

MODUL-MODUL PRAKTIKUM VB. Net

Oleh :

CHALIFA CHAZAR



MODUL 5

ARRAY & FOR EACH...NEXT

Tujuan :

Mahasiswa bisa mendefinisikan dan menggunakan penggunaan ARRAY dan FOR EACH..NEXT dalam membuat aplikasi VB.NET

Pustaka :

Wibowo HR, Enterprise J, (2014): Buku Pintar VB .NET. Elex Media Komputindo. Jakarta.

1. Array

Array adalah sekumpulan nilai yang berhubungan satu sama lain, seperti jumlah murid dalam tiap kelas. Pada aplikasinya, Array akan lebih mudah jika menggunakan looping.

Nilai-nilai yang memiliki nama yang sama dan menggunakan sebuah angka yang disebut index untuk membedakannya.

Array merupakan suatu Type Data yang mengandung banyak variable-variable yang dapat diakses satu persatu sesuai dengan urutannya dalam array, dimana urutan indexnya dimulai dari 0. Variable-variable yang terkandung dalam array disebut juga element dari array, dan semuanya memiliki tipe yang sama. Setiap array mewarisi (inherits) semua member yang dideklarasikan pada sistem array.

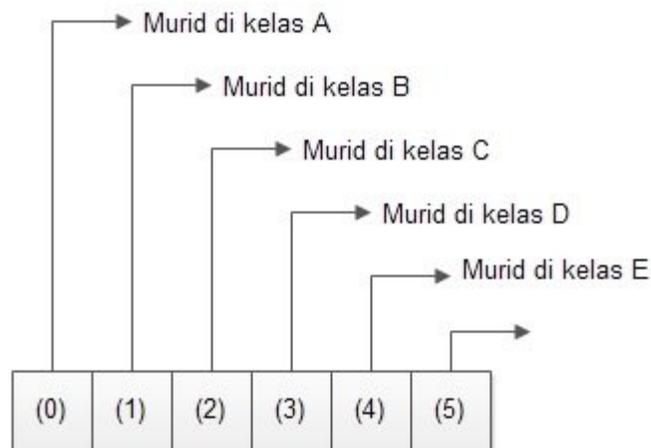
Contoh statement berikut akan mendeklarasikan sebuah variable array yang akan menyimpan jumlah murid dalam kelas A sampai dengan F.

```
Dim murid(5) As Integer
```

Array murid dalam contoh berikut ini memiliki 6 element, mulai dari element 0 sampai 5. Pendeklarasian ini lebih mudah dibandingkan Anda harus menyebutkan 6 buah variable seperti murid1, murid2, dan seterusnya.

Berikut ini adalah ilustrasi dari array murid. Untuk setiap element array tersebut:

- Index dari element mewakili kelasnya (index 0 mewakili kelas A)
- Nilai yang terkandung di dalamnya menunjukkan jumlah murid dalam kelas tersebut



Dalam Visual Basic, cara untuk menggunakan array bisa ditunjukkan dalam statement berikut, dimana jumlah murid untuk kelas A, B, dan E akan ditampilkan.

```
Dim kelasA As Integer = murid(0)
Dim kelasB As Integer = murid(1)
Dim kelasE As Integer = murid(5)
MsgBox ("jumlah murid dalam kelas A = " & CStr(kelasA))
MsgBox ("jumlah murid dalam kelas B = " & CStr(kelasB))
MsgBox ("jumlah murid dalam kelas E = " & CStr(kelasE))
```

Array pada contoh diatas menggunakan satu buah index dan biasa disebut sebagai array satu dimensi (one-dimensional). Array yang memiliki lebih dari satu index disebut array multi-dimensi (multidimensional).

1.1 Mendeklarasikan Array

Ada beberapa cara dalam mendeklarasikan sebuah array. Anda dapat menentukan ukuran dari array tersebut (berapa jumlah element di dalamnya) pada saat deklarasi.

```
'mendeklarasikan array 1 dimensi
Dim murid(5) As Integer
'mendeklarasikan array 2 dimensi
Dim AvgKelas(5, 5) As Double
```

Dalam pemrograman, setelah Anda mendeskripsikan suatu variable array, akan terjadi penambahan ukuran.

Misal, pada variable AvgKelas(5, 5), dimana Anda mencatat nilai rata-rata setiap kelas-nya, dari kelas 1A sampai F, kelas 2A sampai F, dan seterusnya sampai 6A sampai F. Namun dalam perjalanan program, kelas ditambah sampai kelas 9.

Jika kemudian Anda lakukan lagi deklarasi

```
Dim AvgKelas(8, 5) As Double
```

Anda akan mendapatkan sebuah array AvgKelas dengan ukuran (8, 5). Namun, pada saat mendeklarasikan Dim, isi dari variable tersebut akan kembali kosong. Maka apabila sebelumnya data rata-rata nilai untuk kelas 1 sampai 6 sudah disimpan, maka pada saat mendeklarasikan Dim, semua data akan hilang dan harus dilakukan perhitungan ulang.

Karena itu, diperkenalkan statement ReDim dan Preserve.

Statement ReDim akan mengubah ukuran dari array yang sudah ada. Dan statement Preserve akan mempertahankan nilai yang sudah ada.

```
Dim AvgKelas(5, 5) As Double
```

Jika ingin merubah ukuran dari array untuk penambahan kelas:

```
ReDim AvgKelas(8, 5) As Double
```

Namun data akan hilang. Jika ingin mempertahankan data:

```
ReDim Preserve AvgKelas(8, 5) As Double
```

Perlu diingat, jika melakukan pengurangan ukuran sebuah array, dan menggunakan statement Preserver, maka data akan disimpan sebanyak ukuran baru array.

```
' deklarasi murid() dengan 6 element
Dim murid(5) As Integer
' deklarasi murid() dengan mempertahankan hanya 3 nilai awal
ReDim Preserve murid(2) As Integer
' deklarasi murid() dengan 9 element dan menghapus nilai awal
ReDim murid(8) As Integer
```

Contoh pemakaian ReDim dan Preserver dalam Visual Basic adalah sebagai berikut.

```
Dim marks() As Integer
Dim i As Integer
ReDim marks(2)
marks(0) = 85
marks(1) = 75
```

```
marks(2) = 90
ReDim Preserve marks(10)
marks(3) = 80
marks(4) = 76
marks(5) = 92
marks(6) = 99
marks(7) = 79
marks(8) = 75
For i = 0 To 10
    TextBox1.Text &= (i & vbTab & marks(i) & vbNewLi ne)
Next i
```

1.2 Menyimpan Nilai Dalam Array

Anda bisa merujuk pada tiap element dalam sebuah array dengan menggunakan index-nya. Penulisan index diapit oleh sebuah tanda kurung, dengan pembagian antar index untuk array multidimensional menggunakan tanda koma.

```
Dim murid(5) As Integer
' menyimpan nilai dalam array
murid(0) = 0
murid(1) = 8
murid(5) = 10

Dim kelasA As Integer = murid(0)
Dim kelasB As Integer = murid(1)
Dim kelasE As Integer = murid(5)
MsgBox("jumlah murid dalam kelas A = " & CStr(kelasA))
MsgBox("jumlah murid dalam kelas B = " & CStr(kelasB))
MsgBox("jumlah murid dalam kelas E = " & CStr(kelasE))
```

Contoh di atas menunjukkan bagaimana cara untuk menyimpan suatu nilai dalam array.

1.3 Nilai Awal pada Array

Data awal pada sebuah array dapat dituliskan langsung pada saat deklarasi. Nilai yang akan diisikan dipisah dengan koma, dan diapit oleh sepasang tanda kurung kurawal ({}).

Pada saat membuat array dengan menuliskan datanya langsung, Anda bisa menentukan tipe datanya pada statement tersebut.

```
Dim intData() As Integer = {12, 16, 20, 24, 28, 32}
Dim names() As String = {"Karthik", "Sandhya", "Shivangi", "Ashwita"}
' "... " (string), 12D (double), 16UI (Unified Integer), "A" (char)
Dim miscData() As Object = {"Hello World", 12D, 16UI, "A"}
```

Atau juga meminta Visual Basic untuk menentukan tipe data dari datanya sendiri.

```
' mendeklarasikan array yang belum diketahui indexnya
Dim angka = New Integer() {1, 2, 4, 8}
' mendeklarasikan array yang belum diketahui indexnya dan tipe datanya
Dim doubles = {1.5, 2, 9.9, 18}
```

1.4 Perulangan Pada Array

Seperti yang sudah dituliskan sebelumnya, operasi array akan jauh dipermudah dengan menggunakan looping

Contoh berikut ini melakukan perulangan pada sebuah array satu dimensi menggunakan sebuah statement For...Next. Method GetUpperBound digunakan untuk mendapatkan nilai tertinggi dari index untuk array tersebut. Index terendah selalu adalah 0.

```
Dim angka = {10, 20, 30}
For index = 0 To angka.GetUpperBound(0)
    TextBox1.Text &= angka(index) & vbNewLine
Next
```

1.5 Jagged Arrays

Sebuah array yang memiliki array lain sebagai element dikenal sebagai sebuah array dari array, atau juga sebuah jagged array. Sebuah jagged array dan setiap element di dalamnya bisa memiliki satu atau lebih element.

2. For Each...Next

Setelah mempelajari mengenai Array, kali ini Anda kembali lagi ke pembahasan looping, dan sekarang Anda masuk ke statement For Each...Next.

Statement For Each...Next digunakan untuk mengulang suatu block statement untuk setiap element dari sebuah koleksi.

```
For Each element [ As data type ] In group
  [ statement ]
  [ continue For ]
  [ statement ]
  [ exit for ]
  [ statement ]
Next [element]
```

| Term | Definition |
|--------------|---|
| Element | Diperlukan dalam statement For Each, optional dalam statement Next. Digunakan dalam pengulangan untuk tiap element dalam koleksi |
| Data type | Tipe data dari element |
| Group | Sebuah variable dengan tipe koleksi |
| Statement | Optional. Satu atau lebih statement yang akan dilakukan di antara For Each dan Next yang dijalankan untuk setiap item dalam koleksi |
| Continue For | Optional. Lompat ke perhitungan For Each selanjutnya |
| Exit For | Optional. Lompat keluar dari block For Each |
| Next | Akhir dari block For Each...Next |

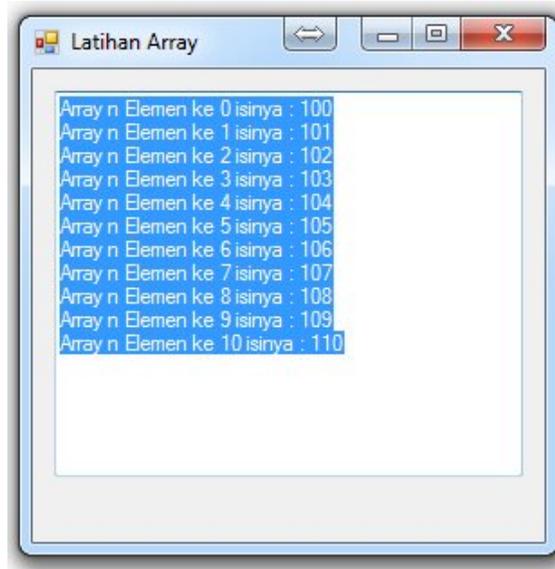
Gunakan sebuah For Each...Next ketika Anda ingin mengulang statement untuk setiap item dalam koleksi.

Contoh berikut ini akan mengulang untuk setiap element dalam sebuah array satu dimensi.

```
Dim angka = {10, 20, 30}
For Each index In angka
  TextBox1.Text &= index & vbNewLi ne
Next
```

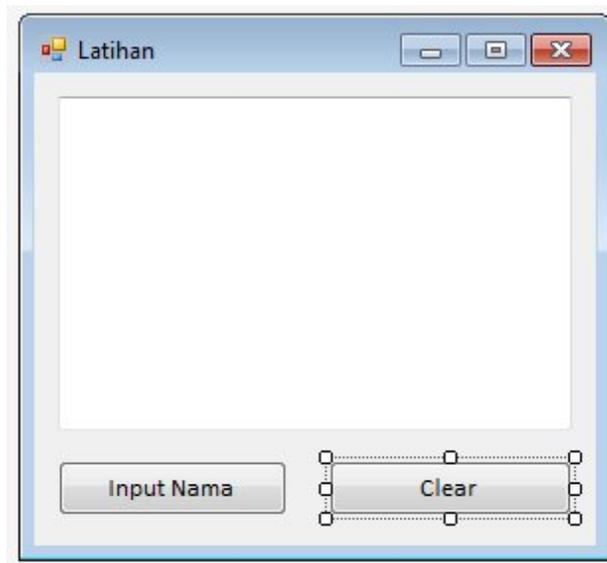
Latihan 1

Buatlah tampilan program sesuai dengan contoh dibawah ini. Simpan dengan nama Latihan2. Gunakan statement pengulangan For...Next.



Latihan 2

Buatlah tampilan program sesuai dengan contoh dibawah ini. Simpan dengan nama Latihan2.



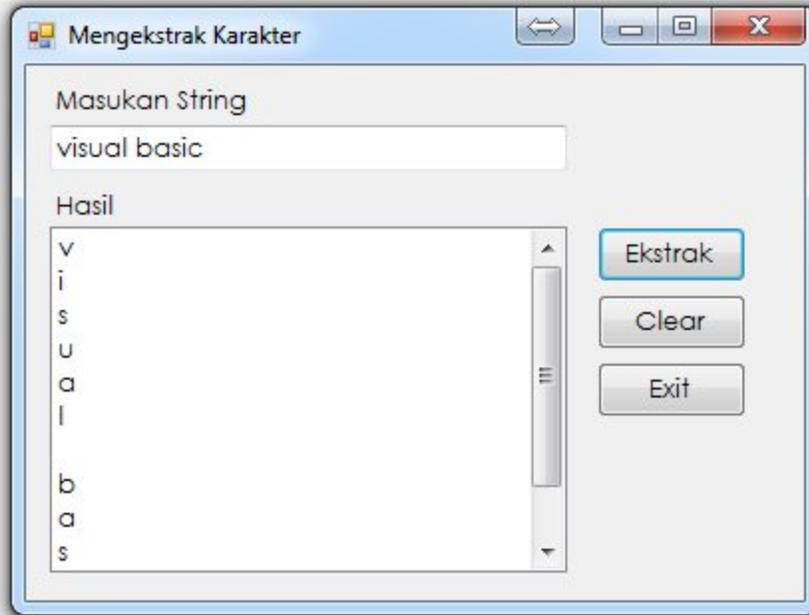
Catatan

Input nama menggunakan perintah InputBox("masukan nama")



Latihan 3

Buatlah tampilan program sesuai dengan contoh dibawah ini. Simpan dengan nama Latihan3.

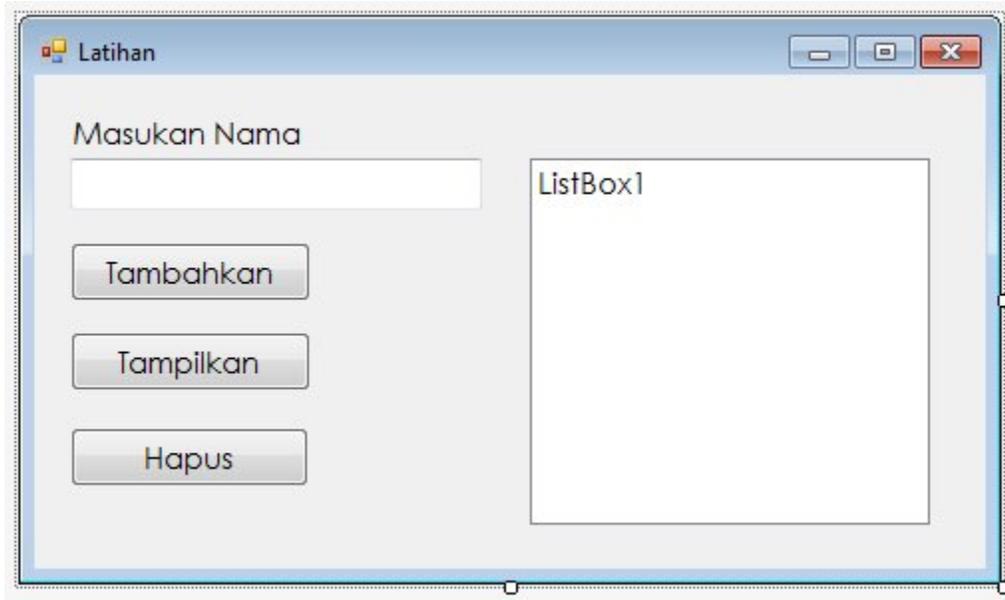


Skenario Form

Mengekstrak huruf dari masukan sebuah kalimat

Latihan 4

Buatlah tampilan program sesuai dengan contoh dibawah ini. Simpan dengan nama Latihan4.



----- GOOD LUCK -----