

MODUL IV

PERULANGAN (LOOPING)

beberapa jenis perulangan dalam pemrograman Visual Basic pada umumnya tidak jauh berbeda dengan bahasa pemrograman lainnya, berikut ialah beberapa jenis looping (perulangan) dalam pemrograman Visual Basic :

1. Do.....Loop

Gunakan Do loop untuk mengeksekusi suatu blok statement dengan jumlah perulanganyang tak terhingga, ada beberapa Do...Loop statement, tetapi masing-masingmengevaluasi suatu kondisi untuk menentukan apakah melanjutkan eksekusi Berikut ini adalah statement Do... Loop, yang dijalankan selama kondisi benar :

```
Do While condition
statements
Loop
```

Ketika Visual menjalankan Do loop ini, pertama kali akan di coba kondisinya, jika kondisi False (zero), akan diloncati semua statements yang mengikuti kondisi tersebut. Visual Basic akan menjalankan statements jika kondisi benar dan kembali ke Do...Loop berikutnya.

Contoh :

```
function faktorial(x as integer)
dim hasil as long
dim i as integer
hasil=hasil*i
for i=1 to x do
hasil=hasil*i
next I
end function
```

berikut ialah penerapan perulangan pada contoh aplikasi :

Contoh 1 :

Dim X As Integer

Dim HASIL As Long

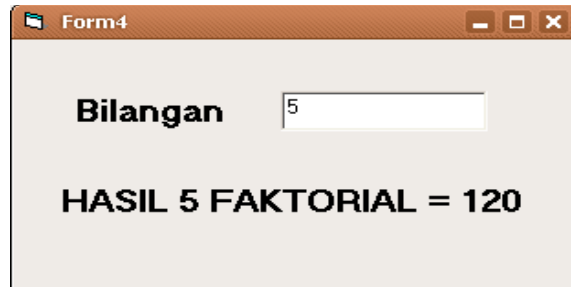
Dim I As Integer

Private Sub Text1_KeyPress(KeyAscii As Integer)

```

X = Val(Text1)
If KeyAscii = 13 Then
HASIL = 1
For I = 1 To X
HASIL = HASIL * I
Next I
Label1.Caption = "HASIL " & X & " FAKTORIAL = " & CLng(HASIL)
End If
End Sub

```



2. For...Next

For Next adalah pilihan yang cukup baik, tidak seperti Do..Loop, For..Next menggunakan suatu variable yang disebut counter yang mana akan bertambah atau berkurang pada setiap perulangan, syntaxnya adalah :

```

For Counter = Start To End [step increment]
.statements
Next [Counter]

```

Argumen Counter, start, end dan increement semuanya adalah numerik. Yang dilakukan visual Basic ketika melakukan For Next adalah :

- a) Menset nilai counter dengan start
- b) mencoba apakah counter lebih besar dari end. Jika ya, visual basic meninggalkan looping tersebut
- c) Menjalankan statements tersebut
- d) menaikkan counter dengan 1 atau dengan nilai step yang ditentukan
- e) mengulangi langkah 2 sampai 4

3. While...Wend

While...Wend digunakan untuk mengeksekusi suatu blok statement dengan jumlah perulangan yang tak terhingga. Berikut ini adalah statement While-Wend yang dijalankan selama kondisi benar :

```

While Condition
statements
Wend

```

ketika menjalankan while ini, pertama kali akan dicoba kondisinya, jika kondisi false (zero), akan diloncati semua statement yang mengikuti kondisi tersebut. Program akan menjalankan statement jika

kondisi benar dan kembali ke while berikutnya.

JENIS LOOPING Do loop until

BANYAKNYA LOOPING 66

kiezane

LOOPING

kiezane data ke - 1 Do -- Loop Until
kiezane data ke - 2 Do -- Loop Until
kiezane data ke - 3 Do -- Loop Until
kiezane data ke - 4 Do -- Loop Until
kiezane data ke - 5 Do -- Loop Until
kiezane data ke - 6 Do -- Loop Until
kiezane data ke - 7 Do -- Loop Until
kiezane data ke - 8 Do -- Loop Until
kiezane data ke - 9 Do -- Loop Until
kiezane data ke - 10 Do -- Loop Until
kiezane data ke - 11 Do -- Loop Until
kiezane data ke - 12 Do -- Loop Until
kiezane data ke - 13 Do -- Loop Until

EXIT

Contoh 2 :

Komponen	Property	Nilai / Value
Label	Name	Label1, label2
Text Box	Name	Text1, Text2
Command Button	Name	Command1, Command2
List Box	name	List1
Combo Box	Name	cmbJenisLooping

Berikut ialah listing Programnya :

```
Private Sub Command1_Click()
```

```
Dim Count As Integer
```

```
Dim i As Integer
```

```
Count = Val(Text1.Text)
```

```
If Count <= 0 Then
```

```
MsgBox "Nilai harus besar dari 0", vbInformation, "Error"
```

```
Else
```

```
List1.Clear
```

```
i = 0
```

```
Select Case cmbJenisLooping.ListIndex
```

```
Case 0:
```

```
For i = 0 To Count - 1
```

```

List1.AddItem Text2.Text & " data ke - " & i + 1 & " For -- Next", i
Next i
Case 1:
While i < Count
List1.AddItem Text2.Text & " data ke - " & i + 1 & " While -- Wend", i
i = i + 1
Wend
Case Else
Do
List1.AddItem Text2.Text & " data ke - " & i + 1 & " Do -- Loop Until", i
i = i + 1
Loop Until i > Count - 1
End Select
End If
End Sub

```

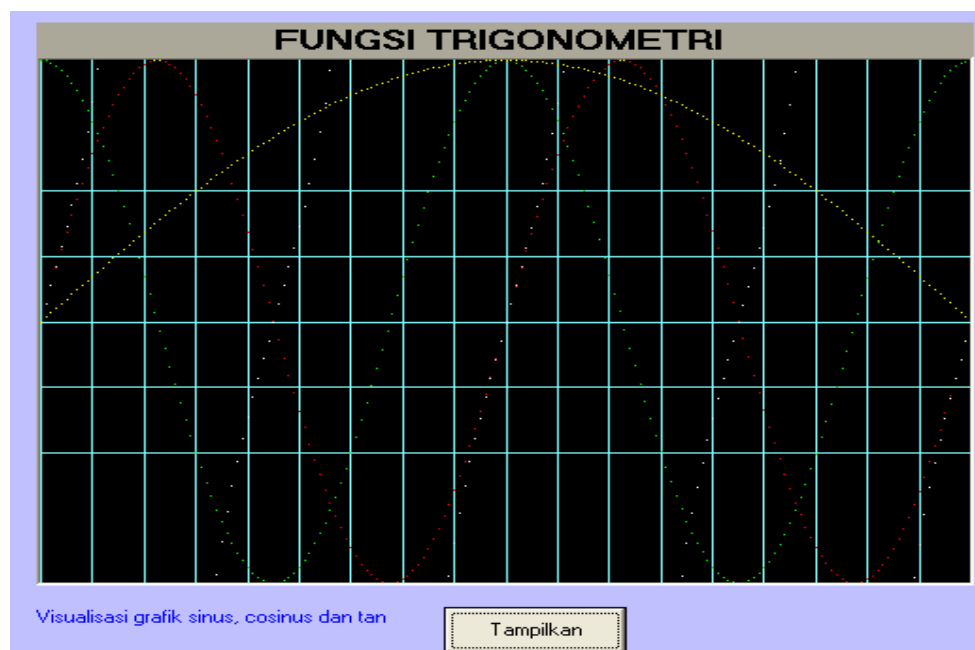
```

Private Sub Form_Load()
cmbJenisLooping.AddItem "For -- next"
cmbJenisLooping.AddItem "While -- wend"
cmbJenisLooping.AddItem "Do loop until"
cmbJenisLooping.ListIndex = 0
End Sub

```

Latihan Aplikasi

pengenalan konfigurasi objek picture (dalam grafik fungsi Trigonometri)



Komponen	Property	Nilai / Value
Command	name	command1
pictureBox	Name BackColor Appearance	MyPicture1 &H000000000& 1-3D
Label	Name	Label1, Label2

Berikut Adalah Scriptnya

Dim x As String

Dim angka As Double

Private Sub Command1_Click()

mypicture.ScaleMode = 0

mypicture.ScaleWidth = 180

mypicture.ScaleHeight = 2

mypicture.ScaleLeft = 0

mypicture.ScaleTop = -1

For j = 0 To 360

If j Mod 10 = 0 Then

mypicture.Line (j, -1)-(j, 1)

End If

Next j

mypicture.Line (0, -0.5)-(360, -0.5)

mypicture.Line (90, -1)-(90, 1)

mypicture.Line (0, -0.25)-(360, -0.25)

mypicture.Line (0, 0.25)-(360, 0.25)

mypicture.Line (0, 0)-(360, 0)

mypicture.Line (0, 0.5)-(360, 0.5)

For i = 0 To 360

mypicture.ForeColor = vbRed

mypicture.PSet (i, -Sin(i * 3.14 / 45))

mypicture.ForeColor = vbGreen

mypicture.PSet (i, -Cos(i * 3.14 / 45))

mypicture.ForeColor = vbWhite

mypicture.PSet (i, -Tan(i * 3.14 / 45))

Next i

For i = 0 To 180

```

mypicture.ForeColor = vbYellow
mypicture.PSet (i, -Sin(i * 3.14 / 180))
Next i
For i = 1 To 360
mypicture.ForeColor = vbYellow
mypicture.PSet (i, -1 / (Sin(i * 3.14 / 45)))
Next i

```

End Sub

dari contoh diatas, akan dibuat 3 buah fungsi trigonometri sederhana melalui perulangan (Looping) untuk menggambarkan titik-titik yang akan membentuk pola grafik dari Fungsi Sinus, Cosinus Dan Tan, berikut penggambaran grid secara berulang.

Latihan Soal.

Buatlah sebuah random Generator bilangan, dimana batas awal dan batas akhir nilai yang akan di Randomnya dapat ditentukan / di setting oleh user. Data akan tampil pada listbox, kemudian lakukanlah pengurutan (sorting) dengan algorithma sebagai berikut :

Public Sub SelectionSort(numbers() As Long, ByVal num As Long)

```

Dim i As Long
    Dim j As Long
    Dim min As Long
    Dim temp As Long
    temp = 0
    'do the selection sort
    For i = 0 To (num - 2)
        min = i
        For j = (i + 1) To (num - 1)
            If numbers(j) < numbers(min) Then
                min = j
            End If
        Next j
        temp = numbers(i)
        numbers(i) = numbers(min)
        numbers(min) = temp
    Next i

```

End Sub

List	Sorted List
382	2
179	2
54	3
354	11
25	13
269	14
365	15
308	15

sort