

一、存货类型数据透视表

代码块

```
1 Sub CreateInventoryPivot_Fixed_Row1()  
2     Dim wsSource As Worksheet  
3     Dim wsPivot As Worksheet  
4     Dim lastRow As Long, lastCol As Long, r As Long, c As Long  
5     Dim rngSource As Range  
6     Dim pc As PivotCache  
7     Dim pt As PivotTable  
8     Dim valClass As String, catName As String  
9     Dim targetCol As Long  
10  
11     On Error Resume Next  
12     Set wsSource = ThisWorkbook.Sheets("库存")  
13     On Error GoTo 0  
14  
15     If wsSource Is Nothing Then  
16         MsgBox "未找到名为 '库存' 的工作表。", vbCritical  
17         Exit Sub  
18     End If  
19  
20     wsSource.Activate  
21  
22     ' 【关键修改点】您删除了第一行，现在标题在第 1 行了！  
23     Const HeaderRow As Integer = 1  
24  
25     ' 获取最后一行  
26     lastRow = wsSource.Cells(wsSource.Rows.Count, "H").End(xlUp).Row  
27     If lastRow <= HeaderRow Then  
28         MsgBox "数据行数不足，无法处理。", vbExclamation  
29         Exit Sub  
30     End If  
31  
32     ' 自动修复空白标题  
33     lastCol = wsSource.Cells(HeaderRow,  
wsSource.Columns.Count).End(xlToLeft).Column  
34     If lastCol < 29 Then lastCol = 35  
35  
36     For c = 1 To lastCol  
37         If Trim(wsSource.Cells(HeaderRow, c).Value) = "" Then
```

```
38         wsSource.Cells(HeaderRow, c).Value = "空白列_" & c
39         wsSource.Cells(HeaderRow, c).Interior.Color = vbYellow
40     End If
41 Next c
42
43 ' 添加辅助列
44 Application.ScreenUpdating = False
45
46 lastCol = wsSource.Cells(HeaderRow,
wsSource.Columns.Count).End(xlToLeft).Column
47
48 If wsSource.Cells(HeaderRow, lastCol).Value = "VBA自动归类" Then
49     targetCol = lastCol
50 Else
51     targetCol = lastCol + 1
52     wsSource.Cells(HeaderRow, targetCol).Value = "VBA自动归类"
53 End If
54
55 ' 循环归类
56 For r = (HeaderRow + 1) To lastRow
57     valClass = Trim(CStr(wsSource.Cells(r, 8).Value)) ' H列
58     Select Case valClass
59         Case "3100", "3200", "3300", "3400"
60             catName = "原材料及辅料"
61         Case "7900"
62             catName = "半成品"
63         Case "7920"
64             catName = "成品"
65         Case Else
66             catName = "其他"
67     End Select
68     wsSource.Cells(r, targetCol).Value = catName
69 Next r
70
71 ' 定义数据源
72 Set rngSource = wsSource.Range(wsSource.Cells(HeaderRow, 1),
wsSource.Cells(lastRow, targetCol))
73
74 ' 删除旧透视表
75 Application.DisplayAlerts = False
76 On Error Resume Next
77 Sheets("库存透视表").Delete
78 On Error GoTo 0
79 Application.DisplayAlerts = True
80
81 ' 新建透视表
82 Set wsPivot = Worksheets.Add
```

```

83     wsPivot.Name = "库存透视表"
84
85     Set pc = ThisWorkbook.PivotCaches.Create( _
86         SourceType:=xlDatabase, _
87         SourceData:=rngSource)
88
89     Set pt = pc.CreatePivotTable( _
90         TableDestination:=wsPivot.Range("A3"), _
91         TableName:="InventoryPT_" & Format(Now, "hhmmss"))
92
93     With pt
94         On Error Resume Next
95
96         ' 设置行和值
97         .PivotFields("客户").Orientation = xlRowField
98         .PivotFields("客户").Position = 1
99
100        .PivotFields("VBA自动归类").Orientation = xlRowField
101        .PivotFields("VBA自动归类").Position = 2
102
103        .AddDataField .PivotFields("期末库存金额"), "库存金额汇总", xlSum
104
105        On Error GoTo 0
106
107        .RowAxisLayout xlTabularRow
108        .DataBodyRange.NumberFormat = "#,##0.00_ "
109    End With
110
111    Application.ScreenUpdating = True
112    MsgBox "修复完成! 透视表已重新生成。", vbInformation
113 End Sub
114
115

```

二、存货表中存货自动填充

代码块

```

1 Sub UpdateStock_Safe_Row1()
2     Dim wsReport As Worksheet
3     Dim wsSource As Worksheet
4     Dim wsLog As Worksheet
5     Dim lastRowReport As Long, i As Long
6     Dim rawName As String, searchName As String
7     Dim colCustomer As Range, colCategory As Range, colAmount As Range
8     Dim colNumCust As Long, colNumCat As Long, colNumAmt As Long

```

```

9      Dim valFinished As Double, valSemi As Double, valRaw As Double
10     Dim failCount As Long
11     Dim arrFailures() As Variant
12
13     failCount = 0
14
15     '===== 1. 设置工作表 =====
16     On Error Resume Next
17     Set wsReport = ThisWorkbook.Sheets("存货")
18     Set wsSource = ThisWorkbook.Sheets("库存")
19     On Error GoTo 0
20
21     If wsReport Is Nothing Then
22         MsgBox "找不到名为 '存货' 的表格。", vbCritical
23         Exit Sub
24     End If
25     If wsSource Is Nothing Then
26         MsgBox "找不到名为 '库存' 的表格。", vbCritical
27         Exit Sub
28     End If
29
30     '===== 2. 查找列号 (修正：去第1行找) =====
31     ' 【重要修改】因为您删除了原表第一行，现在标题在 Row(1)
32     With wsSource.Rows(1)
33         Set colCustomer = .Find("客户", LookAt:=xlWhole)
34         Set colCategory = .Find("VBA自动归类", LookAt:=xlWhole)
35         Set colAmount = .Find("期末库存金额", LookAt:=xlWhole)
36     End With
37
38     ' 检查是否找到
39     If colCustomer Is Nothing Or colCategory Is Nothing Or colAmount Is
Nothing Then
40         MsgBox "在'库存'表的第1行找不到必要列（客户、VBA自动归类、期末库存金额）。" &
vbCrLf & _
41             "请确认您已经运行了【生成透视表】的那一步，生成了'VBA自动归类'列。",
vbCritical
42         Exit Sub
43     End If
44
45     colNumCust = colCustomer.Column
46     colNumCat = colCategory.Column
47     colNumAmt = colAmount.Column
48
49     '===== 3. 循环填数 =====
50     Application.ScreenUpdating = False
51
52     lastRowReport = wsReport.Cells(wsReport.Rows.Count, "B").End(xlUp).Row

```

```

53
54     For i = 7 To lastRowReport
55         rawName = Trim(wsReport.Cells(i, 2).Value)
56
57         If rawName <> "" And InStr(rawName, "合计") = 0 Then
58
59             ' 清洗名称
60             If InStr(rawName, "(") > 0 Then
61                 searchName = Trim(Left(rawName, InStr(rawName, "(") - 1))
62             ElseIf InStr(rawName, " (") > 0 Then
63                 searchName = Trim(Left(rawName, InStr(rawName, " (") - 1))
64             Else
65                 searchName = rawName
66             End If
67
68             ' 计算
69             valFinished =
Application.WorksheetFunction.SumIfs(wsSource.Columns(colNumAmt),
wsSource.Columns(colNumCust), searchName, wsSource.Columns(colNumCat), "成品")
70             valSemi =
Application.WorksheetFunction.SumIfs(wsSource.Columns(colNumAmt),
wsSource.Columns(colNumCust), searchName, wsSource.Columns(colNumCat), "半成品")
71             valRaw =
Application.WorksheetFunction.SumIfs(wsSource.Columns(colNumAmt),
wsSource.Columns(colNumCust), searchName, wsSource.Columns(colNumCat), "原材料及
辅料")
72
73             ' 写入 (带错误保护)
74             On Error Resume Next
75             wsReport.Cells(i, 3).Value = valFinished
76             wsReport.Cells(i, 4).Value = valSemi
77             wsReport.Cells(i, 5).Value = valRaw
78
79             If Err.Number <> 0 Then
80                 MsgBox "写入出错! 请检查第 " & i & " 行是否被保护.", vbCritical
81                 Application.ScreenUpdating = True
82                 Exit Sub
83             End If
84             On Error GoTo 0
85
86             ' 记录未匹配
87             If (valFinished + valSemi + valRaw) = 0 Then
88                 failCount = failCount + 1
89                 ReDim Preserve arrFailures(1 To 3, 1 To failCount)
90                 arrFailures(1, failCount) = rawName
91                 arrFailures(2, failCount) = searchName
92                 arrFailures(3, failCount) = "未找到数据"

```

```

93         End If
94     End If
95 Next i
96
97 '===== 4. 结果反馈 =====
98 If failCount > 0 Then
99     Set wsLog = Worksheets.Add
100    wsLog.Name = "未匹配清单_" & Format(Now, "hhmm")
101    wsLog.Range("A1:C1").Value = Array("原名", "搜索名", "原因")
102    wsLog.Range("A2").Resize(failCount, 3).Value =
Application.Transpose(arrFailures)
103    MsgBox "更新完成, 有 " & failCount & " 个未匹配项.", vbExclamation
104 Else
105     MsgBox "更新完成, 所有数据匹配成功!", vbInformation
106 End If
107
108 wsReport.Activate
109 Application.ScreenUpdating = True
110 End Sub
111
112

```

三、存货表中发出商品自动填充

代码块

```

1  Sub FillIssuedCost_Final()
2      Dim wsReport As Worksheet
3      Dim wsSource As Worksheet
4      Dim lastRowReport As Long, i As Long
5      Dim cleanName As String, rawName As String
6
7      Dim colSourceCust As Range, colSourceCost As Range
8      Dim colTargetCost As Range
9      Dim colNumSourceCust As Long, colNumSourceCost As Long, colNumTargetCost As
Long
10
11
12     On Error Resume Next
13     Set wsReport = ThisWorkbook.Sheets("存货")
14     Set wsSource = ThisWorkbook.Sheets("发出商品")
15     On Error GoTo 0
16
17     If wsReport Is Nothing Then
18         MsgBox "未找到名为 '存货' 的工作表, 请检查名称.", vbCritical
19         Exit Sub

```

```

20 End If
21 If wsSource Is Nothing Then
22     MsgBox "未找到名为 '发出商品' 的工作表，请检查名称。", vbCritical
23     Exit Sub
24 End If
25
26 With wsSource.Rows(1)
27     Set colSourceCust = .Find("客户简称", LookAt:=xlWhole)
28     Set colSourceCost = .Find("本月发出商品成本", LookAt:=xlWhole)
29 End With
30
31 Set colTargetCost = wsReport.Range("A1:Z6").Find("本月发出商品成本",
LookAt:=xlPart)
32
33 If colSourceCust Is Nothing Then
34     MsgBox "在【发出商品】表中找不到 '客户简称' 列。", vbCritical
35     Exit Sub
36 End If
37
38 If colSourceCost Is Nothing Then
39     MsgBox "在【发出商品】表中找不到 '本月发出商品成本' 列。", vbCritical
40     Exit Sub
41 End If
42
43 If colTargetCost Is Nothing Then
44     MsgBox "在【存货】表中找不到 '本月发出商品成本' 列。", vbCritical
45     Exit Sub
46 End If
47
48 colNumSourceCust = colSourceCust.Column
49 colNumSourceCost = colSourceCost.Column
50 colNumTargetCost = colTargetCost.Column
51
52 Application.ScreenUpdating = False
53
54 lastRowReport = wsReport.Cells(wsReport.Rows.Count, "B").End(xlUp).Row
55
56 For i = 7 To lastRowReport
57     rawName = Trim(wsReport.Cells(i, 2).Value)
58
59     If rawName <> "" And InStr(rawName, "合计") = 0 Then
60
61         If InStr(rawName, "(") > 0 Then
62             cleanName = Trim(Left(rawName, InStr(rawName, "(") - 1))
63         ElseIf InStr(rawName, " (") > 0 Then
64             cleanName = Trim(Left(rawName, InStr(rawName, " (") - 1))
65         Else

```

```

66         cleanName = rawName
67     End If
68
69     wsReport.Cells(i, colNumTargetCost).Value =
Application.WorksheetFunction.SumIfs( _
70         wsSource.Columns(colNumSourceCost), _
71         wsSource.Columns(colNumSourceCust), cleanName)
72
73     End If
74 Next i
75
76 Application.ScreenUpdating = True
77 MsgBox "更新完毕！数据已从【发出商品】同步到【存货】表。", vbInformation
78 End Sub

```

四、应收账款表中应收账款自动填充

代码块

```

1 Sub UpdateAR_FullColumns_WithBUFix()
2     Dim wsTarget As Worksheet
3     Dim wsSource As Worksheet
4     Dim lastRowTarget As Long
5     Dim i As Long
6
7     ' 变量
8     Dim custCode As String
9     Dim targetBU As String, searchBU As String
10
11     ' 强制锁定源表【匹配列】
12     Dim rngSrcBU As Range      ' A列: BU
13     Dim rngSrcCode As Range   ' E列: 客户代码
14
15     ' 自动查找源表【数据列】
16     Dim colSrc_Total As Long, colSrc_Current As Long
17     Dim colSrc_0_1 As Long, colSrc_1_2 As Long, colSrc_2_6 As Long
18     Dim colSrc_6_12 As Long, colSrc_12_24 As Long, colSrc_24Plus As Long
19
20     '===== 1. 设置工作表 =====
21     On Error Resume Next
22     Set wsTarget = ThisWorkbook.Sheets("应收账款")
23     Set wsSource = ThisWorkbook.Sheets("应收账款原始数据")
24     On Error GoTo 0
25
26     If wsTarget Is Nothing Or wsSource Is Nothing Then
27         MsgBox "找不到工作表。", vbCritical

```



```

28         Exit Sub
29     End If
30
31     '===== 2. 锁定与查找列 =====
32     ' 1. 锁定匹配列 (强制 A列 和 E列)
33     Set rngSrcBU = wsSource.Columns("A")
34     Set rngSrcCode = wsSource.Columns("E")
35
36     ' 2. 查找数据列 (在前2行找标题)
37     With wsSource
38         Dim rngHeader As Range
39         Set rngHeader = .Range("A1:Z2")
40
41         colSrc_Total = FindCol(rngHeader, "应收款合计")
42         If colSrc_Total = 0 Then colSrc_Total = FindCol(rngHeader, "应收账款合
计")
43
44         colSrc_Current = FindCol(rngHeader, "账期内")
45         colSrc_0_1 = FindCol(rngHeader, "0-1个") ' 模糊匹配 0-1个月
46         colSrc_1_2 = FindCol(rngHeader, "1-2个")
47         colSrc_2_6 = FindCol(rngHeader, "2-6个")
48         colSrc_6_12 = FindCol(rngHeader, "6-12个")
49         colSrc_12_24 = FindCol(rngHeader, "12-24个")
50         colSrc_24Plus = FindCol(rngHeader, "24个月及以上")
51         If colSrc_24Plus = 0 Then colSrc_24Plus = FindCol(rngHeader, "逾期24个
月")
52     End With
53
54     '===== 3. 循环填数 (全列) =====
55     Application.ScreenUpdating = False
56     lastRowTarget = wsTarget.Cells(wsTarget.Rows.Count, "B").End(xlUp).Row
57
58     For i = 4 To lastRowTarget
59         custCode = Trim(wsTarget.Cells(i, 1).Value) ' A列: 客户代码
60         targetBU = Trim(wsTarget.Cells(i, 2).Value) ' B列: 标准表的BU
61
62         If custCode <> "" And InStr(custCode, "合计") = 0 Then
63
64             ' === BU 智能转换 ===
65             If targetBU = "ATBU2" Then
66                 searchBU = "ATBU"
67             Else
68                 searchBU = targetBU
69             End If
70
71             ' === 开始填数 (F列 到 M列) ===
72

```

```
73         ' F列: 总额
74         If colSrc_Total > 0 Then wsTarget.Cells(i, "F").Value =
Application.WorksheetFunction.SumIfs( _
75             wsSource.Columns(colSrc_Total), rngSrcCode, custCode,
rngSrcBU, searchBU)
76
77         ' G列: 账期内
78         If colSrc_Current > 0 Then wsTarget.Cells(i, "G").Value =
Application.WorksheetFunction.SumIfs( _
79             wsSource.Columns(colSrc_Current), rngSrcCode, custCode,
rngSrcBU, searchBU)
80
81         ' H列: 0-1个月
82         If colSrc_0_1 > 0 Then wsTarget.Cells(i, "H").Value =
Application.WorksheetFunction.SumIfs( _
83             wsSource.Columns(colSrc_0_1), rngSrcCode, custCode, rngSrcBU,
searchBU)
84
85         ' I列: 1-2个月
86         If colSrc_1_2 > 0 Then wsTarget.Cells(i, "I").Value =
Application.WorksheetFunction.SumIfs( _
87             wsSource.Columns(colSrc_1_2), rngSrcCode, custCode, rngSrcBU,
searchBU)
88
89         ' J列: 2-6个月
90         If colSrc_2_6 > 0 Then wsTarget.Cells(i, "J").Value =
Application.WorksheetFunction.SumIfs( _
91             wsSource.Columns(colSrc_2_6), rngSrcCode, custCode, rngSrcBU,
searchBU)
92
93         ' K列: 6-12个月
94         If colSrc_6_12 > 0 Then wsTarget.Cells(i, "K").Value =
Application.WorksheetFunction.SumIfs( _
95             wsSource.Columns(colSrc_6_12), rngSrcCode, custCode, rngSrcBU,
searchBU)
96
97         ' L列: 12-24个月
98         If colSrc_12_24 > 0 Then wsTarget.Cells(i, "L").Value =
Application.WorksheetFunction.SumIfs( _
99             wsSource.Columns(colSrc_12_24), rngSrcCode, custCode,
rngSrcBU, searchBU)
100
101         ' M列: 24个月以上
102         If colSrc_24Plus > 0 Then wsTarget.Cells(i, "M").Value =
Application.WorksheetFunction.SumIfs( _
103             wsSource.Columns(colSrc_24Plus), rngSrcCode, custCode,
rngSrcBU, searchBU)
```

```

104
105         ' 清除颜色
106         wsTarget.Range(wsTarget.Cells(i, 1), wsTarget.Cells(i,
107 17)).Interior.ColorIndex = xlNone
108     End If
109 Next i
110 Application.ScreenUpdating = True
111 MsgBox "更新完成！已填充从总额到24个月以上的所有列。", vbInformation
112 End Sub
113
114 ' 辅助函数
115 Function FindCol(rng As Range, txt As String) As Long
116     Dim c As Range
117     Set c = rng.Find(txt, LookAt:=xlPart)
118     If Not c Is Nothing Then
119         FindCol = c.Column
120     Else
121         FindCol = 0
122     End If
123 End Function

```

五、应收账款表中发出商品填充

代码块

```

1 Sub UpdateIssuedGoods_WithBU_Fix()
2     Dim wsTarget As Worksheet
3     Dim wsSource As Worksheet
4     Dim lastRowTarget As Long
5     Dim i As Long
6
7     ' 变量
8     Dim custCode As String
9     Dim targetBU As String, searchBU As String
10
11     ' 源表列对象
12     Dim rngSrcBU As Range        ' A列: BU
13     Dim rngSrcCode As Range     ' E列: 客户代码
14     Dim rngSrcTotal As Range    ' P列: 发出商品总额
15     Dim rngSrcCurrent As Range  ' Q列: 发出商品账期内
16     Dim rngSrcOverdue As Range  ' R列: 发出商品逾期
17
18     '===== 1. 设置工作表 =====
19     On Error Resume Next
20     Set wsTarget = ThisWorkbook.Sheets("应收账款")

```

```

21 Set wsSource = ThisWorkbook.Sheets("应收账款原始数据")
22 On Error GoTo 0
23
24 If wsTarget Is Nothing Or wsSource Is Nothing Then
25     MsgBox "找不到工作表，请检查名称。", vbCritical
26     Exit Sub
27 End If
28
29 '===== 2. 锁定源表列 (A, E, P, Q, R) =====
30 Set rngSrcBU = wsSource.Columns("A") ' 用于区分 BU
31 Set rngSrcCode = wsSource.Columns("E") ' 用于匹配客户
32
33 ' 根据您之前的指示，发出商品数据在 P, Q, R 列
34 Set rngSrcTotal = wsSource.Columns("P")
35 Set rngSrcCurrent = wsSource.Columns("Q")
36 Set rngSrcOverdue = wsSource.Columns("R")
37
38 '===== 3. 循环填数 (从第5行开始) =====
39 Application.ScreenUpdating = False
40 lastRowTarget = wsTarget.Cells(wsTarget.Rows.Count, "B").End(xlUp).Row
41
42 For i = 5 To lastRowTarget
43     custCode = Trim(wsTarget.Cells(i, 1).Value) ' A列：客户代码
44     targetBU = Trim(wsTarget.Cells(i, 2).Value) ' B列：标准表的BU
45
46     If custCode <> "" And InStr(custCode, "合计") = 0 Then
47
48         ' === BU 智能转换 ===
49         If targetBU = "ATBU2" Then
50             searchBU = "ATBU"
51         Else
52             searchBU = targetBU
53         End If
54
55         ' === 双条件匹配填数 ===
56
57         ' 1. N列 (总额) <- 源表 P列
58         wsTarget.Cells(i, "N").Value =
59 Application.WorksheetFunction.SumIfs( _
60             rngSrcTotal, _
61             rngSrcCode, custCode, _
62             rngSrcBU, searchBU)
63
64         ' 2. O列 (账期内) <- 源表 Q列
65         wsTarget.Cells(i, "O").Value =
66 Application.WorksheetFunction.SumIfs( _

```

```
66         rngSrcCode, custCode, _
67         rngSrcBU, searchBU)
68
69     ' 3. P列 (逾期) <- 源表 R列
70     wsTarget.Cells(i, "P").Value =
Application.WorksheetFunction.SumIfs( _
71         rngSrcOverdue, _
72         rngSrcCode, custCode, _
73         rngSrcBU, searchBU)
74
75     End If
76 Next i
77
78 Application.ScreenUpdating = True
79 MsgBox "发出商品更新完成！" & vbCrLf & _
80     "已加入 BU 区分逻辑，数据不会混淆了。", vbInformation
81 End Sub
```