

## 18) VBA code used in the extraction of the material combination pattern

- \* These codes were designed with and are run using Microsoft Excel.
- \* The data-sets imported from the relational database for this operation are:

Column A: Eb.no  
Column B: Total number of ingredient: (a) >2, (b) >3  
Column C: Material ID  
Column D: Material name

### (a) Combination pattern between 2 materials

```
' define variables

Dim store(100) As String
Dim storeID(100) As Integer
Dim takenum As Integer
Dim CTF As Integer
Dim CTS As Integer
Dim prepointer As Integer
Dim pastpointer As Long
Dim posiF As Long
Dim posiS As Long
Dim i As Integer
Dim j As Integer

' set variables

j = 1
prepointer = 1
pastpointer = 1

10:

' take values of total number

takenum = Cells(prepointer, 2).Value

For i = 1 To takenum
    storeID(i) = Cells((prepointer - 1) + i, 3)
    store(i) = Cells((prepointer - 1) + i, 4)
Next i

' copy past cells

CTF = 1
CTS = 2

20:

Cells(pastpointer, 5) = storeID(CTF)
Cells(pastpointer, 6) = storeID(CTS)
Cells(pastpointer, 7) = store(CTF)
Cells(pastpointer, 8) = store(CTS)

CTS = CTS + 1

If CTS > takenum Then CTF = CTF + 1: CTS = CTF + 1
If CTF > takenum - 1 Then GoTo 30 Else pastpointer = pastpointer + 1: GoTo 20

' set prepointer
```

30:

```
    pastpointer = pastpointer + 1
    prepointer = prepointer + takenum
' judge stop point
    If Cells(prepointer, 1).Value > 1 Then GoTo 10
' swap position and combine words
Do While Cells(j, 5) <> ""
    posiF = Cells(j, 5).Value
    posiS = Cells(j, 6).Value
    If posiF < posiS Then Cells(j, 9) = posiF: Cells(j, 10) = posiS: Cells(j, 11) = Cells(j, 7): Cells(j, 12) =
        Cells(j, 8) Else Cells(j, 9) = posiS: Cells(j, 10) = posiF: Cells(j, 11) = Cells(j, 8): Cells(j, 12) =
            Cells(j, 7)
    Cells(j, 13) = CStr(Cells(j, 11)) + " : " + CStr(Cells(j, 12))
    j = j + 1
Loop
' delete A-L & sort A
Columns("A:L").Select
Selection.Delete Shift:=xlToLeft
Columns("A:A").Select
Selection.Sort Key1:=Range("A1"), Order1:=xlAscending, Header:=xlGuess, _
OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom
' subtotal
Rows("1:1").Select
Selection.Insert Shift:=xlDown
Range("A1").Select
ActiveCell.FormulaR1C1 = "Count"
Range("A1").Select
Selection.Subtotal GroupBy:=1, Function:=xlCount, TotalList:=Array(1), _
Replace:=True, PageBreaks:=False, SummaryBelowData:=False
ActiveWindow.SmallScroll Down:=-6
Columns("A:A").ColumnWidth = 20.43
ActiveSheet.Outline.ShowLevels RowLevels:=2
```

(b) Code for combination pattern between 3 materials

```
' define variables
Dim store(100) As String
Dim storeID(100) As Integer
Dim takenum As Integer
Dim CTF As Integer
Dim CTS As Integer
Dim CTT As Integer
Dim CTSB As Integer
Dim CTTB As Integer
Dim prepointer As Integer
Dim pastpointer As Long
```

```

Dim posiF As Long
Dim posiS As Long
Dim posiT As Long
Dim posiFM As String
Dim posiSM As String
Dim posiTM As String
Dim I As Integer
Dim j As Integer

```

```

prepointer = 1
pastpointer = 1

```

10:

' take value of total number

```

takenum = Cells(prepointer, 2).Value

```

```

For I = 1 To takenum
    storeID(I) = Cells((prepointer - 1) + I, 3)
    store(I) = Cells((prepointer - 1) + I, 4)
Next I

```

' copy past cells

```

CTF = 1
CTS = 2
CTT = 3
CTSB = 2
CTTB = 3

```

20:

```

Cells(pastpointer, 5) = storeID(CTF)
Cells(pastpointer, 6) = storeID(CTS)
Cells(pastpointer, 7) = storeID(CTT)
Cells(pastpointer, 8) = store(CTF)
Cells(pastpointer, 9) = store(CTS)
Cells(pastpointer, 10) = store(CTT)

```

```

CTT = CTT + 1

```

```

If CTT > takenum Then CTS = CTS + 1: CTTB = CTTB + 1: CTT = CTTB
If CTS > takenum - 1 Then CTF = CTF + 1: CTSB = CTSB + 1: CTS = CTSB: CTTB = CTS + 1: CTT
    = CTTB
If CTF > takenum - 2 Then GoTo 30 Else pastpointer = pastpointer + 1: GoTo 20

```

' set prepointer

30:

```

pastpointer = pastpointer + 1
prepointer = prepointer + takenum

```

' judge stop point

```

If Cells(prepointer, 1).Value > 1 Then GoTo 10

```

' swap position

```

For I = 1 To 2
    j = 1

```

```

Do While Cells(j, 5) <> ""

```

```

    posiS = Cells(j, 6).Value: posiSM = Cells(j, 9)

```

```

    posiT = Cells(j, 7).Value: posiTM = Cells(j, 10)
    If posiS < posiT Then Else Cells(j, 6) = posiT: Cells(j, 7) = posiS: Cells(j, 9) = posiTM: Cells(j,
        10) = posiSM
    j = j + 1
Loop
j = 1
Do While Cells(j, 5) <> ""
    posiF = Cells(j, 5).Value: posiFM = Cells(j, 8)
    posiS = Cells(j, 6).Value: posiSM = Cells(j, 9)
    If posiF < posiS Then Else Cells(j, 5) = posiS: Cells(j, 6) = posiF: Cells(j, 8) = posiSM: Cells(j,
        9) = posiFM
    j = j + 1
Loop
Next I
' combine words
j = 1
Do While Cells(j, 5) <> ""
    Cells(j, 11) = CStr(Cells(j, 8)) + " : " + CStr(Cells(j, 9)) + " : " + CStr(Cells(j, 10))
    j = j + 1
Loop
' delete A-J & sort A
Columns("A:J").Select
Selection.Delete Shift:=xlToLeft
Columns("A:A").Select
Selection.Sort Key1:=Range("A1"), Order1:=xlAscending, Header:=xlGuess, _
    OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom
' subtotal
Rows("1:1").Select
Selection.Insert Shift:=xlDown
Range("A1").Select
ActiveCell.FormulaR1C1 = "Count"
Range("A1").Select

Selection.Subtotal GroupBy:=1, Function:=xlCount, TotalList:=Array(1), _
    Replace:=True, PageBreaks:=False, SummaryBelowData:=False
ActiveWindow.SmallScroll Down:=-6
Columns("A:A").ColumnWidth = 20.43
ActiveSheet.Outline.ShowLevels RowLevels:=2

```