

Set up User-defined Function Design List Based on Excel

在 Excel 上设置自定义函数定义报表

提要：在用 Excel 管理数据时，如何方便高效地定义报表是一个难题，本文介绍设置自定义函数解决这一问题的方法。

关键词：Excel 自定义函数 报表

刘 曜（重庆邮电学院管理学院 400065）

1 问题的提出

Excel 电子表格形式上是一个“表册”，用以管理数据方便直观，且可利用其强大的功能进行数据的后续处理，但如何方便高效地定义报表是一个难题。下面举例说明。

设在 Excel 的一个“存货管理”工作簿的“库存数量”工作表上有如图 1 所示数据。

假定要在“存货管理”工作簿另一命名为“盘存报表”的工作表上的 D2 单元格，反映代号为 A10201 的商品库存数量。一般情况是在 D2 单元格中输入公式“=库存数量!B4”，或“=库存数量!\$B\$4”。这种通过建立与原始数据所在单元格的联系来获取报表数据的方法，在定义报表时由于原始数据所在单元格地址与报表内容无关，因而容易出错；当某一原始数据所在单元格地址变动时，可能导致其他原始数据所在单元格地址相应变动，从而需要很麻烦地重新定义报表。本文通过设置以数据特征为变量的 Excel 自定义函数来定义报表，解决这一问题。

2 设置以商品品种代号为变量的自定义函数

下面则用前例说明，如何设置以数据特征(商品品种代号)为变量的自

	A	B
1	商品品种代号	库存数量 (件)
2	A10101	1000
3	A10102	230
4	A10201	7500
5	A10202	2340
6	A10203	3710

图 1 “库存数量”工作表数据

定义函数定义报表。在“存货管理”工作簿的模块表上编写如下的 VBA 代码，设置以商品品种代号为变量的自定义函数 KCSL：

Function KCSL (商品品种代号)

x = 1

Do While Not (IsEmpty (Sheets (“库存数量”). Cells (x, 1). Value))

x = x + 1

Loop

以上四行代码，确定“库存数量”工作表第一列从第一行起到第一个空行的行数 x。

Found = False

‘设定 Found 为 False.

For t = 2 To x - 1

If 商品品种代号 = Sheets (“库存数量”). Cells (t, 1) Then

Found = True

Exit For

End If

Next t

以上六行代码，在“库存数量”工作表第一列从第二行起到第 x - 1 行查找函数变量所定义的商品品种代号，如果找到了，Found 就由 False 变为 True；否则，Found 仍为 False。

If Found = True Then

KCSL = Sheets (“库存数量”). Cells (t, 2)

以上两行代码，如果找到了函数变量所定义的商品品种代号 (Found = True)，那么，函数值为由“库存数量”工作表第二列与函数变量所定义的商品品种代号所在行决定的单元格值，即函数变量所定义的商品品种代号

对应的商品库存数量。

Else

KCSL = "商品品种代号错"

'如果未代到 (Found 仍为 False), 那么, 显现"商品品种代号错"字样。

End If

End Function

然后, 在"盘存报表"工作表的 D2 单元格定义以 A10201 为变量值的自定义函数 KCSL, 即输入"=KCSL("A10201")", 回车后, D2 单元格即显现代号为 A10201 的商品库存数量 7500。如果输入的商品品种代号不存在, 会给出"商品品种代号错"的提示。

当然, 也可以在"盘存报表"工作表的 D2 单元格中输入"=KCSL("A10201")+KCSL("A10202")+KCSL("A10203)", 在 D2 反映代号为 A10201、A10202 和 A10203 的商品库存数量之和, 或者自定义函数 KCSL 的其他一些形式, 以反映商品库存数量之间的关系。事实上, 所设置的自定义函数 KCSL 可以在"存货管理"工作簿的任意位置, 以合乎规则的任何形式方便地调用它。

3 两种方法的比较

在本文一开始我们就指出, 通过建立与原始数据所在单元格的联系来获取报表数据的方法存在以下缺陷:

(1) 由于原始数据所在单元格地址与报表内容无关, 这样在定义报表时很容易出错。

(2) 当某一原始数据单元格地址发生变动, 可能导致整个报表需要重新定义, 以建立报表数据与新的原始数据单元格地址之间的关系。

例如在上例中, 在商品品种代号"A10102"与"A10201"之间插入一新的商品品种代号"A10103" (假定库存 200 件), 如图 2 所示。

这样, "A10201"商品后的所有商品库存数量的单元格地址均发生了变化, 因而与之相对应的报表定义均需重新修改。由于商品品种可

	A	B
1	商品品种代号	库存数量 (件)
2	A10101	1000
3	A10102	230
4	A10103	200
5	A10201	7500
6	A10202	2340
7	A10203	3710

图 2 插入"A10103"商品后"库存数量"工作表数据

能随时发生增删, 也就需要相应频繁变动报表定义, 这一工作量是可想而知的。

而现在我们看到, 根据数据特征用自定义函数定义报表的方法完全解决了以上问题。从示例中自定义函数 KCSL 的设计可以看出, 无论商品品种怎样增删, 函数均可自动检索出商品品种代号相应的库存数量, 因而在商品品种代号地址由于增删发生变动时, 无需重新定义报表。并且, 由于函数 KCSL 与数据特征 (商品品种代号) 相联系, 可以按报表有关内容方便准确地定义报表。

4 更新自定义函数值

至此, 似乎本文提出的问题已经得到解决。但有一点应当注意, 自定义函数值不能自动更新。

如将"库存数量"工作表上代号为"A10201"的商品库存数量改为 8000 后, "盘存报表"工作表上定义了函数"=KCSL("A10201")"的 D2 单元格的数值却不能以新的数值替代, 仍反映为原来的 7500 而不是 8000。又如何解决这一新的问题呢?

可以在"盘存报表"工作表的 D2 单元格中重新输入函数"=KCSL("A10201")", 回车后, D2 单元格即显现新的数据 8000。但是这一解决办法, 当报表上定义的自定义函数比较多时, 如此一一更新显然是很麻烦的。一个好的解决办法是在"存货管理"工作簿的模块表上设置一个下面的宏, 在定义了自定义函数的报表工作表 (例中为"盘存报表"工作表) 为当前工作表时, 只要执行它, 报表上所有的自定义函数值都会立即更新。

```
Sub 更新自定义函数值()
```

```
Columns("IS:IS").Select
```

```
Selection.Cut
```

```
Columns("IT:IT").Select
```

```
ActiveSheet.Paste
```

```
Selection.Copy
```

```
Columns("IS:IS").Select
```

```
ActiveSheet.Paste
```

```
Application.CutCopyMode = False
```

```
Range("IS5").Select
```

```
End Sub
```

宏代码中的操作是对当前工作表最右方的 IS 和 IT 列进行, 其目的是避开工作表上的数据, 以免破坏它们。如果不担心的话, 可以选择其他列达到同样的效果。 ■