

PERTEMUAN KE-6

STRUKTUR PERULANGAN

(menggunakan Loop dan Timer)

Tujuan:

1. Menggunakan loop For ...Next untuk menjalankan pernyataan selama beberapa kali.
2. Menampilkan keluaran pada form menggunakan metode Print
3. Menggunakan loop Do untuk mengeksekusi pernyataan sampai kondisi tertentu terpenuhi
4. Melakukan loop selama beberapa waktu menggunakan objek timer
5. Dapat membuat jam digital dan alarm perjanjian

Pada pembahasan ini, Anda akan belajar cara menjalankan blok pernyataan berulang kali menggunakan loop. Anda akan menggunakan loop ForNext untuk menjalankan pernyataan beberapa kali, dan menggunakan loop Do untuk menjalankan pernyataan sampai suatu kondisi pada loop bernilai True. Metode Print merupakan metode yang digunakan untuk menampilkan teks dan angka pada form serta menggunakan objek timer untuk menjaankan kode pda interval waktu tertentu pada program Anda.

A. Loop For Next

Loop For ... Next berfungsi mengeksekusi sekelompok pernyataan program selama beberapa kali dalam suatu event procedure.

Sintaks untuk loop For ... Next adalah sebagai berikut:

```
For variable = start To end  
    Statements to be repeated  
Next variable
```

Contoh, loop For ... Next berikut ini akan membunyikan beep empat kali berturut-turut dari speaker komputer:

```
For i = 1 to 4  
    Beep  
Next i
```

Loop ini setara dengan menuliskan pernyataan Beep empat kali pada sebuah prosedur.

Variabel yang digunakan loop adalah i, yang merupakan counter integer pertama pada loop For ... Next. Setiap kali loop dijalankan, variabel counter akan dinaikkan satu.

Salah satu teknik untuk menampilkan variabel counter adalah menggunakan metode Print, sebuah pernyataan khusus yang menampilkan keluaran ke dalam form atau mencetak keluaran pada printer. Metode Print memiliki sintaks sebagai berikut:

```
Print expression
```

Expression adalah variabel, properti , nilai teks, atau numerik pada prosedur.

A.1 Menampilkan informasi dengan loop For ... Next

1. Bukalah sebuah objek baru
2. Gunakan tombol CommandButton untuk membuat tombol perintah di sebelah kanan form
3. Ubahlah properti Caption dari tombol perintah menjadi "Loop"
4. Ubahlah properti AutoRedraw menjadi True.
5. Ketikkan pernyataan program berikut ini pada prosedur:

```
Private Sub Command1_Click()  
For i = 1 To 10  
    Print "Line: "; i  
Next i  
End Sub
```

6. Simpan dengan nama MyForLoop
7. Jalankan program.

A2. Mengubah properti FontSize

1. Buka event procedure Command1_Click
2. Sisipkanlah instruksi berikut ini di bawah pernyataan For:
FontSize = 10 + i
3. Simpan dengan nama MyGrowFont.
4. Jalankan dan amati apa yang terjadi.

A.3 Membuat Loop For ...Next yang lebih rumit

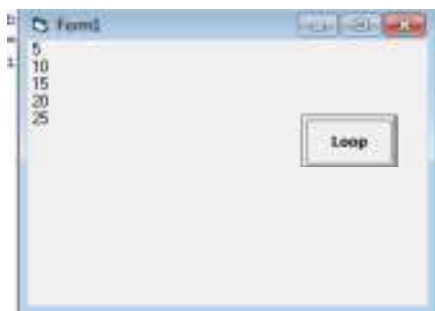
Untuk membuat loop dengan pola counter selain 1, 2, 3, 4, dan seterusnya, Anda dapat menentukan nilai yang berbeda untuk *start* pada loop, lalu menggunakan Step untuk menaikkan counter dengan interval yang berbeda.

Contoh:

```
For i = 5 To 25 step 5  
    Print i
```

```
Next i
```

Hasil



Menentukan nilai desimal pada loop:

```
For i = 1 To 2.5 step 0.5  
    Print i
```

```
Next i
```

Hasil



Kegunaan counter:

1. Menampilkan variabel counter
2. Mengatur properti
3. Menghitung nilai
4. Memproses file

B. Pernyataan Exit For

Pernyataan Exit For berfungsi untuk keluar dari loop For... Next sebelum loop selesai dieksekusi. Dengan cara ini, Anda dapat merespon event tertentu yang terjadi sebelum loop dijalankan sesuai jumlahnya yang telah ditentukan. Contoh, pada loop For...Next berikut ini:

```
For i = 1 To 10
    InpName = InputBox("Enter your name or type down to quit.")
    If InpName = "Down" Then Exit For
    Print InpName
Next i
```

Loop meminta pemakai memasukkan 10 nama dan mencetaknya pada form, kecuali jika kata Down diketikkan (dalam hal ini, program akan melompat ke pernyataan Next). Pernyataan Exit For biasanya digunakan dengan pernyataan If. Pernyataan ini sangat berguna untuk menangani kasus tertentu yang muncul pada loop, seperti berhenti apabila telah mencapai batas yang telah ditentukan.

C. Menulis Loop Do

Alternatif untuk loop For..Next, adalah dengan menuliskan loop Do yang menjalankan sekelompok pernyataan sampai kondisi tertentu nilainya True pada loop tersebut.

Loop Do mempunyai beberapa format, tergantung pada di mana dan bagaimana kondisi loop dievaluasi. Sintaks yang paling umum adalah sebagai berikut:

```
Do While condition
    Block of statements to be executed
Loop
```

Sebagai contoh, loop Do berikut ini akan memproses masukan sampai kata Done diketikkan:

```
Do While InpName <> "Done"  
    InpName = InputBox("Enter your name or type Done to quit.")  
    If InpName <> "Done" Then Print InpName  
Loop
```

Pernyataan berkondisi pada loop ini adalah `InpName <> "Done"`, yang diterjemahkan oleh compiler Visual Basic sebagai "lakukan loop selama variabel `InpName` tidak mengandung kata `Done`." Di sini dapat diketahui sifat loop `Do`: jika kondisi di bagian atas loop tidak `True` pada waktu pernyataan `Do` pertama kali diperiksa, loop `Do` tidak dijalankan. Di sini variabel `InpName` tidak mengandung string teks `"Done"` sebelum loop dimulai (mungkin dari penugasan sebelumnya pada event procedure), Visual Basic akan melompati loop dan melanjutkan baris di bawah kata `Loop`. Perhatikan jenis loop ini membutuhkan `If...Then` tambahan untuk mencegah nilai exit ditampilkan apabila pemakai mengetikkannya.

Jika Anda ingin agar loop dijalankan setidaknya sekali pada program, letakkan pengujian kondisi di bagian bawah loop. Sebagai contoh:

```
Do  
    InpName = InputBox("Enter your name or type Done to quit.")  
    If InpName <> "Done" Then Print InpName  
Loop While InpName <> "Done"
```

Program di atas sebenarnya sama dengan loop `Do` sebelumnya, tetapi di sini kondisi loop diuji setelah nama diterima dari fungsi `InputBox`. Cara ini memiliki keuntungan yaitu dapat meng-update variabel `InpName` sebelum pengujian kondisi pada loop, jadi nilai `"Done"` yang sudah ada sebelumnya tidak akan menyebabkan loop dilompati. Menguji kondisi loop di bagian bawah memastikan agar loop Anda dijalankan setidaknya satu kali, tetapi terkadang hal ini membutuhkan beberapa pernyataan tambahan untuk memproses data.

Perhatikan contoh berikut:

```
Do  
    Number = InputBox("Enter a number to square. Type -1 to quit.")  
    Number = Number * Number  
    Print Number  
Loop While Number >=0
```

Pada loop ini, pemakai mengetikkan angka secara terus-menerus, dan program mengangkat setiap angka dan mencetaknya pada layar. **Apakah kekurangan dari program di atas? Bagaimana cara mengatasinya?**

D. Latihan menunjukkan bagaimana cara menggunakan loop Do

D1. mengubah temperatur dari Farenheit menjadi Celcius.

1. Klik perintah New Project
2. Buka jendela propertis, lalu ubah properti visible dari form mejadi False
3. Amati apa yang terjadi!
4. Ketikkan pernyataan program berikut:

```
Private Sub Form_Load()
    Prompt = "Enter a farenheit temperature."
    Do
        FTemp = InputBox(Prompt, "Fahrenheit to Celcius")
        If FTemp <> "" Then
            Celcius = Int((FTemp + 40) * 5 / 9 - 40)
            MsgBox (Celcius), , "Temperature in Celcius"
        End If
    Loop While FTemp <> ""
End Sub
```

D2. Pengulangan atau Looping

Buatlah design berikut:



Object	Propertis	Setting
Form1	Name Caption StartUpPostion	Frmlooping ...:Program Looping:... 2 - CenterScreen
TextBox1	Name Text	Text1 Tulisan text1 di hapus
TextBox2	Name Text	Text2 Tulisan text2 di hapus
ListBox	Name	List1
Command Button1	Name Caption	cmduntil1 Do until 1
CommandButton2	Name Caption	cmduntil2 Do until 2
CommandButton3	Name Caption	cmdnext1 For next 1

CommandButton4	Name Caption	cmdnext2 For next 2
CommandButton5	Name Caption	Cmdwhile1 Do while 1
CommandButton6	Name Caption	cmdwhile2 Do while 2
CommandButton7	Name Caption	Cmdclear clear
Frame1	Caption	Looping until
Frame2	Caption	Looping next
Frame3	Caption	Looping While
ComboBox	Name Text	combo1 dihapus

```

Private Sub cmdclear_Click()
    List1.Clear
    combo1.Clear
    Text1 = ""
    Text2 = ""
End Sub

Private Sub cmdnext1_Click()
    List1.Clear
    For i = 1 To 10 Step 2
        List1.AddItem "ANGKA" & i
    Next i
End Sub

Private Sub cmdnext2_Click()
    List1.Clear
    For i = 2 To 10 Step 2
        List1.AddItem "ANGKA" & i
    Next i
End Sub

Private Sub cmduntil1_Click()
    Text1 = ""
    i = 1
    Do Until i > 10
        Text1.SelText = i & ""
        i = i + 2
    Loop

End Sub

```

```

Private Sub cmduntil2_Click()
    Text2 = ""
    i = 2
    Do Until i > 10
        Text2.Text = i & " "
        i = i + 3
    Loop
End Sub

Private Sub cmdwhile1_Click()
    combol.Clear
    i = 10
    Do While i > 0
        combol.AddItem "ANGKA" & i
        i = i - 2
    Loop
    combol.ListIndex = 0
End Sub

Private Sub cmdwhile2_Click()
    combol.Clear
    i = 9
    Do While i > 0
        combol.AddItem "ANGKA" & i
        i = i - 2
    Loop
    combol.ListIndex = 0
End Sub

```

E. Menggunakan Objek Timer

Object Timer adalah sebuah stopwatch tersembunyi yang memberikan akses ke dalam waktu sistem program Anda.

Objek Timer memiliki keakuratan hingga 1 milidetik atau 1/1000 detik. Walaupun timer tidak kelihatan saat program berjalan, setiap timer berhubungan dengan event procedure yang akan berjalan setiap kali waktu interval timer berakhir. Interval timer diatur menggunakan properti Interval, dan timer diaktifkan dengan mengubah properti Enable menjadi True. Setelah timer diaktifkan, ia akan berjalan terus (menjalankan event procedure-nya pada interval yang telah ditentukan) sampai pemakai menghentikan program atau timer tersebut dihentikan.

Latihan:

E1. Membuat jam digital menggunakan objek timer:

1. Klik New Project
2. Buatlah form dengan ukuran kecil
3. Klik kontrol timer pada toolbox
4. Letakkan objek timer di sebelah kiri form
5. Klik kontrol label

Objek	Properti	Setting
Label1	Caption	(Empty)
	Font	Times New Roman, Bold, 24-point
	Alignment	2-center
Timer1	Interval	1000
	Enabled	True

Form1	Caption	"Digital Clock"
-------	---------	-----------------

6. Ketikkan pernyataan program berikut pada Event procedure Timer_1

```
Label1.Caption = Time
```

7. Simpan dengan MyDigClock

E2. Mengatur Batas Waktu

1. Klik New Project
2. Ubah ukuran form menjadi jendela kecil seukuran kotak Input
3. Klik kontrol TextBox
4. Letakkan di tengah-tengah form
5. Klik kontrol Label, buatlah sebuah label panjang di atas kotak teks
6. Klik kontrol CommandButton, buatlah sebuah tombol perintah di bawah kotak teks
7. Klik tombol timer, buatlah sebuah objek timer di sudut kiri bawah form

Objek	Properti	Setting
Text1	Text	(Empty)
	PasswordChar	*
Label1	Caption	"Enter your password within 15 seconds"
Command1	Caption	"Try Password"
Timer1	Interval	15000
	Enabled	True
Form1	Caption	"Password"

8. Ketikkan pernyataan berikut pada objek timer

```
MsgBox("Sorry, your time is up.")
End
```

9. Ketikkan pernyataan program berikut:

```
Private Sub Command1_Click()
    If Text1.Text = "Secret" Then
        Timer1.Enabled = False
        MsgBox ("Welcome to the system!")
    End
Else
    MsgBox ("Sorry, Friend, I don't know you.")
End If
End Sub
```

10. Simpan dengan nama MyTimesPass

E3. Buatlah suatu pengingat (Personal Appointment Reminder)