

## Struktur Kondisi If..Else, Select Case, Tipe data dan operator.

### Deklarasi Variabel

Dim namavariabel as tipedata

Contoh :

Dim tm as Integer  
Dim mid as Integer  
Dim final as Integer  
Dim NA as Single

### Tipe Data

Macam-macam tipe data dalam Visual Basic .NET

<b>Tipe Data</b>	<b>.NET Class</b>	<b>Keterangan</b>
Byte	System.Byte (Structure)	Tipe data ini berkisar antara 0-255. Berukuran 1 byte
Short	System.Int16 (Structure)	Tipe data ini dapat menampung bilangan bulat antara -32,768 hingga 32,767. Tipe data ini berukuran 2 byte.
Integer	System.Int32 (Structure)	Tipe data ini dapat menampung bilangan bulat antara -2,147,483,648 hingga 2,147,483,647. Tipe data ini berukuran 4 byte.
Long	System.Int64 (Structure)	Tipe data ini dapat menampung bilangan bulat antara - 9,223,372,036,854,775,808 hingga 9,223,372,036,854,775,807
Single	System.Single (Structure)	Tipe data ini dapat menampung bilangan real antara -3.4028235E+38 hingga -1.401298E-45 untuk nilai negative, sedangkan untuk nilai positif antara 1.401298E-45 hingga 3.4028235E+38
Double	System.Double (Structure)	Tipe data ini dapat menampung bilangan real antara -1.79769313486231570E+308 hingga -4.94065645841246544E-324 sedangkan untuk nilai positif antara 4.94065645841246544E-324 hingga 1.79769313486231570E+308
Decimal	System.Decimal (Structure)	Tipe data ini dapat menampung bilangan decimal dengan jumlah digit 0 sampai 28.
Boolean	System.Boolean (Structure)	Tipe data ini hanya dapat menampung 2 jenis nilai, yaitu <i>true</i> dan <i>false</i> . Jika direpresentasikan dengan angka nilai <i>false</i> dinyatakan dengan 0 dan nilai <i>true</i> dinyatakan dengan nilai selain 0.

Date	System.DateTime (Structure)	Tipe data ini dikhususkan untuk menampung nilai yang berupa tanggal dan waktu. Dengan memanfaatkan tipe data ini kita dapat mengolah data tanggal dengan memakai fungsi-fungsi tertentu
Char	System.Char (Structure)	Tipe data ini hanya dapat menampung satu buah karakter. Jika diisi lebih dari satu karakter maka secara otomatis akan dipotong menjadi satu.
String	System.String (Class)	Tipe data ini digunakan untuk menampung rangkaian karakter atau kata. Jumlah memory yang dipakai sesuai dengan jumlah karakter.

### Konversi Implisit dan Eksplisit

Dim Jml\_Beli As Integer  
 Dim Harga As Long  
 Dim Diskon As Single

Jml\_Beli = Cint(txtJmlBeli.Text)  
 Harga = CLng(LblHarga.Text)  
 Diskon = CSng(txtDiskon.Text)

### Fungsi-fungsi konversi antar tipe data

Nama Fungsi	Keterangan
CBool()	Untuk mengkonversi suatu nilai menjadi tipe Boolean
Cdbl()	Untuk mengkonversi suatu nilai menjadi tipe Double
CObj()	Untuk mengkonversi suatu nilai menjadi tipe Object
CByte()	Untuk mengkonversi suatu nilai menjadi tipe Byte
CDec()	Untuk mengkonversi suatu nilai menjadi tipe Decimal
CShort()	Untuk mengkonversi suatu nilai menjadi tipe Short
CChar()	Untuk mengkonversi suatu nilai menjadi tipe Char
CInt()	Untuk mengkonversi suatu nilai menjadi tipe Integer
CSng()	Untuk mengkonversi suatu nilai menjadi tipe Single
CDate()	Untuk mengkonversi suatu nilai menjadi tipe Date
CLng()	Untuk mengkonversi suatu nilai menjadi tipe Long

CStr()	Untuk mengkonversi suatu nilai menjadi tipe String
--------	--

### Struktur Kondisi If .. Then

<b>If kondisi Then</b> ..... ..... <b>End If</b>	<b>If kondisi Then</b> ..... <b>Else</b> ..... <b>End If</b>	<b>If kondisi Then</b> ..... <b>ElseIf kondisi Then</b> ..... <b>End If</b>
---	--	---

### Struktur Kondisi Select Case

<b>Select Case</b> <i>ekspresi</i> <b>Case 0</b> ..... <b>Case 1,3</b> ..... <b>Case 5 to 10</b> ..... <b>Case Is &gt; 10</b> ..... <b>Case Else</b> ..... <b>End Select</b>
---

#### 1. Operator

Visual basic menyediakan operator aritmatika, perbandingan, logika dan satu lagi yaitu operator assignment (penugasan) biasanya untuk mendefinisikan suatu nilai variabel atau nilai property.

##### a. Operator Aritmatika

Operator	Operasi
^	Pemangkatan
* /	Perkalian dan Pembagian
\	Pembagian Integer
Mod	Modulus (sisa pembagian)
+ -	Penambahan dan Pengurangan
+ &	Penggabungan String

Contoh : 1)  $A = 2^3$   
 Hasil:  $A = 8 \rightarrow 2 \times 2 \times 2$   
 2) Judul = "Visual" & "Basic"  
 Hasil: Judul = "VisualBasic"

##### b. Operator Perbandingan (Comparison)

Operator	Operasi
>	Lebih Besar
<	Lebih Kecil
>=	Lebih Besar atau sama dengan
<=	Lebih Kecil atau sama dengan
=	Sama dengan
<>	Tidak sama dengan

Contoh : IF  $2 > 4$  THEN  
 A = "Benar"  
 ELSE  
 A = "Salah"  
 END IF  
 Hasil: A = "Salah"

### c. Operator Logika

Operator	Operasi
Not	Logika tidak
And	Logika dan
Or	Logika atau

Tabel Operator Logika "And"

Pernyataan1	Pernyataan2	AND
T	T	T
T	F	F
F	T	F
F	F	F

Tabel Operator Logika "Or"

Pernyataan1	Pernyataan2	OR
T	T	T
T	F	T
F	T	T
F	F	F

Tabel Operator "NOT"

Pernyataan	NOT
T	F
F	T

Merupakan Operator pembalik nilai pernyataan.

## 2. Pengkondisian

merupakan struktur pengambilan keputusan yang berfungsi mengeksekusi satu atau lebih perintah sesuai dengan kondisi atau keadaannya. Jenis-jenisnya terdiri dari :

### a. If.. Then

<p><b>1. If kondisi Then Perintah</b></p> <p><i>Ket</i> : Digunakan jika perintah setelah <b>Then</b> hanya berisi satu baris perintah saja</p>	<p><b>Contoh :</b></p> <p><b>If 2 &gt; 1 Then</b> Print "Kondisi Benar"</p> <p>Hasil : Kondisi Benar</p>
---	--

### d. Operator Penugasan (Assignment)

Hanya berupa notasi (=) untuk mendefinisikan nilai suatu variabel atau nilai property.

Contoh : `LblKeterangan.Caption = "Baik"`  
`TxtNama.Text = "Anto"`  
`Jumlah_Beli = 100`

Contoh :

Pernyataan1		Pernyataan2	Hasil
( 2 > 0 )	And	( 5 < 8 )	→ T
( 2 > 0 )	And	( 5 > 7 )	→ F
( 1 >= 8 )	And	( 2 = 2 )	→ F
( 1 = 8 )	And	( 2 >= 6 )	→ F

**Note** : Jika salah satu pernyataan Salah/False(F) maka hasilnya False(F). Untuk bernilai Benar/True(T) kedua pernyataan harus bernilai Benar/True(T)

Contoh :

Pernyataan1		Pernyataan2	Hasil
( 2 > 0 )	OR	( 5 < 8 )	→ T
( 2 > 0 )	OR	( 5 > 7 )	→ T
( 1 >= 8 )	OR	( 2 = 2 )	→ T
( 1 = 8 )	OR	( 2 >= 6 )	→ F

**Note** : Jika salah satu saja pernyataan Benar/True(T) maka hasilnya Benar/True(T).

Contoh :

Pernyataan	Hasil
NOT ( 2 > 0 )	→ F
NOT ( 2 > 2 )	→ T

**Note** : Membalik nilai pernyataan dengan (Not /Tidak),  
 Tidak Benar (T) → Salah (F) , Tidak Salah (F) → Benar (T)

<p><b>2. If kondisi Then</b>          Blok Perintah  <b>Else</b>          Blok Perintah  <b>End if</b></p> <p><i>Ket</i> : Digunakan jika perintah setelah <b>Then</b> berisi banyak perintah (blok perintah), sedangkan <b>Else</b> akan dieksekusi jika kondisi <b>If</b> tidak terpenuhi</p>	<p>Contoh :  <b>If</b> 2 &gt; 1 <b>Then</b>          Print "Kondisi Benar"          Print "Program Visual Basic"  <b>Else</b>          Print "Kondisi Salah"          Print "Program Salah"  <b>End If</b>          Hasil : Kondisi Benar          Program Visual Basic</p>
<p><b>3. If kondisi Then</b>          Blok Perintah1  <b>ElseIf</b> kondisi <b>Then</b>          Blok Perintah2  <b>Else</b>          Blok Perintah3  <b>End if</b></p> <p><i>Ket</i> : untuk banyak kondisi yang beragam.</p>	<p>Contoh :  <b>If</b> Kode_Jur = "24" <b>Then</b>          LblJur.Caption = "Sistem Informasi"  <b>ElseIf</b> Kode_Jur = "25" <b>Then</b>          LblJur.Caption = "Teknik Informatika"  <b>Else</b>          LblJur.Caption = "Jurusan Belum Terdaftar"  <b>End If</b></p>
<p>Pengkondisian dengan operator logika "And"</p> <p><b>If</b> (2 = 2) <b>AND</b> (4 &gt; 2) <b>Then</b>          LblStatus.Caption = "KONDISI BENAR"  <b>Else</b>          LblStatus.Caption = "KONDISI SALAH"  <b>End If</b></p> <p><i>Ket</i> : Untuk And kedua kondisi harus bernilai benar agar program akan mengeksekusi perintah setelah then. Kondisi diatas akan mengeksekusi (menghasilkan) LblStatus dengan nilai caption "Kondisi Benar".</p>	<p>Pengkondisian dengan operator logika "Or"</p> <p><b>If</b> (2 = 2) <b>OR</b> (2 = 0) <b>Then</b>          LblStatus.Caption = "KONDISI BENAR"  <b>Else</b>          LblStatus.Caption = "KONDISI SALAH"  <b>End If</b></p> <p><i>Ket</i> : Untuk Or jika salah satu saja kondisi bernilai benar, maka program akan mengeksekusi perintah setelah then. Kondisi diatas akan mengeksekusi (menghasilkan) LblStatus dengan nilai caption "Kondisi Benar".</p>

**b. Select Case**

Merupakan Alternatif dari penulisan **If..Then..Else**. Penulisan nya :

```

Select Case Ekspresi
Case Nilai1
    (Blok Perintah 1)
→
Case Nilai2
    (Blok Perintah 2)
...
Case Else
    (Blok Perintah n)
End Select

```

**Contoh-1 :**

```

Select Case txtKODE.Text
Case "SI"
    LblJUR.Caption = "Sistem Informasi"
    LblProgram.Caption = "Strata-1"
Case "TI"
    LblJUR.Caption = "Teknik Informatika"
    LblProgram.Caption = "Strata-1"
Case Else
    LblJUR.Caption = "Jurusan Belum" & _
    terdaftar"
End Select

```

**Contoh-2** (Jika data berupa data angka/number) :

```
Select Case Val(txtAngka.Text)
  Case 80 To 100
    LblNilai.Caption = "A"
    LblKet.Caption = "Memuaskan"
  Case 70 To 79
    LblNilai.Caption = "B"
    LblKet.Caption = "Baik"
  Case 60 To 69
    LblNilai.Caption = "C"
    LblKet.Caption = "Cukup"
  Case 50 To 59
    LblNilai.Caption = "D"
    LblKet.Caption = "Kurang"
  Case 0 To 49
    LblNilai.Caption = "E"
    LblKet.Caption = "Gagal"
  Case Else
    MessageBox.Show("Nilai
angkanya salah", "Nilai Salah",
MessageBoxButtons.OK,
MessageBoxIcon.Information)
End Select
```

**Note:**

- 1) TxtAngka karena nilainya berupa teks sedangkan untuk jangkauan kondisi dalam case-nya berupa jangkauan nilai angka (nominal), maka txtangka harus diapit dengan fungsi **Val** agar merubah isi teksnya menjadi angka(number).
- 2) Untuk perintah didalam blok **Case Else** akan di-eksekusi/dijalankan, apabila semua kondisi tidak terpenuhi, yaitu akan menampilkan kotak pesan pemberitahuan bahwa nilai angka yang diisikan salah.