



# PEMROGRAMAN VISUAL

- ❑ Fitri Ayuning Tyas, M. Kom.
- ❑ yas.0373@gmail.com
- ❑ 081804767700
- ❑ tyas-tamimy.com

# PRAKTIKUM IV



- A. Tipe Data
- B. Variabel
- C. Konstanta
- D. Operator

# Pendahuluan



- Sebuah program tersusun dari kumpulan objek yang saling berkomunikasi satu dengan yang lain
- Data yang terbentuk dari jenis data struktur bertujuan untuk menyimpan satu nilai saja dan sering disebut dengan istilah **tipe data** sederhana
- Kemampuan objek dalam menyimpan data selama program berjalan dibedakan menjadi dua jenis yaitu objek yang dapat diubah nilainya selama program berjalan yang disebut dengan **variabel** dan objek yang tidak dapat diubah nilainya selama program berjalan yang disebut dengan istilah **konstanta**



# Tipe Data

# Daftar Tipe Data Dalam VB. NET



- **Tipe data** adalah jenis data yang disimpan dalam variabel
- Tipe Data Dalam VB. NET adalah sebagai berikut:

Tipe	Tipe .NET (CTS)	Keterangan
<b>Boolean</b>	System.Boolean	Tipe logika (True atau False)
<b>Char</b>	System.Char	Tipe karakter
<b>Date</b>	System.DateTime	Tipe tanggal dan waktu
<b>Decimal</b>	System.Decimal	Tipe numeric untuk uang ( $1.0 \times 10^{-28}$ s.d. $7.9 \times 10^{28}$ )
<b>Double</b>	System.Double	Bilangan riil (64-bit) ( $5.0 \times 10^{-324}$ s.d. $1.7 \times 10^{308}$ )
<b>Single</b>	System.Single	Bilangan riil (32-bit) ( $1.5 \times 10^{-45}$ s.d. $3.4 \times 10^{38}$ )
<b>Byte</b>	System.Byte	Bilangan bulat tak bertanda (8-bit) (0 s.d. 255)
<b>Sbyte</b>	System.SByte	Bilangan bulat bertanda (8-bit) (-128 s.d. 127)

# Daftar Tipe Data Dalam VB. NET



Tipe	Tipe .NET (CTS)	Keterangan
<b>Integer</b>	System.Int32	Bilangan Bulat (32-bit) (-2.147.483.648 s.d. 2.147.483.647)
<b>Long</b>	System.Int64	Bilangan bulat dengan rentang panjang (64-bit) (-9.223.372.036.854.775.808 s.d. 9.223.372.036.854.775.807)
<b>Short</b>	System.Int16	Bilangan bulat dengan rentang pendek (16-bit) (-32.768 s.d. 32.767)
<b>UInteger</b>	System.UInt32	Tipe Integer tak bertanda (32-bit) (0 s.d. 4.294.967.295)
<b>ULong</b>	System.UInt64	Tipe Long tak bertanda (64-bit) (0 s.d. 18.446.744.073.709.551.615)
<b>UShort</b>	System.Int16	Tipe Short tak bertanda (16-bit) (0 s.d. 65.535)



# Variabel

# Variabel



- **Variabel** adalah salah satu pengenal (*identifier*) yang digunakan untuk menampung suatu nilai yang diperlukan di dalam program
- Dalam program, nilai akan disimpan dalam memori komputer pada alamat memori tertentu.
- Untuk mengakses ke nilai tersebut, dibutuhkan suatu nama yang direlasikan dengan nilai bersangkutan. Nama inilah yang disebut **variabel**.
- Nilai variabel dapat **diubah** sesuai kebutuhan sesuai kebutuhan

# Aturan Penamaan Variabel



Aturan penamaan variabel adalah sebagai berikut:

- Harus dimulai dengan sebuah huruf
- Tidak lebih dari 255 karakter
- Tidak boleh menggunakan nama statement, fungsi, metode, objek, dan sebagainya yang merupakan bahasa dari Visual BASIC
- Tidak mengandung spasi, tanda titik(.), tanda seru(!), atau karakter @, &, \$, dan #.
- Nama variabel harus bersifat **unik** di dalam suatu blok program

# Deklarasi Variabel dengan Tipe Data



- Dalam Visual Basic, variabel dideklarasikan menggunakan kata kunci Dim dan As
- Bentuk Umum:

```
Dim NamaVariabel As TipeData
```

- Contoh:

```
'mendeklarasikan variabel a dengan tipe data Integer  
Dim a As Integer
```

```
'mendeklarasikan variabel b dengan tipe data Double  
Dim b As Double
```

```
'mendeklarasikan variabel nama dengan tipe data String  
Dim nama As String
```

```
'mendeklarasikan beberapa variabel dengan tipe data yang sama  
'tuliskan dalam satu baris kode  
'pisahkan setiap variabel dengan tanda koma  
Dim x, y, z As Integer
```

# Deklarasi Variabel



- Setiap nilai di dalam program akan disimpan di dalam alamat memori yang terpisah.
- Contoh, jika ada dua variabel a dan b, maka akan ada dua alamat memori yang dialokasikan untuk kebutuhan program.
- Gunakan operator sama dengan (=) untuk menyimpan atau mengisi nilai ke dalam suatu variabel yang sudah dideklarasikan sebelumnya.
- Contoh:

'mengisi nilai ke dalam variabel

```
a = 37
```

```
b = 37.3
```

```
nama = "Tyas Tamimy"
```



# Konstanta

# Konstanta



- **Konstanta** adalah suatu pengenal yang berguna untuk menyimpan tetapan-tetapan (nilai konstan) yang dibutuhkan oleh program, seperti nilai *Phi*, kecepatan cahaya, dan sebagainya.
- Konstanta mirip dengan variabel, bedanya nilai konstanta tidak dapat diubah

# Deklarasi Konstanta



- Bentuk Umum:

```
Const NamaKonstanta As TipeData = Nilai
```

- Contoh:

```
'pendefinisian konstanta  
Const Phi As Double = 3.14  
Const NAMASOFTWARE As String = "Visual Basic .NET"
```

- Dalam *Visual Basic .NET*, konstanta secara otomatis bersifat *Shared*, *Static*, dan *ReadOnly* sehingga kita tidak dapat mendeklarasikan konstanta menggunakan salah satu kata kunci tersebut.



# Operator

# Operator



- **Operator** adalah tanda (dan kata) khusus yang digunakan untuk melakukan operasi-operasi tertentu yang dibutuhkan oleh program. Ex. Perhitungan, perbandingan nilai, manipulasi bit dsb.
- Operator-operator dalam *Visual Basic .NET*
  1. Operator penugasan
  2. Operator aritmatika
  3. Operator relasional
  4. Operator logika

# Operator Penugasan



- *Visual Basic .NET* menggunakan operator = (sama dengan) untuk melakukan penugasan (pengisian nilai) ke dalam suatu variabel.

No	Operator	Fungsi	Ekspresi
1	=	Assignment	NamaVariabel = Nilai

# Operator Aritmatika



- Adalah operator yang digunakan untuk melakukan fungsi aritmatika

No	Operator	Fungsi	Ekspresi
1	+	Penjumlahan	$n = 7 + 3$
2	-	Pengurangan	$n = 7 - 3$
3	*	Perkalian	$n = 7 * 3$
4	/	Pembagian	$n = 7 / 3$
5	\	Pembagian bilangan bulat tanpa memperdulikan sisa baginya	$n = 7 \setminus 3$
6	Mod	Modulus, sisa pembagian	$n = 7 \text{ Mod } 3$
7	^	Perpangkatan	$n = 7 ^ 3$

# Operator Relasional



- Adalah Operator yang digunakan untuk menyatakan relasi atau perbandingan antara dua operand. Hasil dari operasi perbandingan selalu bernilai logika: True atau False

No	Operator	Fungsi	Ekspresi
1	=	Sama dengan	$2 = 2$ (True)
2	>	Lebih besar	$4 > 6$ (False)
3	<	Lebih kecil	$6 < 2$ (False)
4	>=	Lebih besar sama dengan	$8 >= 8$ (True)
5	<=	Lebih kecil sama dengan	$7 <= 9$ (True)
6	<>	Tidak sama dengan	$7 <> 9$ (True)

# Operator Logika



- Adalah operator yang digunakan untuk membandingkan dua buah nilai logika, hasil operasinya juga berupa logika. Asumsikan a bernilai 30 dan b bernilai 70

No	Operator	Fungsi	Ekspresi
1	And	Logika AND	$(a = b) \text{ And } (a < b)$ bernilai False
2	Or	Logika OR	$(a <> b) \text{ Or } (a > b)$ bernilai True
3	Xor	Logika XOR	$(a = b) \text{ Xor } (a < b)$ bernilai False
4	Not	Logika NOT	Not $(a > b)$ bernilai True

# Logika AND



- Operasi logika AND akan bernilai True hanya jika kedua operand bernilai True. Apabila ada salah satu operand bernilai False maka hasilnya False.

a	b	a And b
True	True	True
True	False	False
False	True	False
False	False	False

# Logika OR



- Operasi logika OR akan bernilai True jika salah satu operand bernilai True. Apabila kedua operand bernilai False maka hasilnya False.

a	b	a Or b
True	True	True
True	False	True
False	True	True
False	False	False

# Logika XOR



- Operasi logika XOR akan bernilai `True` jika salah satu operand bernilai `True` dan operand lainnya bernilai `False` . Apabila kedua operand bernilai `True` maka hasilnya bernilai `False`, begitu juga apabila kedua operand bernilai `False` maka hasilnya bernilai `False`.

a	b	a Xor b
True	True	False
True	False	True
False	True	True
False	False	False



\*Note

# Penulisan Tipe Data



- Penambahan akhiran pada tipe data:
  - untuk tipe Short adalah s atau S
  - untuk tipe UShort adalah us atau US

```
Dim x As Short = 88S  
Dim y As UShort = 88US
```

- untuk tipe Long adalah l atau L
- untuk tipe ULong adalah ul atau UL
- untuk tipe UInteger adalah ui atau UI

```
Dim a As Integer = 99  
Dim b As UInteger = 99UI  
Dim c As Long = 99L  
Dim d As ULong = 99UL
```

# Penulisan Tipe Data



- Penambahan akhiran pada tipe data:
  - untuk tipe `Single` adalah `f` atau `F`
  - (optional) untuk tipe `Double` adalah `r` atau `R`

```
Dim m As Single = 10.25F
```

```
Dim y As Double = 37.00 'atau ditulis: 37.00R
```

- (optional) untuk tipe `Decimal` adalah `d` atau `D`



# Praktikum

# Membuat Desain *Form* Program Kalkulator



- Buatlah desain *form* dengan kontrol/ objek berikut:



- Pengaturan *properties*

No	Kontrol/ Objek	<i>Property</i>	Nilai
1	<i>Form 1</i>	<i>Text</i> <i>StartPosition</i>	Kalkulator <i>CenterScreen</i>
2	<i>Label 1</i>	<i>Text</i>	Angka Pertama
3	<i>Label 2</i>	<i>Text</i>	Angka Kedua
4	<i>Label 3</i>	<i>Text</i>	Hasil
5	<i>TextBox1</i>	<i>Name</i>	txtangkapertama:
6	<i>TextBox2</i>	<i>Name</i>	txtangkakedua:
7	<i>TextBox3</i>	<i>Name</i>	txthasil:
8	<i>Button1</i>	<i>Name</i> <i>Text</i>	btnpenjumlahan +
9	<i>Button2</i>	<i>Name</i> <i>Text</i>	btnpengurangan -
10	<i>Button3</i>	<i>Name</i> <i>Text</i>	btnperkalian x
11	<i>Button4</i>	<i>Name</i> <i>Text</i>	btnpembagian /

# Menambahkan Kode Program



- Tombol +

```
Private Sub btnpenjumlahan_Click(sender As Object, e As EventArgs) Handles btnpenjumlahan.Click
    Dim AngkaPertama, AngkaKedua, Hasil As Double 'Deklarasi variabel
    'Ada tiga nama variabel yaitu: AngkaPertama, Angka Kedua, dan Hasil
    'Tipe data dari ketiga variabelnya adalah Double
    AngkaPertama = Val(txtangkapertama.Text) 'fungsi Val() digunakan untuk mengubah nilai STRING menjadi NUMERIK
    AngkaKedua = Val(txtangkakedua.Text)
    Hasil = AngkaPertama + AngkaKedua 'Memasukan rumus penjumlahan pada variabel Hasil
    txthasil.Text = Hasil 'Menampung hasil pada objek Texbox
End Sub
```

- Lanjutkan dengan tombol lainnya. Sesuaikan penggunaan operator aritmatika pada setiap rumusnya.

# Hasil *Running* Program



Kalkulator

Angka Pertama: 13

Angka Kedua: 2

Hasil: 15

+ - x /

Kalkulator

Angka Pertama: 13

Angka Kedua: 2

Hasil: 11

+ - x /

Kalkulator

Angka Pertama: 13

Angka Kedua: 2

Hasil: 26

+ - x /

Kalkulator

Angka Pertama: 13

Angka Kedua: 2

Hasil: 6,5

+ - x /

Selesai  
Terima Kasih