



Pemrograman Komputer

oleh

Djoko Luknanto

Departemen Teknik Sipil dan Lingkungan

Fakultas Teknik Universitas Gadjah Mada

Pemrosesan sekelompok bilangan

- Dalam pemrograman komputer, sering dijumpai pemrosesan sekelompok bilangan; tidak hanya satu atau dua bilangan.
- Misal: menjumlahkan, menghitung minimum dan maksimum sekelompok bilangan.
- Dalam Microsoft Excel sekelompok bilangan dapat diacu sebagai “Range”

Cara Penggunaan Range (1/2)

Sekelompok bilangan sebagai range (A2:D4)

	B	C	D	
	Data input berupa "Range"			
2	12,40	34,50	87,90	80,00
3	1,50	23,50	8,00	-89,60
4	45,00	67,80	0,00	7,00
5				

Range (A2:D4) digunakan sebagai input

Hasil dengan Excel Macro

		= SUM (A2:D4)	278,00
--	--	---------------	--------

Hasil dengan VBA-Excel Function

9			
10		= JumlahBilangan (A2:D4)	278,00
11		= JumlahCerdas (A2:D4)	278,00

Cara Penggunaan Range (2/2)

Sekelompok bilangan sebagai range (A2:D4)

Bolehkan input nir-bilangan digunakan?

	A	B	C	D	E
1	Data input berupa "Range"				
2	12,40	34,50	87,90	80,00	
3	A	23,50 q		-89,60	
4	45,00	67,80 a,g	Luk		
5					
6	Minimum =	-89,6		=Minimum(A2:D4)	
7	Maksimum =	87,9		=Maksimum(A2:D4)	
8	Perkalian =	-19.325.810.728.949,80		=Perkalian(A2:D4)	
9	Product =	-19.325.810.728.949,80		=PRODUCT(A2:D4)	

Range (A2:D4) digunakan sebagai input

VBA-Excel: JumlahBilangan (A2:D4)

Range (A2:D4)
sebagai input diberi
nama Data

```
Function JumlahBilangan(Data As Range) As Double  
' Menghitung jumlah range berupa bilangan saja
```

```
Dim Jumlah As Double
```

```
Jumlah = 0#
```

```
For Each Item In Data
```

```
    Jumlah = Jumlah + Item.Value
```

```
Next Item
```

```
JumlahBilangan = Jumlah
```

```
End Function
```

Data mempunyai
anggota banyak
disebut Item

Cara
memproses
setiap anggota
dalam Range

Setiap anggota
Range dijumlahkan

VBA-Excel: JumlahCerdas (A2:D4)

Data mempunyai anggota banyak disebut Item

Range (A2:D4) sebagai input diberi nama Data

```
Function JumlahCerdas(Data As Range) As Double  
' Menghitung jumlah range berupa sembarang nilai
```

```
Dim Jumlah As Double
```

```
Jumlah = 0#
```

```
For Each Item In Data
```

```
    If Application.IsNumber(Item.Value) Then  
        Jumlah = Jumlah + Item.Value
```

```
Next Item
```

```
JumlahCerdas = Jumlah
```

```
End Function
```

Menggunakan fungsi IsNumber() dalam Excel untuk cek input

Cara memproses setiap anggota dalam Range

Hanya bilangan yang dijumlahkan

VBA-Excel: Perkalian (A2:D4)

Data mempunyai anggota banyak disebut Item

Range (A2:D4) sebagai input diberi nama Data

```
Function Perkalian(Data As Range) As Double  
' Menghitung perkalian range berupa sembarang nilai
```

```
Dim Hasil As Double
```

```
Hasil = 1#
```

```
For Each Item In Data
```

Menggunakan fungsi IsNumber() dalam Excel untuk cek input

```
    If Application.IsNumber(Item.Value) Then  
        Hasil = Hasil * Item.Value
```

```
    Next Item
```

```
Perkalian = Hasil
```

Hanya bilangan yang dikalikan

```
End Function
```

Cara memproses setiap anggota dalam Range

VBA-Excel: Maksimum (A2:D4)

Data mempunyai anggota banyak disebut Item

Range (A2:D4) sebagai input

```
Function Maksimum(Data As Range) As Double  
' Menghitung maksimum range berupa sembarang nilai
```

Diisi bilangan kecil sekali

```
Dim Hasil As Double
```

```
Hasil = -99999999.99999  
For Each Item In Data
```

Cara memproses setiap anggota dalam Range

kemudian dibandingkan terhadap setiap anggota "Range"

```
    If Application.IsNumber(Item.Value)  
    And Item.Value > Hasil Then Hasil = Item.Value
```

```
Next Item
```

```
Maksimum = Hasil
```

Hanya bilangan yang diproses

```
End Function
```


VBA-Excel: Minimum (A2:D4)

Data mempunyai anggota banyak disebut Item

Range (A2:D4) sebagai input

```
Function Minimum(Data As Range) As Double  
' Menghitung minimum range berupa sembarang nilai
```

Diisi bilangan besar sekali

```
Dim Hasil As Double
```

Cara memproses setiap anggota dalam Range

```
Hasil = 99999999.99999  
For Each Item In Data
```

kemudian dibandingkan terhadap setiap anggota "Range"

```
If Application.IsNumber(Item.Value)  
And Item.Value < Hasil Then Hasil = Item.Value
```

Hanya bilangan yang diproses

```
Next Item
```

```
Minimum = Hasil
```

```
End Function
```

Variasi Penggunaan Range

Sekelompok
bilangan sebagai
range (A2:D4)

	A	B	C	D	E
1	Data input berupa "Range"				
2	Embuh	34,50	87,90	80,00	
3	34,68	23,50	q	-89,60	
4	45,00	67,80	a,g	Luk	
5					
6	Minimum =	-89,6		=Minimum(A2:D4)	
7	Maksimum =	87,9		=Maksimum(A2:D4)	
8	Perkalian =	-54.046.811.649.880,60		=Perkalian(A2:D4)	
9	Product =	-54.046.811.649.880,60		=PRODUCT(A2:D4)	
10	Maxmin =	87,9		=Maxmin(TRUE;A2:D4)	
11	Maxmin =	-89,6		=Maxmin(FALSE;A2:D4)	

Range (A2:D4) digunakan
sebagai input

Variasi fungsi
maksimum dan minimum

Apa yang terjadi jika input teks semua?

Sekelompok teks sebagai range (A2:D4)

	A	B	C	D	F	G
1	Data input berupa "Range"					
2	Embuh	ora	weruh	aku		
3	Aku	ya	ora	weruh ki		
4	Iki	ya	dha	ngapa tha!		
6	Minimum =		100000000,00	=Minimum(A2:D4)	Salah	
7	Excel: Min =		0,00	=MIN(A2:D4)	Benar	
8	Maksimum =		-100000000,00	=Maksimum(A2:D4)	Salah	
9	Excel: Max =		0,00	=MAX(A2:D4)	Benar	
10	Perkalian =		1,00	=Perkalian(A2:D4)	Salah	
11	Excel: Product =		0,00	=PRODUCT(A2:D4)	Benar	
12	Maxmin =	Input teks semua!		=Maxmin(TRUE;A2:D4)	Lebih informatif!	
13	Maxmin =	Input teks semua!		=Maxmin(FALSE;A2:D4)	Lebih informatif!	
14	Minmax =	Input teks semua!		=Minmax(TRUE;A2:D4)	Alternatif!	
15	Minmax =	Input teks semua!		=Minmax(FALSE;A2:D4)	Alternatif!	
16	Proses =	Input teks semua!		=Proses(F16;A2:D4)	Rerata	
17	Excel: Average =		#DIV/0!	=AVERAGE(A2:D4)	Error	

Pikirkan perbaikannya untuk yang salah

Range (A2:D4) digunakan sebagai input

VBA-Excel: Maxmin (A2:D4) Elegan

```
Function Maxmin(Max As Boolean, Data As Range) As Variant  
' Menghitung maksimum atau minimum range berupa sembarang nilai
```

```
Dim Hasil As Variant
```

```
' 1. Inisialisasi awal
```

```
Hasil = "Input teks semua!"
```

```
For Each Item In Data
```

```
    If Application.IsNumber(Item.Value) Then
```

```
        Hasil = Item.Value
```

```
        Exit For
```

```
    End If
```

```
Next Item
```

```
If Hasil = "Input teks semua!" Then GoTo Selesai
```

```
'2. Mencari maksimum atau minimum dengan perbandingan
```

```
For Each Item In Data
```

```
    If Max Then
```

```
        If Application.IsNumber(Item.Value) _
```

```
            And Item.Value > Hasil Then Hasil = Item.Value
```

```
    Else
```

```
        If Application.IsNumber(Item.Value) _
```

```
            And Item.Value < Hasil Then Hasil = Item.Value
```

```
    End If
```

```
Next Item
```

```
'3. Menyimpan hasilnya
```

```
Selesai:
```

```
    Maxmin = Hasil
```

```
End Function
```

Pada awalnya Hasil harus diisi dengan bilangan

Untuk memastikan paling tidak ada SATU bilangan dalam Data

Jika terdapat bilangan maka dibandingkan terhadap setiap anggota "Range"

Blok Max

Blok Min

Lebih elegan karena dapat menyelesaikan Max dan Min!

VBA-Excel: Maxmin (A2:D4) Alternatif!

```
Function Minmax(Max As Boolean, Data As Range) As Variant  
' Menghitung maksimum atau minimum range berupa sembarang nilai
```

```
Dim Hasil As Variant
```

```
' 1. Inisialisasi awal  
Hasil = "Input teks semua!"  
For Each Item In Data  
  If Application.IsNumber(Item.Value) Then  
    Hasil = Item.Value  
    Exit For  
  End If  
Next Item
```

```
If Hasil <> "Input teks semua!" Then  
' 2. Mencari maksimum atau minimum dengan perbandingan
```

```
  For Each Item In Data  
    If Max Then  
      If Application.IsNumber(Item.Value)  
        And Item.Value > Hasil Then Hasil = Item.Value  
    Else  
      If Application.IsNumber(Item.Value)  
        And Item.Value < Hasil Then Hasil = Item.Value  
    End If  
  Next Item  
End If
```

```
' 3. Menyimpan hasilnya  
Minmax = Hasil
```

```
End Function
```

Pada awalnya Hasil harus diisi dengan bilangan

Jika terdapat bilangan maka dibandingkan terhadap setiap anggota "Range"

Jika diinginkan GoTo dapat dihindari dengan mengubah sedikit!

GoTo diubah menjadi blok If ... then ... End If

Blok Max

Blok Min

VBA-Excel: Minmax (A2:D4) "IsNumeric"

```
Function Minmax(Max As Boolean, Data As Range) As Variant  
' Menghitung maksimum atau minimum range berupa sembarang nilai
```

```
Dim Hasil As Variant
```

```
' 1. Inisialisasi awal  
Hasil = "Input teks semua!"  
For Each Item In Data  
  If IsNumeric(Item.Value) Then  
    Hasil = Item.Value  
    Exit For  
  End If  
Next Item
```

Pada awalnya Hasil harus diisi dengan bilangan

Jika diinginkan GoTo dapat dihindari dengan mengubah sedikit!

GoTo diubah menjadi blok If ... then ... End If

```
If Hasil <> "Input teks semua!" Then  
' 2. Mencari maksimum atau minimum dengan perbandingan
```

```
For Each Item In Data  
  If Max Then  
    If IsNumeric(Item.Value) And Item.Value > Hasil Then Hasil = Item.Value  
  Else  
    If IsNumeric(Item.Value) And Item.Value < Hasil Then Hasil = Item.Value  
  End If  
Next Item  
End If
```

Jika terdapat bilangan maka dibandingkan terhadap setiap anggota "Range"

Blok Max

Blok Min

```
' 3. Menyimpan hasilnya  
Minmax = Hasil
```

```
End Function
```

Application.IsNumber milik Excel diganti dengan IsNumeric milik VBA

Select Case ... End Select

Select Case Permintaan

Case "Rerata" 'Mencari rerata

Hasil = 0#

NData = 0

For Each Sel **In** Data

If IsNumeric(Sel.Value) **Then**

NData = NData + 1

Hasil = Hasil + Sel.Value

End If

Next Sel

Hasil = Hasil / NData

Case Else

→ Hasil = "Tidak tersedia pilihan:" & Permintaan

End Select

Beberapa Case dapat dimasukkan dalam Blok 1

Kasus lainnya dimasukkan ke Case Else dalam Blok 2



Contoh: Select Case ... End Select



```
Select Case Permintaan
```

```
Case "Max" 'Mencari maksimum
```

```
For Each Sel In Data
```

```
    If IsNumeric(Sel.Value) _
```

```
        And Sel.Value > Hasil Then Hasil = Sel.Value
```

```
Next Sel
```

Blok Max

```
Case "Min" 'Mencari minimum
```

```
For Each Sel In Data
```

```
    If IsNumeric(Sel.Value) _
```

```
        And Sel.Value < Hasil Then Hasil = Sel.Value
```

```
Next Sel
```

Blok Min

```
Case "Jumlah" 'Mencari penjumlahan
```

```
Hasil = 0#
```

```
For Each Sel In Data
```

```
    If IsNumeric(Sel.Value) Then Hasil = Hasil + Sel.Value
```

```
Next Sel
```

Blok Jumlah

```
Case "Kali" 'Mencari perkalian
```

```
Hasil = 1#
```

```
For Each Sel In Data
```

```
    If IsNumeric(Sel.Value) Then Hasil = Hasil * Sel.Value
```

```
Next Sel
```

Blok Kali

```
Case "Rerata" 'Mencari rerata
```

```
Hasil = 0#
```

```
NData = 0
```

```
For Each Sel In Data
```

```
    If IsNumeric(Sel.Value) Then
```

```
        NData = NData + 1
```

```
        Hasil = Hasil + Sel.Value
```

```
    End If
```

```
Next Sel
```

```
Hasil = Hasil / NData
```

Blok Rerata

```
Case Else
```

```
    Hasil = "Tidak tersedia pilihan: " & Permintaan
```

```
End Select
```



Online Manual



VISUAL BASIC LANGUAGE REFERENCE

Syntax: For Each ... Next Loop

For Each *element* **In** *group*

[*statements*]

[**Exit For**]

[*statements*]

Next [*element*]

Perhatikan konvensi penulisan dalam manual Bahasa Pemrograman VBA

Penjelasan dalam [...] boleh tidak ada!

The **For...Each...Next** statement syntax has these parts:

Part	Description
<i>element</i>	Required. <u>Variable</u> used to iterate through the elements of the collection or array. For collections, <i>element</i> can only be a <u>Variant</u> variable, a generic object variable, or any specific object variable. For arrays, <i>element</i> can only be a Variant variable.
<i>group</i>	Required. Name of an object collection or array (except an array of <u>user-defined types</u>).
<i>statements</i>	Optional. One or more statements that are executed on each item in <i>group</i> .

Syntax: Select Case ... End Select

Select Case *testexpression*

[**Case** *expressionlist-n* [*statements-n*]]

[**Case Else** [*elasticsearch*]]

End Select

**Penjelasan
dalam [...]
boleh tidak
ada!**

**Perhatikan konvensi
penulisan dalam
manual Bahasa
Pemrograman VBA**

The **Select Case** statement syntax has these parts:

Part	Description
<i>testexpression</i>	Required. Any <u>numeric expression</u> or <u>string expression</u> .
<i>expressionlist-n</i>	Required if a Case appears. Delimited list of one or more of the following forms: <i>expression</i> , <i>expression</i> To <i>expression</i> , Is <i>comparisonoperator</i> <i>expression</i> . The To <u>keyword</u> specifies a range of values. If you use the To keyword, the smaller value must appear before To . Use the Is keyword with <u>comparison operators</u> (except Is and Like) to specify a range of values. If not supplied, the Is keyword is automatically inserted.
<i>statements-n</i>	Optional. One or more statements executed if <i>testexpression</i> matches any part of <i>expressionlist-n</i> .
<i>elsestatements</i>	Optional. One or more statements executed if <i>testexpression</i> doesn't match any of the Case clause.

Contoh Select Case

```
Function Bonus (performance, salary)
  Select Case performance
    Case 1
      Bonus = salary * 0.1
    Case 2, 3
      Bonus = salary * 0.09
    Case 4 To 6
      Bonus = salary * 0.07
    Case Is > 8
      Bonus = 100
    Case Else
      Bonus = 0
  End Select
End Function
```