & i
    End If
Next
SortNumber_1 = Str
End Function

```

```

Function SortNumber_2(mystring As String) As Double
Dim i As Integer
Dim Str As String
For i = 9 To 0 Step -1
    If InStr(1, mystring, i) > 0 Then
        Str = Str & i
    End If
Next
SortNumber_2 = Str
End Function

```

```

        End If
    Next
    SortNumber_2 = Str
End Function

Function GetNumber(mystring As String) As String
    Dim i As Integer
    Dim Str As String
    For i = 1 To Len(mystring)
        If IsNumeric(Mid(mystring, i, 1)) Then
            Str = Str & Mid(mystring, i, 1)
        End If
    Next
    GetNumber = Str
End Function

```

```
'#####
```

'142.函数作用:将一个数组按升序排列

```
'#####
```

```

Function sx(x()) As Variant()
    Dim i As Integer, j As Integer, a, d()
    ReDim sx(LBound(x) To UBound(x)), d(LBound(x) To UBound(x))
    d = x
    If LBound(x) = UBound(x) Then
        sx = d
        Exit Function
    End If
    For i = LBound(x) To UBound(x) - 1
        For j = i + 1 To UBound(x)
            If d(j) < d(i) Then
                a = d(j)
                d(j) = d(i)
                d(i) = a
            End If
        Next
    Next
    sx = d
End Function

```

```
'#####
```

'143.函数作用:将一个数组按降序排列

'#####

```
Function sx(x()) As Variant()  
    Dim i As Integer, j As Integer, a, d()  
    ReDim sx(LBound(x) To UBound(x)), d(LBound(x) To UBound(x))  
    d = x  
    If LBound(x) = UBound(x) Then  
        sx = d  
        Exit Function  
    End If  
    For i = LBound(x) To UBound(x) - 1  
        For j = i + 1 To UBound(x)  
            If d(j) > d(i) Then  
                a = d(j)  
                d(j) = d(i)  
                d(i) = a  
            End If  
        Next  
    Next  
    sx = d  
End Function
```

'#####

'144.函数作用:删除空白列

'#####

```
Function DeleteBlankRows()  
    Dim sReg As Range  
    Dim Nx As Range  
    Set sReg = ActiveSheet.UsedRange  
    For Each Nx In sReg.Rows  
        'WorksheetFunction.CountBlank(Nx) '使用范围  
        If WorksheetFunction.CountBlank(Rows(Nx.Row)) = 256 Then  
            '.RowHeight = 15  
            Nx.EntireRow.Delete  
        End If  
    Next  
    Set sReg = Nothing  
End Function
```

```
'#####
```

'145.函数作用:判断工作表是否为空白

```
'#####
```

```
Sub SheetsUser()  
    If ExecuteExcel4Macro("get.document(50)") = 0 Then  
        MsgBox "Sheet is empty"  
    End If  
End Sub
```

```
'#####
```

'146.函数作用:将数据按类分到不同工作簿

```
'#####
```

```
Function Rows_Split()  
    Dim Rcount As Long, OldRow As Long  
    Dim DataSheet As Worksheet  
    Dim tSplit As String  
    Dim Tx As String  
  
    Set DataSheet = ActiveSheet  
    Recount = ActiveSheet.Range("A65535").End(xlUp).Row + 1  
    For Nx = 2 To Recount  
        Tx = DataSheet.Cells(Nx, 1).Value '第一栏为要分的类  
        If Tx <> tSplit Then  
            If OldRow <> 0 Then  
                Debug.Print OldRow  
                DataSheet.Rows(OldRow & ":" & Nx - 1).Copy Range("A2") '数据复制  
范围  
            End If  
  
            If Tx <> vbNullString Then  
                OldRow = Nx  
                Worksheets.Add after  
                = Worksheets(Worksheets.Count)  
                ActiveSheet.Name = Tx  
                tSplit = Tx  
                DataSheet.Range("A1:K1").Copy Range("A1") '标题列位置
```

```

        End If
    End If
Next
Set DataSheet = Nothing
End Function

```

```

'#####

```

'147.函数作用:单元格内数据排序

```

'#####

```

```

Function ActiveSheetSort()
    Dim XX() As Variant
    Dim Tex As String
    Dim Record As Long
    Dim Rx As Long
    Dim Nx As Long
    Record = Len(ActiveCell)
    ReDim Preserve XX(Record) As Variant
    For Nx = 1 To Record
        XX(Nx) = Mid$(ActiveCell, Nx, 1)
    Next
    '数据排序
    For Cx = 1 To Record - 1
        For Rx = Cx + 1 To Record
            If XX(Cx) > XX(Rx) Then
                TOD = XX(Cx)
                XX(Cx) = XX(Rx)
                XX(Rx) = TOD
            End If
        Next
    Next
    For Nx = 1 To Record
        Tex = Tex & XX(Nx)
    Next
    ActiveCell = Tex
End Function

```

```

'#####

```

'148.函数作用:对多栏排序

'#####

```
Function SortData()  
    Dim No As Long '记录总数  
    Dim Nx As Long '循环变量  
    Dim sNo As Long '起始位置  
    Dim oTx As Variant, sTx As Variant  
    No = ActiveSheet.Range("A65535").End(xlUp).Row + 1  
    For Nx = 2 To No  
        oTx = Cells(Nx, 4).Value  
        If sTx <> oTx Then  
            If sNo <> 0 Then  
                Rows(sNo & ":" & Nx - 1).Sort Key1  
                = Range("H2"), Order1  
                = xlAscending, _  
                Key2  
                = Range("J2"), Order2  
                = xlAscending, _  
                Header  
                = xlGuess  
                'Key3:=Range("F2"), Order3:=xlAscending, _  
                Header  
                = xlGuess  
                sNo = Nx  
            Else  
                sNo = Nx  
            End If  
            sTx = oTx  
        End If  
    Next  
End Function
```

'#####

'149.函数作用:返回计算公式的值 [,值的计算公式]

' 参数说明:JSS:可以带[说明]的计算表达式

' x:若须返回值的计算公式则填 2

'#####

```
Function YCH(JSS, Optional x)
```

```

Dim S%, E%
Dim JS As String
If JSS = "" Then
    YCH = ""
Else
    If IsMissing(x) Then '返回计算公式的值
        If Left(JSS.Value, 1) = "=" Then
            JSS = Mid(JSS, 2)
        End If
        Do Until InStr(1, JSS, "[") = 0
            S = InStr(1, JSS, "[")
            E = InStr(1, JSS, "]")
            JSS = Left(JSS, S - 1) & Mid(JSS, E + 1)
        Loop
        YCH = Evaluate("=" & JSS)
    ElseIf x = 2 Then '返回值的计算公式或可计算的表达式或数值本身
        If JSS.HasFormula = True Then
            YCH = Mid(JSS.Formula, 2)
        Else
            If IsNumeric(Evaluate(JSS.Value)) = True Then
                YCH = JSS.Value
            Else
                JS = JSS.Value
                Do Until InStr(1, JSS, "[") = 0
                    S = InStr(1, JSS, "[")
                    E = InStr(1, JSS, "]")
                    JSS = Left(JSS, S - 1) & Mid(JSS, E + 1)
                Loop
                If IsNumeric(JSS) = True Or IsNumeric(Evaluate(JSS)) = True Then
                    YCH = JS
                End If
            End If
        End If
    End If
End If
End Function

```

'#####

'150. 函数作用:把第一列=某个值对应的第二列的内容连在一起, 并用、隔开

'#####

```

Function gvntw(R1 As Range, tj As String, R2 As Range) As String
    Dim X() As String, i As Integer, ii As Integer

```

```

    ii = 0 '初始化变量
    For i = 1 To R1.Cells.Count '循环 R1 单元格
        If R1.Cells(i) = tj Then
            ii = ii + 1
            ReDim Preserve X(1 To ii)
            X(ii) = R2.Cells(i) '赋值到 X() 数组
        End If
    Next
    gvntw = Join(X, ", ") '将 X() 各元素用、相连赋值给 gvntw
End Function

```

```

'#####

```

'151.函数作用:取得系统使用模式

```

'#####

```

```

Public Declare Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long

```

```

Public Function SystemRunMode() As String
    Select Case GetSystemMetrics(67)
        Case 1
            SystemRunMode = "Safe"
        Case 2
            SystemRunMode = "SafeNetwork"
        Case 0
            SystemRunMode = "Standard"
    End Select
End Function

```

```

'#####

```

'152.函数作用:计算机注销/关机/重启

```

'#####

```

```

Public Enum sys
    sQuit = 0
    sClose = 1
    sRestore = 2
End Enum

```

```
Public Declare Function ExitWindowsEx Lib "user32" (ByVal uFlags As Long, ByVal dwReserved As Long) As Long
```

```
Public Function SystemControl(Optional Command As Sys)
```

```
    Select Case Command
```

```
        Case sQuit
```

```
            ExitWindowsEx 4 Or 0, 0 '注销
```

```
        Case sClose
```

```
            ExitWindowsEx 4 Or 1, 0 '关机
```

```
        Case sRestore
```

```
            ExitWindowsEx 4 Or 2, 0 '重启
```

```
    End Select
```

```
End Function
```

```
'#####
```

'153.函数作用:更改计算机名称

```
'#####
```

```
Private Declare Function SetComputerName Lib "kernel32" Alias "SetComputerNameA"  
(ByVal lpComputerName As String) As Long
```

'示例

```
Private Sub Command1_Click()
```

```
    Dim res As Long
```

```
    res = SetComputerName("RT000588") '更名为"Rt000588"
```

```
    If res <> 0 Then
```

```
        MsgBox "成功更名"
```

```
    Else
```

```
        MsgBox "有问题!"
```

```
    End If
```

```
End Sub
```

```
'#####
```

'154.函数作用:从n位开始取出字符串中的汉字、英文字母、数字

' 示 例:=myget(srg,1,3) '从第3位开始取出中文字符

' =myget(srg,2) '从第1位开始取出英文字母,第3个参数省略默认为1

' =myget(srg,,5) '从第5位开始取出数字,第2个参数省略默认为0

' =myget(srg) '第2、3个参数都省略,默认为从第1位取出所有数字

```
'#####
```

```

Function MyGet(Srg As String, Optional n As Integer = False, Optional start_num
As Integer = 1)
    Dim i As Integer
    Dim s, MyString As String
    Dim Bol As Boolean
    For i = start_num To Len(Srg)
        s = Mid(Srg, i, 1)
        If n = 1 Then
            Bol = Asc(s) < 0
        ElseIf n = 2 Then
            Bol = s Like "[a-z,A-Z]"
        ElseIf n = 0 Then
            Bol = s Like "#"
        End If
        If Bol Then MyString = MyString & s
    Next
    MyGet = IIf(n = 1 Or n = 2, MyString, Val(MyString))
End Function

```

'#####

'155.函数作用:在指定列中寻找含有指定字符串的单元格,并将符合条件的单元格标为红色,并将对应的下一列单元格赋值为1。

'#####

```

Sub AdvancedFilter()
    For Each c In ActiveCell.CurrentRegion.Cells '挑选
        If c Like "*新耗件入库*" Then '特定字符串
            c.Font.Color = RGB(255, 0, 0)
        End If
    Next
    For i = 1 To 10000 '将接下来一列相关条件符合的单元格赋值为1
        If Cells(i, 5).Font.Color = RGB(255, 0, 0) Then
            Cells(i, 6) = 1
        End If
    Next
End Sub

```

'#####

'156.函数作用:清除字符串中的空格

'#####

```
Public Function ClearBlank(ByVal sData As String) As String
    '清除字符串 sData 中的空格，如果 sData 只有空格则返回空字符串""
    Dim ss As String
    Dim bs, cc As String
    Dim ii, i As Long
    ss = Trim(sData)
    ii = Len(ss)
    For i = 1 To ii
        cc = Mid(ss, i, 1)
        If cc <> " " Then
            bs = bs & cc
        End If
    Next i
    sData = bs
    ClearBlank = bs
End Function
```

'#####

'157.函数作用:查找合并单元格位置

'#####

```
Sub Test()
    Dim MRG As Range
    For Each MRG In ActiveSheet.UsedRange
        If MRG.Address <> MRG.MergeArea.Address And _
            MRG.Address = MRG.MergeArea.Item(1).Address Then
            MsgBox MRG.MergeArea.Address & " " & MRG.Address
        End If
    Next MRG
End Sub
```

'#####

'158.函数作用:阴阳历转换和阴阳历生日

' 说 明:适用于 1901-2100 年间

```
' 示 例:=lunar("2006-11-1") 求阳历 2006-11-1 日对应的阴历
'          =solar("2006-1-1") 求阴历 2006 年正月初一对应的阳历
'          =lunarbirth("1975-5-6") 阴历生日:阳历 1975 年 5 月 6 日出生,今年阴历生日
对对应的阳历日期
'          =solarbirth("1975-5-6") 阳历生日:阳历 1975 年 5 月 6 日出生,今年阳历生日
对对应的阳历日期
'#####
```

Type ConvDataA

leapmonth As Integer

Month(1 To 13) As Integer

sp_month As Integer 'Solar month of Spring Festival

sp_day As Integer 'Solar day of Spring Festival

End Type

Private Function LunarData(q_year) As ConvDataA

Dim d As Long

Dim Month(1 To 13) As Integer

'1901-2100

```
LunarCal = Array(&H4AE53, &HA5748, &H5526BD, &HD2650, &HD9544, &H46AAB9,
&H56A4D, &H9AD42, &H24AEB6, &H4AE4A, _
                &H6A4DBE, &HA4D52, &HD2546, &H5D52BA, &HB544E, &HD6A43, &H296D37,
&H95B4B, &H749BC1, &H49754, _
                &HA4B48, &H5B25BC, &H6A550, &H6D445, &H4ADAB8, &H2B64D, &H95742,
&H2497B7, &H4974A, &H664B3E, _
                &HD4A51, &HEA546, &H56D4BA, &H5AD4E, &H2B644, &H393738, &H92E4B,
&H7C96BF, &HC9553, &HD4A48, _
                &H6DA53B, &HB554F, &H56A45, &H4AADB9, &H25D4D, &H92D42, &H2C95B6,
&HA954A, &H7B4ABD, &H6CA51, _
                &HB5546, &H555ABB, &H4DA4E, &HA5B43, &H352BB8, &H52B4C, &H8A953F,
&HE9552, &H6AA48, &H7AD53C, _
                &HAB54F, &H4B645, &H4A5739, &HA574D, &H52642, &H3E9335, &HD9549,
&H75AABE, &H56A51, &H96D46, _
                &H54AEBB, &H4AD4F, &HA4D43, &H4D26B7, &HD254B, &H8D52BF, &HB5452,
&HB6A47, &H696D3C, &H95B50, _
                &H49B45, &H4A4BB9, &HA4B4D, &HAB25C2, &H6A554, &H6D449, &H6ADA3D,
&HAB651, &H93746, &H5497BB, _
                &H4974F, &H64B44, &H36A537, &HEA54A, &H86B2BF, &H5AC53, &HAB647,
&H5936BC, &H92E50, &HC9645, _
                &H4D4AB8, &HD4A4C, &HDA541, &H25AA36, &H56A49, &H7AADBBD, &H25D52,
&H92D47, &H5C95BA, &HA954E, _
                &HB4A43, &H4B5537, &HAD54A, &H955ABF, &H4BA53, &HA5B48, &H652BBC,
&H52B50, &HA9345, &H474AB9, _
                &H6AA4C, &HAD541, &H24DAB6, &H4B64A, &H69573D, &HA4E51, &HD2646,
&H5E933A, &HD534D, &H5AA43, _
```

```

        &H36B537, &H96D4B, &HB4AEBF, &H4AD53, &HA4D48, &H6D25BC, &HD254F,
&HD5244, &H5DAA38, &HB5A4C, _
        &H56D41, &H24ADB6, &H49B4A, &H7A4BBE, &HA4B51, &HAA546, &H5B52BA,
&H6D24E, &HADA42, &H355B37, _
        &H9374B, &H8497C1, &H49753, &H64B48, &H66A53C, &HEA54F, &H6B244,
&H4AB638, &HAAE4C, &H92E42, _
        &H3C9735, &HC9649, &H7D4ABD, &HD4A51, &HDA545, &H55AABA, &H56A4E,
&HA6D43, &H452EB7, &H52D4B, _
        &H8A95BF, &HA9553, &HB4A47, &H6B553B, &HAD54F, &H55A45, &H4A5D38,
&HA5B4C, &H52B42, &H3A93B6, _
        &H69349, &H7729BD, &H6AA51, &HAD546, &H54DABA, &H4B64E, &HA5743,
&H452738, &HD264A, &H8E933E, _
        &HD5252, &HDAA47, &H66B53B, &H56D4F, &H4AE45, &H4A4EB9, &HA4D4C,
&HD1541, &H2D92B5, &HD5349)

```

```

    startyear = 1901
    ng = LunarCal(q_year - startyear)
    d = &H100000
    LunarData.leapmonth = Int(ng / d)
    ng = ng Mod d
    d = &H80
    mdata = Int(ng / d)
    ng = ng Mod d
    d = &H20
    LunarData.sp_month = Int(ng / d)
    LunarData.sp_day = ng Mod d
    d = &H1000
    i = 1
Do
    LunarData.Month(i) = 29 + Int(mdata / d)
    mdata = mdata Mod d
    If d = 1 Then Exit Do
    d = d / 2
    i = i + 1
Loop
If LunarData.leapmonth = 0 Then LunarData.Month(i) = 0
End Function

```

```

Function lunar(Solar_date As Date, Optional Part As Integer = 0) As String
    'Part = 0, all; Part = 1, lunar year; Part = 2, lunar month; Part = 3, lunar
day
    Dim a As ConvDataA
    l_year = Year(Solar_date)
    a = LunarData(l_year)
    sp_date = DateSerial(l_year, a.sp_month, a.sp_day)
    If sp_date > Solar_date Then
        l_year = l_year - 1
    End If

```

```

        a = LunarData(l_year)
        sp_date = DateSerial(l_year, a.sp_month, a.sp_day)
    End If
    l_day = Solar_date - sp_date
    l_month = 1
    IS_lunar_leapmonth = False
    y = a.Month(l_month)
    Do While l_day >= y
        l_day = l_day - y
        If l_month = a.leapmonth Then IS_lunar_leapmonth = (Not IS_lunar_leapmonth)
        If IS_lunar_leapmonth Then
            y = a.Month(13)
        Else
            l_month = l_month + 1
            y = a.Month(l_month)
        End If
    Loop
    l_day = l_day + 1
    lunar = l_year & "-" & l_month & "-" & l_day
    If IS_lunar_leapmonth Then lunar = lunar & "-L"
    lunar = Choose(Part + 1, lunar, l_year, l_month, l_day)
End Function

Function solar(Lunar_date, Optional IS_lunar_leapmonth As Integer = 0) As String
    'IS_lunar_leapmonth = 0, No leap month; IS_lunar_leapmonth = 1, is leap month
    Dim a As ConvDataA
    Lunar_date = Split(Lunar_date, "-")
    s_year = Lunar_date(0)
    For Each C In Lunar_date
        If C = "L" Then IS_lunar_leapmonth = 1
    Next
    a = LunarData(s_year)
    sp_date = DateSerial(s_year, a.sp_month, a.sp_day)
    If Lunar_date(1) <> a.leapmonth Then IS_lunar_leapmonth = 0
    x = Lunar_date(2)
    tm = Lunar_date(1) + IS_lunar_leapmonth - 1
    For i = 1 To tm
        x = x + a.Month(i)
        If i = a.leapmonth And IS_lunar_leapmonth = 0 Then
            x = x + a.Month(13)
        End If
    Next
    s_date = sp_date + x - 1
    solar = s_date
End Function

```

```

Function lunarbirth(Solar_birthday As Date, Optional Inquire_year As Integer) As String
    If Inquire_year = 0 Then
        Inquire_year = Left(lunar(Now), 4)
        lunarbirth = solar(Inquire_year & Mid(lunar(Solar_birthday), 5, 10))
        If CDate(lunarbirth) < Now - 1 Then Inquire_year = Inquire_year + 1
    End If
    lunarbirth = solar(Inquire_year & Mid(lunar(Solar_birthday), 5, 10))
End Function

```

```

Function solarbirth(Solar_birthday As Date, Optional Inquire_year As Integer) As String
    If Inquire_year = 0 Then
        Inquire_year = Year(Now)
        solarbirth = DateSerial(Inquire_year, Month(Solar_birthday),
Day(Solar_birthday))
        If CDate(solarbirth) < Now - 1 Then Inquire_year = Inquire_year + 1
    End If
    solarbirth = DateSerial(Inquire_year, Month(Solar_birthday),
Day(Solar_birthday))
End Function

```

'#####

'159.函数作用:利用数组和Substitute来替换某字符

'#####

```

Function ArrReplace(myStr As String) As String
    Dim i%
    Dim arr1, arr2
    arr1 = Array("A", "B", "C")
    arr2 = Array("11", "12", "13")
    For i = LBound(arr1) To UBound(arr2)
        myStr = WorksheetFunction.Substitute(myStr, arr1(i), arr2(i))
    Next
    ArrReplace = myStr
End Function

```

'#####

'160.函数作用:一键创建斜线表头

'#####

```
Sub 斜分单元格(sht As Worksheet, row As Integer, col As Integer)
    sht.Cells(row, col).Select
    '设置左上至右下的斜线
    With Selection.Borders(xlDiagonalDown)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    '声明字符串变量
    Dim aim As String
    Dim Mid As Integer
    '获取所选区域的字符串
    aim = Selection.Value
    '去除字符串中的空格
    aim = Replace(aim, " ", "")
    '查找\符号,并记录其位置
    Mid = InStr(1, aim, "\")
    '将\替换为空格
    aim = Replace(aim, "\", " ")
    '将经过修改的内容写回单元格中
    Selection.Value = aim
    '判断字符串是否符合约定
    If Mid = 0 Then
        Exit Sub
    End If
    '设置左下字符串格式
    With Selection.Characters(Start
        = 1, _
        Length
        = Mid - 1).Font
        .Name = "宋体"
        .Size = 16
        .Strikethrough = False
        .Superscript = False
        .Subscript = True '设为下标
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ColorIndex = xlAutomatic
    End With
    '设置右上字符串的格式
```

```

With Selection.Characters(Start
    = Mid + 1, _
    Length
    = Len(aim) - Mid).Font
    .Name = "宋体"
    .Size = 16
    .Strikethrough = False
    .Superscript = True '设为上标
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
End With
' 自动调整选择区域的行高和列宽
With Selection
    .Rows.AutoFit
    .Columns.AutoFit
End With
End Sub

```

'#####

'161.函数作用:自动获取指定月的工作日

'#####

```

Sub 自动填充工作日(month1 As Integer)
    '获取指定月份天数
    Dim days As Integer
    Dim xdate As Date
    xdate = CDate("2008-" + CStr(month1))
    '初始化公共变量 Col2 的值
    col2 = 4
    '调用自定义 Mday()函数获取指定月份的天数
    days = MDay(xdate)
    '循环获取指定月份的工作日
    For i = 1 To days
        '声明变量保存指定日期
        Dim Curdate As String
        Curdate = "2008-" + CStr(month1) + "-" + _
            CStr(i)
        '判断指定日期是否为工作日
        If Weekday(CDate(Curdate)) <> vbSaturday _
            And Weekday(CDate(Curdate)) <> vbSunday Then

```

```
        Cells(2, col2) = i
        col2 = col2 + 1
    End If
Next i
End Sub
```

'获取指定月份的天数

```
Public Function MDay(Optional xdate _
    As Variant = 0) As Integer
    If IsDate(xdate) Then
        MDay = Day(DateSerial(Year(xdate), _
            Month(xdate) + 1, 0))
    Else
        MDay = 0
    End If
End Function
```