

## Visual Basic for Application

Từ Office 95 Microsoft bắt đầu đưa vào một ngôn ngữ lập trình chung, không chỉ áp dụng cho các ứng dụng như Word và Excel mà cho cả các công cụ lập trình như VB. Microsoft gọi ngôn ngữ lập trình này là Visual Basic for Applications (VBA).

### VBA for Excel

Microsoft Excel là một công cụ mạnh để phân tích và trình bày thông tin. Một trong những thế mạnh của Excel là ngôn ngữ macro. Kể từ khi ra đời, Excel luôn là chương trình bảng tính có cài ngôn ngữ macro bao quát và linh hoạt nhất trong các phần mềm bảng tính.

Để viết macros trong Excel, bạn thật sự cần học hai công cụ khác biệt. Trước hết, bạn cần học cách làm việc với Visual Basic. Mọi kiến thức bạn học về Visual Basic không những áp dụng được trong Excel mà còn cho các phần mềm Microsoft khác. Thứ hai bạn học cách sử dụng Excel. Càng nắm vững sử dụng bảng tính Excel, bạn càng xây dựng hiệu quả các macros.

### Bài 1: Chương trình VBA đầu tiên - Ra quyết định sử dụng VBA for Excel

Giả thiết bạn có bảng doanh số bán hàng như hình 1. Bạn muốn tìm xem những mặt hàng nào bán ra ít hơn 100 bộ và tô màu đỏ. Chúng ta có thể sử dụng Format có điều kiện để làm việc này. Tuy nhiên để áp dụng linh hoạt, chúng ta sẽ xây dựng một macro làm việc đó.

	Doanh số bán hàng của quý I và quý II						
		Tháng 1	Tháng 2	Tháng 3	Tháng 4	Tháng 5	Tháng 6
áo phong thể thao		30	120	115	122	98	110
Bộ đồ thể thao		220	180	160	140	120	100
Bộ áo mưa		93	90	101	105	200	220
Quần soóc		125	133	199	83	104	93

Hình 1

### Xây dựng nội dung VBA sử dụng Record Macro

Excel cung cấp một công cụ tiện lợi để thu một Macro.

- Đứng ở một ô cần định dạng
- Trên thanh công cụ toolbar, chọn Tools, Macro, chọn Record New Macro.
- Hộp hội thoại Record New Macro xuất hiện. Trong hộp Macro Name, nhập tên

```
Sub MakeDecision()  
' MakeDecision Macro  
' Macro recorded 2/15/00 by Phu  
'
```

- của Macro cần thu, ví dụ MakeDecision và Selection.Font.ColorIndex = 3  
nhấn nút OK. ActiveCell.Offset(1,
4. Nút Record Macro xuất hiện như trong hình bên cho thấy bộ thu recorder đang hoạt động. Đảm bảo nút Relative Reference được nhấn lõm xuống. 0).Range("A1").Select  
End Sub
  5. Định dạng ô (tô chữ màu đỏ), sau đó nhấn mũi tên xuống dưới một lần
  6. Nhấn nút Stop Macro
  7. Lên Tools, Macro, Macros, một hộp thoại Macros xuất hiện. Chọn MakeDecision dưới mục Macro Name, sau đó nhấn nút Edit. Như vậy chúng ta chuyển sang môi trường lập trình của Excel, đồng thời chuyển tới macro vừa thu.

Đối tượng ActiveCell tham chiếu đến ô hiện hành. Phương thức Offset của đối tượng ActiveCell chọn vùng nằm lệch đi so với ô hiện hành. Trong macro trên chọn ô nằm dưới vùng đang chọn một hàng và trên cùng cột.

### Chạy macro

1. Chọn Sheet1 nếu nó chưa được chọn
2. Đứng ở ô dữ liệu đầu tiên (giao của hàng áo phong thể thao với Tháng 1). Trên Tools menu, chọn Macro, Macros, một hộp thoại Macro xuất hiện.
3. Chọn macro MakeDecision từ danh sách và nhấn OK.

Để tiện thử macro trong khi viết có thể gán nó với một nút

### Gán macro với một nút

1. Nếu chưa xuất hiện thanh công cụ Form, hiện nó lên.
2. Trên thanh công cụ Form chọn nút Button rồi vẽ một nút lên bảng tính (nhấn giữ chuột trái trong khi vẽ một hình chữ nhật)
3. Nhả chuột trái, một hộp thoại Assign Macro xuất hiện, chọn MakeDecision từ danh sách, sau đó nhấn nút OK
4. Một nút xuất hiện trên bảng tính, mỗi khi nhấn chuột lên nút thì macro MakeDecision sẽ được thực hiện.

Chú ý: để sửa thuyết minh trên nút, bạn có thể nhấn Ctrl và nhấp lên nút, khi đó nút sẽ được chọn, bạn có thể nhập thuyết minh mới.

### Định dạng cho toàn cột

Macro trên chỉ định dạng cho một ô. Để định dạng toàn bộ cột, cần thêm hai dòng lệnh ở đầu và cuối thủ tục như sau:

```
Sub MakeDecision()
' MakeDecision Macro
' Macro recorded 2/15/00 by Phu
Do Until ActiveCell = ""
Selection.Font.ColorIndex = 3
ActiveCell.Offset(1, 0).Range("A1").Select
```

Loop

End Sub

Để dừng vòng lặp điều kiện Until ActiveCell = "" được đưa vào câu lệnh Do-Loop. Macro sẽ duyệt toàn bộ các ô trong cột và định dạng cho đến khi gặp ô trống.

### Để macro ra quyết định

Chúng ta cần phải để macro tự ra quyết định nhằm tự động hoá việc định dạng. Tuy nhiên, bộ ghi macro không thể đưa các quyết định vào macro. Chúng ta phải tự làm lấy việc đó. Bạn muốn macro tự quyết định xem có cần phải tô đỏ ô hiện hành hay không.

1. Quay sang VBA-Editor, chọn macro (thủ tục) MakeDecision, nhập câu lệnh If ActiveCell > 100 Then để quyết định xem ô nào có giá trị nhỏ hơn.
2. Nhập End If để kết thúc lệnh If. Macro sẽ như sau:

```
Sub MakeDecision()
```

```
,
```

```
' MakeDecision Macro
```

```
' Macro recorded 2/15/00 by Phu
```

```
Do Until ActiveCell = ""
```

```
If ActiveCell < 100 Then
```

```
    Selection.Font.ColorIndex = 3
```

```
End If
```

```
ActiveCell.Offset(1, 0).Range("A1").Select
```

```
Loop
```

```
End Sub
```

### Để quyết định hoạt động trên toàn bảng

Bạn cần phải để macro chuyển tới cột tiếp theo để chạy khắp bảng tính. Thêm các câu lệnh để chuyển đến đỉnh của cột tiếp theo.

```
Sub MakeDecision()
```

```
,
```

```
' MakeDecision Macro
```

```
' Macro recorded 2/15/00 by Phu
```

```
Do Until ActiveCell = ""
```

```
If ActiveCell < 100 Then
```

```
    Selection.Font.ColorIndex = 3
```

```
End If
```

```
ActiveCell.Offset(1, 0).Range("A1").Select
```

```
Loop
```

```
ActiveCell.Offset(-1, 1).Range("A1").Select
```

```
Selection.End(xlUp).Select
```

```
ActiveCell.Offset(1, 0).Range("A1").Select
```

```
End Sub
```

1. Dòng lệnh mới thêm thứ nhất để Excel lên trên một ô sau đó sang phải một ô.
2. Dòng thứ hai để Excel lên đỉnh cột.
3. Dòng cuối cùng để Excel đang từ đỉnh cột, nghĩa là vùng tiêu đề cột, xuống dưới một ô, tức vùng dữ liệu, như vậy là vùng tiêu đề không bị đưa vào tô màu.

Cuối cùng thêm một câu lệnh lặp Do Loop khác để macro có thể duyệt hết từ cột này sang cột khác cho đến khi gặp phải ô trống.

Phiên bản visual basic macro cuối cùng sẽ như sau:

```
Sub MakeDecision()
'
' MakeDecision Macro
' Macro recorded 2/15/00 by Phu
Do Until ActiveCell = ""
  Do Until ActiveCell = ""
    If ActiveCell < 100 Then
      Selection.Font.ColorIndex = 3
    End If
    ActiveCell.Offset(1, 0).Range("A1").Select
  Loop
  ActiveCell.Offset(-1, 1).Range("A1").Select
  Selection.End(xlUp).Select
  ActiveCell.Offset(1, 0).Range("A1").Select
Loop
End Sub
```

### Chạy macro để định dạng toàn bảng tính

1. Chọn Sheet1 nếu nó chưa được chọn
2. Đứng ở ô dữ liệu đầu tiên (giao của hàng áo phong thể thao với Tháng 1). Trên Tools menu, chọn Macro, Macros, một hộp hội thoại Macro xuất hiện.
3. Chọn macro MakeDecision từ danh sách và nhấn OK.
4. Toàn bộ bảng tính sẽ được duyệt và những doanh số dưới 100 sẽ bị tô đỏ.

## Bài 2: VBA for Excel - tính tổng lũy tiến

### Bài toán

Tính cột tổng lũy tiến và tính lũy tiến tại ô

Excel cung cấp một số phương pháp để tính tổng lũy tiến. Bạn có thể sử dụng hàm =SUM(\$A\$1:A1) trong đó trên cột A bạn nhập các giá trị và nhập công thức tính tổng lũy tiến tại ô đầu tiên trên cột B sau đó copy công thức xuống các ô bên dưới.

Để tính tổng lũy tiến ngay tại một ô, chúng ta có thể sử dụng một macro VBA để lưu một tổng dồn dưới dạng phần chú giải (note) của một ô, khi nhập một số vào trong ô thì giá trị của ô sẽ công thêm với giá trị trong phần chú giải của ô đó.

### Tính cột tổng lũy tiến

A	B
Values	Cumulative Total
2.00	2.00
43.00	45.00
5.00	50.00
43.00	93.00
65.00	158.00
45.00	203.00

Tạo ô cộng lũy tiến

Lên tool, Macro, VBA-Editor để sang môi trường lập trình của Excel

Trong môi trường lập trình, lên menu insert, chọn module

Nhập các chương trình sau

- Trên phần General Declaration đưa vào dòng lệnh  
Option Explicit 'Nghĩa là việc khai báo các biến là bắt buộc
- Thủ tục Auto\_Open  
Auto\_Open là một tên đặc biệt trong Visual Basic, mỗi lần mở workbook, Auto\_Open sẽ tự chạy. Trong ví dụ này Excel sẽ gọi thủ tục "CumTotal" mỗi khi bạn mở sổ bảng tính.  
Sub Auto\_Open()  
    Application.OnEntry = "CumulativeTotal"  
End Sub
- Thủ tục gán tổng lũy tiến AssignCumulativeTotal  
Khi bạn chọn một ô và chạy macro này, Excel sẽ gán trường ghi chú giá trị CumTotal\_ với số trong ô hiện hành. Ví dụ nếu nhập số 12 thì trường ghi chú sẽ là "CumTotal\_12"  
Sub AssignCumulativeTotal()  
If Application.IsNumber(ActiveCell) Then  
    ActiveCell.NoteText Text:="CumTotal\_" & ActiveCell  
Else  
    ActiveCell.NoteText Text:="CumTotal\_" & "0"  
End If  
End Sub
- Thủ tục cộng dồn CumulativeTotal  
Macro Auto\_Open đặt Excel sẽ gọi thủ tục này mỗi khi bạn nhập liệu vào bảng tính.  
Sub CumulativeTotal()  
'Câu lệnh If để kiểm tra xem ô mà bạn vừa  
' nhập liệu vào (Application.Caller) có trường ghi chú bắt đầu  
' với "CumTotal\_" hay không  
If Application.Caller.NoteText(length:=9) = "CumTotal\_" Then  
    'Nếu trường note chứa tổng, câu lệnh sau sẽ cộng tổng này với 'số vừa  
    nhập vào ô.

```

Application.Caller.Value = Application.Caller.Value + _
Application.Caller.NoteText(start:=10)
'Câu lệnh sau lưu tổng cộng dồn mới vào trường note.
Application.Caller.NoteText "CumTotal_" & Application.Caller.Value
End If
End Sub
5. Thủ tục xoá chức năng cộng lũy tiến ra khỏi ô hiện hành CancelCumulativeTotal
Sub CancelCumulativeTotal()
ActiveCell.ClearNotes
End Sub
6. Thủ tục đưa tổng lũy tiến về không ResetCumulativeTotal
Sub ResetCumulativeTotal()
ActiveCell.NoteText Text:="CumTotal_" + "0"
ActiveCell.Value = 0
End Sub

```

### Bài 3: Hàm tự tạo - Hàm cắt tên

```

Function CatTen(HoVaTen As String) As String
Dim l, i As Integer
HoVaTen = Trim(HoVaTen)
l = Len(HoVaTen)
CatTen = ""
For i = l To 1 Step -1
If Mid(HoVaTen, i, 1) = Space(1) Then
k = i
Exit For
End If
Next i
CatTen = Mid(HoVaTen, i + 1)
End Function

```

### Bài 4: Hàm tự tạo - Đổi số ra chữ

```

Function SoToChu(So)
If So = 0 Then
Resp = "Không Đồng"
Else
If Abs(So) > 999999999999.99 Then
Resp = "Số quá lớn"
Else
GoSub ThucHien:
End If
End If
SoToChu = UCase(Left(Resp, 1)) + Mid(Resp, 2)
Exit Function

```

```

ThucHien:
If So < 0 Then
Resp = "Trừ "
Else
Resp = Space(0)
End If
Tien = Format(Abs(So), "#####0.00")
Tien = Right(Space(12) + Tien, 15)
Doc = Space(0): Dem = Doc
Doc = Doc + "trăm mười tỷ "
Doc = Doc + "trăm mười triệu "
Doc = Doc + "trăm mười ngàn "
Doc = Doc + "trăm mười đồng "
Doc = Doc + "trăm mười xu "
Dem = Dem + "một hai ba bốn năm sáu bảy tám chín "

```

```

For i = 1 To 5
Nhom = Mid(Tien, i * 3 - 2, 3)
If Nhom <> Space(3) Then
Select Case Nhom
Case "000"
If i = 4 Then
Chu = "đồng "
Else
Chu = Space(0)
End If
Case ".00"
Chu = "chẵn "
Case Else
So1 = Left(Nhom, 1)
So2 = Mid(Nhom, 2, 1)
So3 = Right(Nhom, 1)
Chu = Space(0)

```

```

For J = 1 To 3
Dich = Space(0)
S = Val(Mid(Nhom, J, 1))
If S > 0 Then

```

```

Dich = Trim(Mid(Dem, S * 5 - 4, 5)) + " "

```

```
If J = 2 Then DichCu = Dich
If J = 3 And Dich = "bốn " And DichCu <> "một " Then Dich = "tư "
```

```
'MsgBox Dich
```

```
Dich = Dich + Trim(Mid(Doc, (i - 1) * 18 + J * 6 - 5, 6)) + " "
```

```
End If
```

```
Select Case J
```

```
Case 2 And S = 1
```

```
Dich = "mười "
```

```
Case 3 And S = 0
```

```
Dich = Trim(Mid(Doc, (i - 1) * 18 + J * 6 - 5, 6)) + " "
```

```
Case 3 And S = 5 And So2 <> Space(1) And So2 <> "0"
```

```
Dich = "l" + Mid(Dich, 2)
```

```
Case 2 And S = 0 And So3 <> "0"
```

```
If (So1 >= "1" And So1 <= "9") Or (So1 = "0" And i = 4) Then
```

```
Dich = "lẻ "
```

```
End If
```

```
End Select
```

```
Chu = Chu + Dich
```

```
Next J
```

```
End Select
```

```
ViTri = InStr(1, Chu, "mười một", 1)
```

```
If ViTri > 0 Then Mid(Chu, ViTri, 9) = "mười một"
```

```
If Chu <> "đồng " And Chu <> "chẵn " And Resp <> Space(0) Then
```

```
Resp = Resp + ", " + Chu
```

```
Else
```

```
Resp = Resp + Chu
```

```
End If
```

```
End If
```

```
Next i
```

```
Return
```

```
End Function
```

## Bài 5 Ví dụ về hàm tự tạo

	A	B	C	D	E	F	G
1	123	-62	-39	102	98	-50	?
2	18589	-34716	-28310	-10065	-33145	20588	??
3							



4							
5	2	-50	-16	-96	-57	-8	?
6	19161	38990	5297	-29889	19228	-16365	??
7							
8							
9	2	3	10	-54	60	75	?
10	-4395	12541	162	-25270	-23856	16377	??
11							
12							
13	200	-43	-57	-16	-63	-29	?
14	38202	22914	16268	-22760	20385	-22947	??

### Yêu Cầu

? bằng tổng của giá trị ngang hàng nằm ở cột đầu với giá trị lớn nhất của cột B+C với các ô trong cột còn lại D, E và F, lấy giá trị tìm được nếu dương, còn không để trống

?? Cột A cộng với giá trị tương ứng với giá trị lớn nhất tìm thấy ở trên, trường hợp giá trị lớn nhất là tổng B+C thì lấy tổng tương ứng của B+C, nếu tổng của ? là dương, còn không để trống

Function MySumIf1(Vung As Range)

Dim SoHang, SoCot, i As Byte

SoCot = Vung.Columns.Count

SoHang = Vung.Rows.Count

If SoCot < 4 Or SoHang < 2 Then

MySumIf = "The range must contain at least two rows and four columns"

Exit Function

End If

Dim Max As Double

Max = Vung.Cells(1, 2) + Vung.Cells(1, 3)

For i = 4 To SoCot

If Abs(Vung.Cells(1, i)) > Abs(Max) Then Max = Vung.Cells(1, i)

Next i

MySumIf1 = If(Vung.Cells(1, 1) + Max > 0, Vung.Cells(1, 1) + Max, "")

End Function

Function MySumIf2(Vung As Range)

Dim SoHang, SoCot, i As Byte

SoCot = Vung.Columns.Count

SoHang = Vung.Rows.Count

If SoCot < 4 Or SoHang < 2 Then

MySumIf = "The range must contain at least two rows and four columns"

Exit Function

End If

Dim Max, Tong As Double

Max = Vung.Cells(1, 2) + Vung.Cells(1, 3)

```
Tong = Vung.Cells(2, 2) + Vung.Cells(2, 3)
For i = 4 To SoCot
    If Abs(Vung.Cells(1, i)) > Abs(Max) Then Tong = Vung.Cells(2, i)
Next i
Tong = Vung.Cells(2, 1) + Tong
MySumIf2 = If(IsNumeric(Vung.Cells(1, SoCot + 1)), Tong, "")
End Function
```

---