

MODUL  
PEMROGRAMAN VISUAL  
Visual Basic.NET



Oleh:  
Fery Updi, M.Kom

Ver.1.2017

# 1

## IDE Visual Basic .NET

### POKOK BAHASAN:

- Pengenalan Visual Basic.NET
- Konsep Pemrograman Berbasis Visual
- Tampilan Visual Basic .NET

### TUJUAN BELAJAR:

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami platform Visual Basic.NET
2. Memahami pemrograman berbasis visual
3. Memahami tampilan Visual Basic .NET

### Pengenalan Visual Basic .NET

- Visual Basic adalah salah satu bahasa pemrograman.
- Bahasa pemrograman adalah perintah-perintah yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu.
- Dikembangkan oleh Microsoft pada tahun 1991
- Merupakan pengembangan dari pendahulunya yaitu bahasa pemrograman BASIC (Beginner's All-purpose Symbolic Instruction Code)
- Bahasa **BASIC** diciptakan oleh Professor John Kemeny dan Thomas Kurtz dari Kampus Dartmouth pada pertengahan tahun 1960-an (Deitel&Deitel, 1999)

### Apa itu Visual?

- VISUAL adalah cara yang digunakan untuk membuat Graphical User Interface (GUI)
- Tidak perlu menuliskan intruksi pemrograman dalam kode-kode baris
- Cukup melakukan "drag" dan "drop" objek-objek yang akan digunakan
- Visual Basic merupakan salah satu bahasa pemrograman komputer yang mendukung object (Object Oriented Programming = OOP)

### Konsep Pemrograman Berbasis Visual

- Program berbasis visual memakai konsep event-driven:
  1. kode program tidak mengikuti alur yang ditetapkan awal
  2. eksekusi program dapat berlainan sesuai dengan event yang diberikan.
- Urutan event menentukan urutan kode yang dieksekusi, jadi alur jalannya program bisa berbeda untuk setiap eksekusi program

### IDE (Integrated Development Environment (IDE))

- Penulisan program banyak dilakukan dengan berbagai editor, misal: Notepad
- Dengan menggunakan IDE, Programmer dapat membuat user interface, melakukan koding, melakukan testing dan debugging serta mengkompilasi program menjadi executable

## Visual Basic

- **Visual Basic** merupakan bahasa pemrograman yang sangat mudah dipelajari, dengan teknik pemrograman visual yang memungkinkan penggunanya untuk berkreasi lebih baik dalam menghasilkan suatu program aplikasi
- Ini terlihat dari dasar pembuatan dalam visual basic adalah **FORM**, dimana pengguna dapat mengatur tampilan form kemudian dijalankan dalam script yang sangat mudah

## Sejarah perkembangan Visual Basic

Hingga saat ini, Visual Basic sudah hadir dalam 10 versi. Berikut peluncuran dari masing-masing versi.

1. Pada tahun 1991 => Microsoft Visual Basic Versi 1.0
2. Pada tahun 1992 => Microsoft Visual Basic Versi 2.0
3. Pada tahun 1993 => Microsoft Visual Basic Versi 3.0
4. Pada tahun 1996 => Microsoft Visual Basic Versi 4.0
5. Pada tahun 1997 => Microsoft Visual Basic Versi 5.0
6. Pada tahun 1998 => Microsoft Visual Basic Versi 6.0
7. Pada tahun 2003 => Microsoft Visual Basic Versi 7.0
8. Pada tahun 2005 => Microsoft Visual Basic Versi 8.0
9. Pada tahun 2008 => Microsoft Visual Basic Versi 9.0
10. Pada tahun 2010 => Microsoft Visual Basic Versi 10.0

## Visual Basic .NET

VB .Net adalah salah satu bahasa pemrograman dalam .Net framework. Cikal bakal dari VB .Net adalah bahasa BASIC (Beginer All-Purpose Symbolic Instruction Code) yang diciptakan tahun 1964 oleh professor John Kemeny dan Thomas Kurtz

## Platform dan Layer

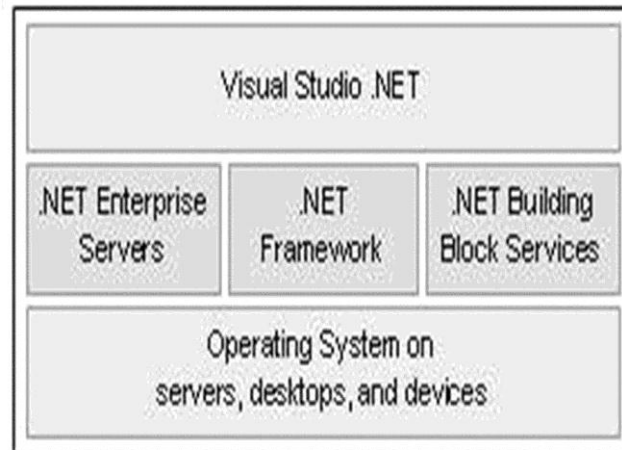
- **Platform .Net Framework**

Microsoft .Net : yang awalnya disebut Next Generation Windows Services (NGWS) adalah suatu platform untuk membangun dan menjalankan generasi penerus aplikasi-aplikasi . Microsoft.NET merupakan framework (kerangka) pengembangan yang menyediakan antarmuka pemrograman baru untuk layanan Windows dan API (Application Programming Interface)

- **Layer**

1. Microsoft .NET merupakan strategi Microsoft untuk menghubungkan sistem, informasi, dan alat (device), sehingga orang dapat berkomunikasi serta berkolaborasi dengan lebih efektif.
2. Teknologi .NET terintegrasi penuh melalui produk-produk Microsoft, dan menyediakan kemampuan untuk mengembangkan solusi dengan menggunakan Web service.

- Platform Microsoft .NET terdiri dari lima komponen utama yang tersusun dalam tiga lapisan (layer). Lapisan paling bawah adalah sistem operasi; lapisan kedua terdiri dari tiga komponen; lapisan teratas adalah Visual Studio .NET.

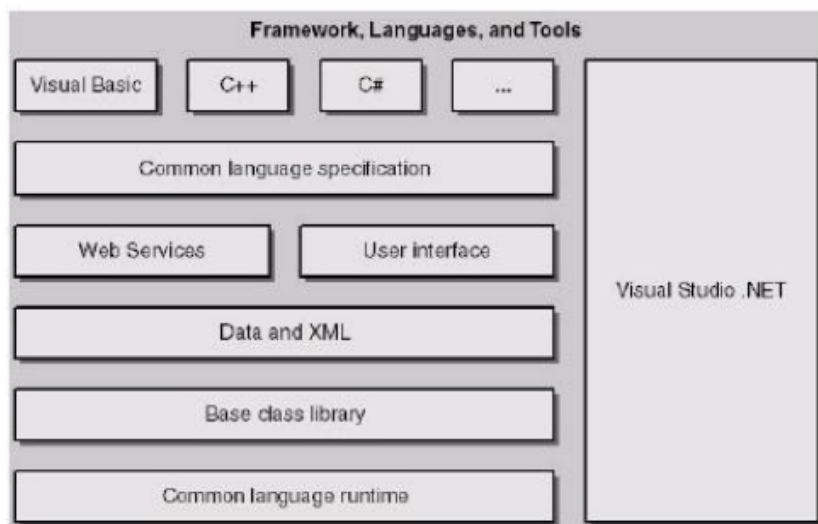


### Keuntungan mempelajari .NET

Beberapa keuntungan ketika menggunakan .Net, adalah sebagai berikut:

#### 1. Multi Language

Arsitektur .NET bersifat terbuka, sehingga memungkinkan berbagai bahasa pemrograman mengakses CLR dengan mulus. Banyak kalangan menyebut .NET sebagai "open source" versi Microsoft. Saat ini .NET dapat diprogram menggunakan Visual Basic.NET, C++.NET, Visual C#, Jscript, dan J#.



#### 2. No DLL Hell

DLL merupakan blok atau modul-modul obyek dari sebuah aplikasi. Perannya sangat penting, sekaligus memusingkan. Sering terjadi dalam dunia windows, kompatibilitas dan registrasi DLL di masing-masing Workstation menjadi isu besar dalam deployment aplikasi

#### 3. Strong Typing dan Type Safety

.NET menyediakan strong typing, dimana setiap variabel wajib didefinisikan scope dan tipe datanya. Demikian pula dengan fasilitas type safety yang sangat bermanfaat untuk membantu dalam coding pemrograman, terutama fasilitas intellisense yang membimbing pemrogram dalam menentukan property, method, maupun function yang akan dipakai.

#### 4. Cross Platform Possibility

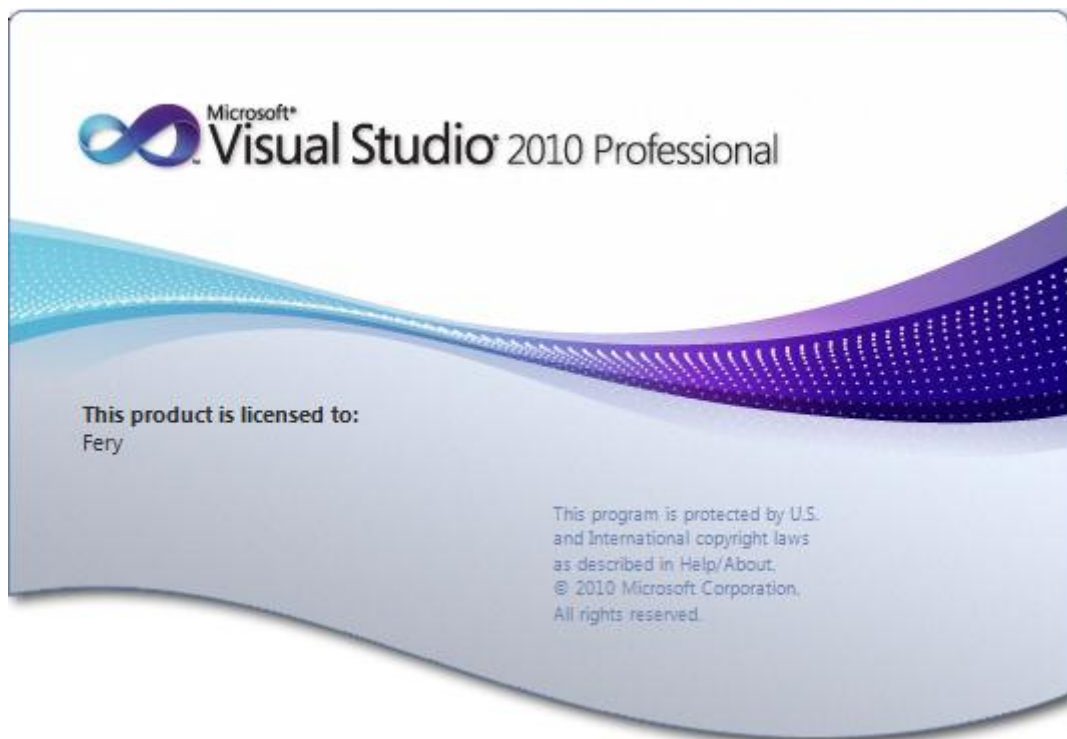
.Net menyimpan dan mengirim data dalam bentuk XML yang merupakan format data universal di internet. Dengan demikian integrasi data antar platform lebih mudah dilakukan, selama platform tersebut mendukung XML. Manipulasi format data dalam bentuk XML, .txt, maupun .rtf merupakan sesuatu yang menantang para programmer untuk membuat aplikasi lintas platform.

#### 5. Code Once, More Application

Interface pemrograman bersifat konsisten, dengan object model yang sama pada setiap bahasa yang digunakan. Suatu object baik berbentuk class, library, maupun web services dapat diakses dengan mudah oleh berbagai aplikasi windows maupun web.

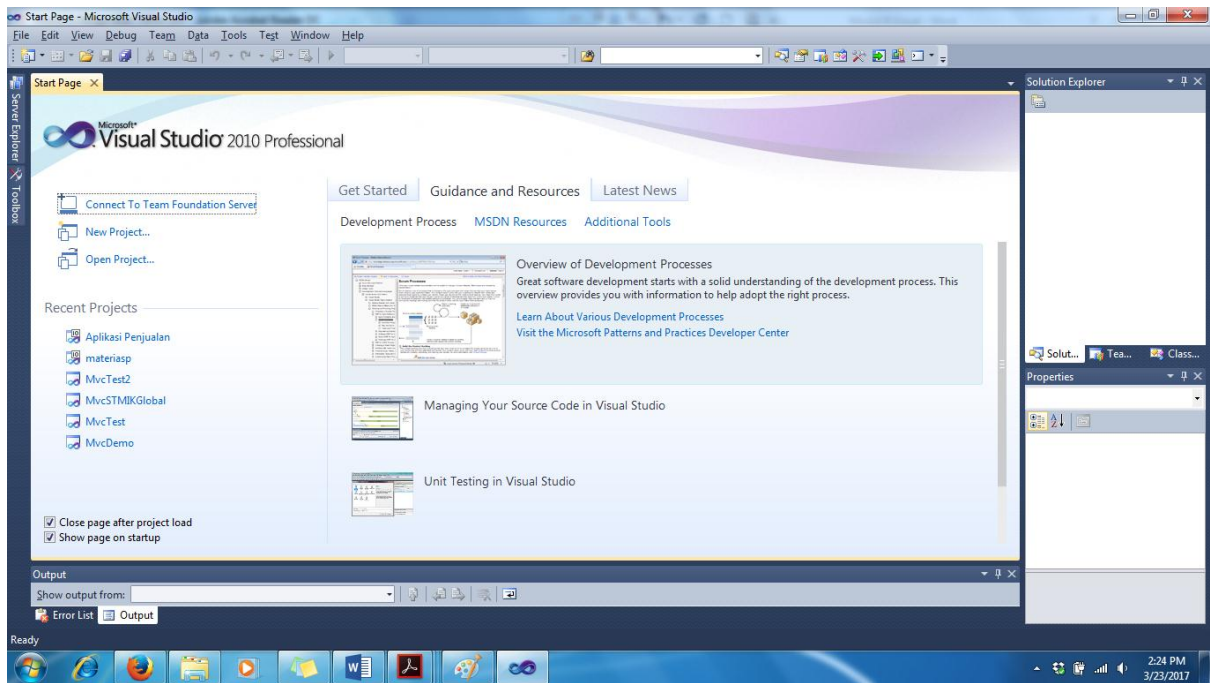
### Tampilan Visual Studio 2010

Apabila berhasil instalasinya, maka akan muncul tampilan Visual Studio 2010 seperti dibawah ini:



Gambar 1.1

Tampilan layar kerja visual basic.Net apabila instalasi berhasil dilakukan adalah sebagai berikut:



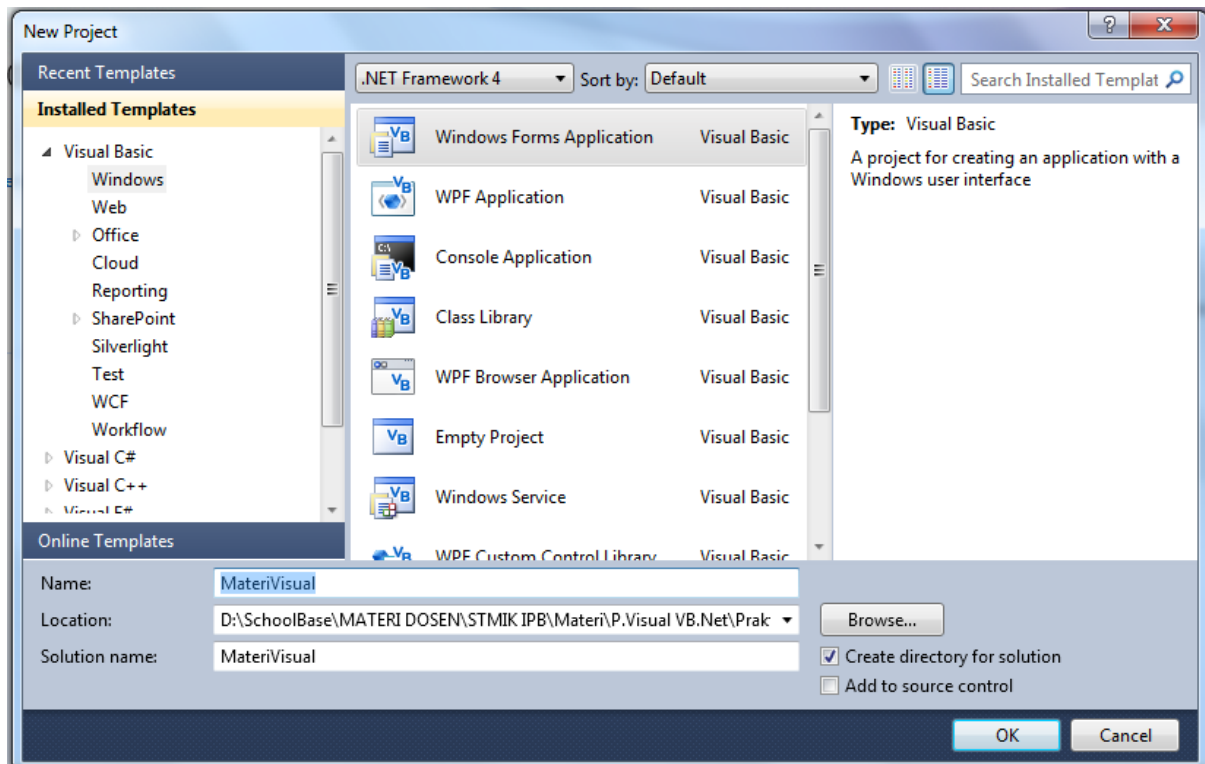
Gambar 1.2

## PRAKTIKUM

### 1. Membuat project baru di Visual Basic 2010

**Project** merupakan kerangka dasar aplikasi yang menentukan jenis aplikasi yang akan dibuat. Langkah-langkah pembuatan project adalah sebagai berikut:

- Jalankan Visual Basic 2010
- Pada bagian **Start Page** pilih **New Project**
- Pada bagian New Project pilih bahasa **Visual Basic**, Project Types: **Windows** dan Templates: **Windows From Application**
- Klik **OK**

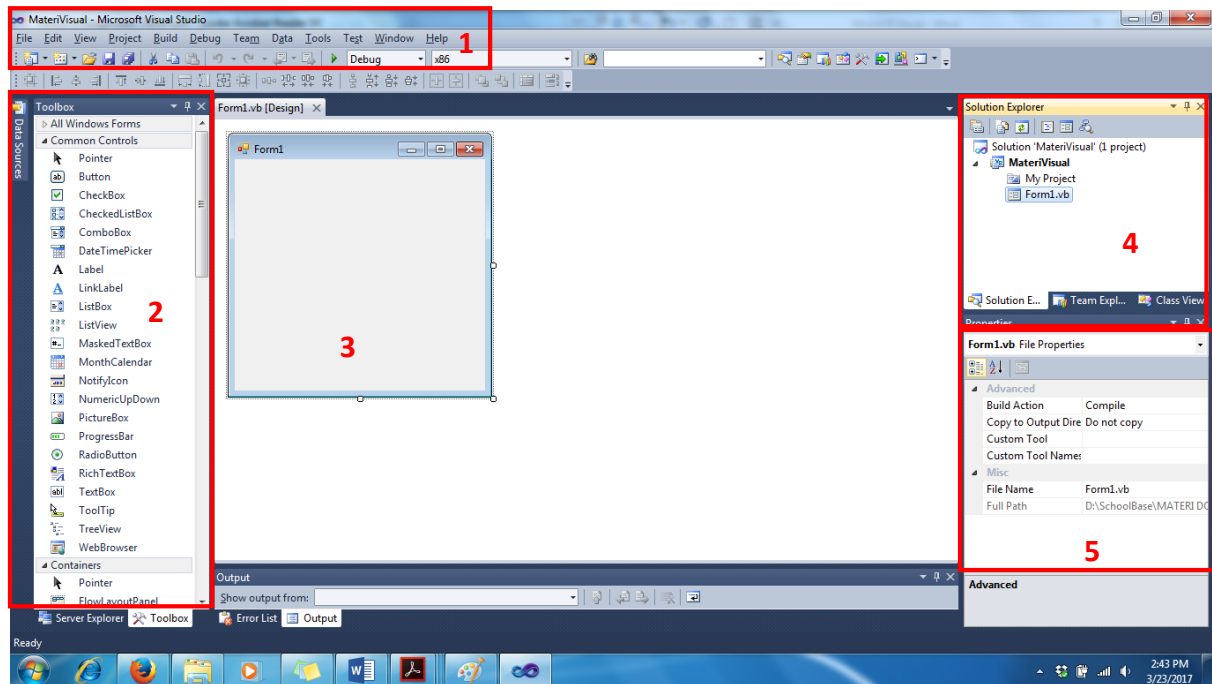


Gambar 1.3

No	Jenis Project	Macam Template
1	Windows	<ul style="list-style-type: none"> <li>• Windows Form Application</li> <li>• Class Library</li> <li>• Windows Service</li> <li>• dll</li> </ul>
2	Web	<ul style="list-style-type: none"> <li>• ASP .NET Application</li> <li>• ASP .NET Server Control</li> <li>• WCF Service Application</li> <li>• dll</li> </ul>
3	Smart Device	<ul style="list-style-type: none"> <li>• Smart Device Project</li> <li>• dll</li> </ul>
4	Database	<ul style="list-style-type: none"> <li>• SQL Server Project</li> </ul>

## 2. Mengetahui IDE Visual Basic

Dalam upaya memudahkan penggunaan IDE, maka diperlukan pemahaman yang baik mengenai IDE Visual Basic 2012



Gambar 1.4

### Keterangan:

1. Menu Bar (menu standar visual basic)
2. Toolbox (daftar control yang ditambahkan ke dalam program sebagai interface)
3. Form Design (digunakan untuk mengedit tampilan form serta mengatur posisi control pada form)
4. Solution Explorer (digunakan untuk mengolah file dan project)
5. Properties (digunakan untuk mengedit dari form dan control yang sedang diedit)

## 3. Praktikum I (Poin: 30)

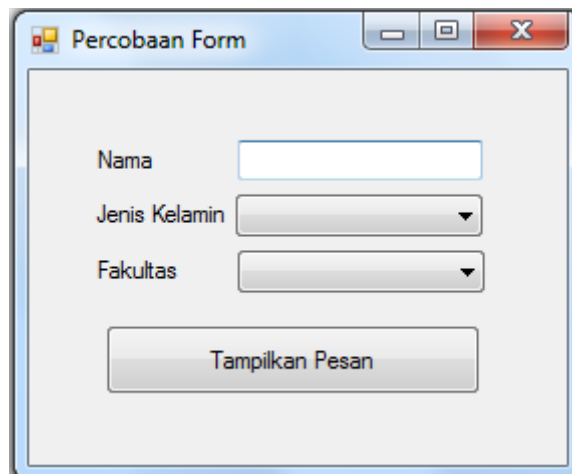
- a. Buatlah sebuah Project Baru dengan nama prj\_vbNet\_NIM
- b. Tambahkan control-control berikut ini ke dalam Form Design

No	Object	Property	Nilai
1	Form	Name Text	Frmcoba Percobaan Form
2	Label	Text	Nama
3	Label	Text	Jenis Kelamin
4	Label	Text	Fakultas
5	TextBox	Name Text	txtnama (dikosongkan)
6	ComboBox	Name Items DropDownStyle	cbojeniskelamin -Laki-Laki -Perempuan DropDownList
7	ComboBox	Name	cbofakultas



		Items	<ul style="list-style-type: none"> <li>- FAI</li> <li>- FBS</li> <li>- FIA</li> <li>- FIK</li> <li>- FT</li> </ul>
		Dropdownstyle	Dropdownlist
8	Button	Name	Btntampilkan
		Text	Tampilkan Pesan

c. Desain tampilan yang dihasilkan kurang lebih adalah sebagai berikut:



d. Sisipkan code berikut ini pada **btntampilkan**

```

btntampilkan Click
Public Class frmCoba
    Private Sub btntampilkan_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btntampilkan.Click
        MessageBox.Show(txtnama.Text & vbCrLf & cbojeniskelamin.Text & vbCrLf & cbofakultas.Text,
            "Hasil Pengisian", MessageBoxButtons.OK, MessageBoxIcon.Information)
    End Sub
End Class

```

- e. Jalankan aplikasi dengan menekan tombol F5 atau melalui ikon Start Debugging di toolbar, atau melalui menu Debug > Start Debugging
- f. Jika berhasil, maka program akan dijalankan, dan setelah melakukan pengisian pada txtnama maupun memilih pada cbojeniskelamin dan cbofakultas kemudian klik btntampilkan maka akan muncul pesan sesuai dengan apa yang telah diisikan
- g. Hasil tampilan pada program tersebut


Percobaan Form

Nama

Jenis Kelamin

Fakultas

Hasil Pengisian

 Fery Updi  
Laki-Laki  
FIK

# 2

## Variabel, Tipe Data dan Operator

### POKOK BAHASAN:

- Variabel
- Tipe Data
- Operator

### TUJUAN BELAJAR:

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami penggunaan Variabel
2. Memahami jenis-jenis Tipe Data
3. Memahami penggunaan Operator

### 1. Variabel

Variabel adalah nama atau simbol yang digunakan untuk mewakili suatu nilai. Suatu variabel mempunyai nama dan menyimpan tipe data yang merupakan jenis data variabel.

Aturan penamaan variabel adalah sebagai berikut:

- Harus dimulai dengan sebuah huruf
- Tidak lebih dari 255 karakter
- Tidak boleh sama dengan nama statement, fungsi, metode, objek, dan sebagainya yang merupakan bahasa dari Visual BASIC.
- Tidak boleh ada spasi, tanda titik(.), tanda seru(!), atau karakter @, &, \$, dan #.

Deklarasi variabel dapat dituliskan dengan urutan sebagai berikut:

```
Public <nama_variabel> As <Tipe_Data>
```

Atau

```
Dim <nama_variabel> As <Tipe_Data>
```

### Contoh :

```
Public Angka1 As Integer  
Dim Nama As String
```

### 2. Tipe Data

Tipe data adalah jenis data yang disimpan dalam variabel. Tipe data untuk Visual BASIC adalah sebagai berikut:

- **Tipe Data Numerik:** digunakan untuk menyimpan data numerik, terdiri dari:

Tipe Data	Ukuran	Range
Byte	1 byte	0 sampai 255
Integer	2 byte	-32.768 sampai 32.767
Long	4 byte	-2.147.483.648 sampai 2.147.483.647
Single	4 byte	-3,402823E38 sampai -1,401298E-45; 1,401298E-45 sampai 3,402823E38
Double	8 byte	-1.79769313486232E308 sampai -4,94065645841247E-324; 4,94065645841247E-324 sampai 1.79769313486232E308
Currency	8 byte	-922.337.203.685.477,5808 sampai 922.337.203.685.477,5807

- **Tipe Data String** : digunakan untuk menyimpan data berbentuk karakter. Panjang maksimal karakter yang dapat disimpan adalah 65.400 karakter. Penulisan data dengan tipe ini diawali dan diakhiri dengan tanda petik dua (").

**Contoh:**

```
Dim Nama As String
Nama = "Dewi"
```

- **Tipe Data Logika (Boolean)** : melakukan pengetesan logika. Data dengan tipe data ini hanya dapat bernilai benar (*True*) atau salah (*False*).

**Contoh:**

```
Dim Baru As Boolean
Baru = True
```

### 3. Konstanta

Konstanta adalah suatu nilai konstan yang tidak berubah. Seperti halnya variabel, konstanta dapat diberi nama dimana aturan penamaannya sama dengan variabel.

**Contoh:**

```
Const A = 10
```

### 4. Operator

- **Operator Pemberi Nilai**

Deklarasi pemberian nilai pada Visual BASIC = Bahasa BASIC yaitu menggunakan operator sama dengan (=).

**Contoh :**

```
a = 24
nama = "Fery Updi"
```

- **Operator Arimatika**

Operator	Operasi
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
\	Pembagian dengan hasil bilangan bulat
Mod	Sisa pembagian (Modulus)

- **Operator Boolean**

Operator	Operasi
Not	Negasi
And	Logika and
Or	Logika or
Xor	Logika xor

- **Operator Perbandingan**

Operator	Operasi
=	Sama dengan

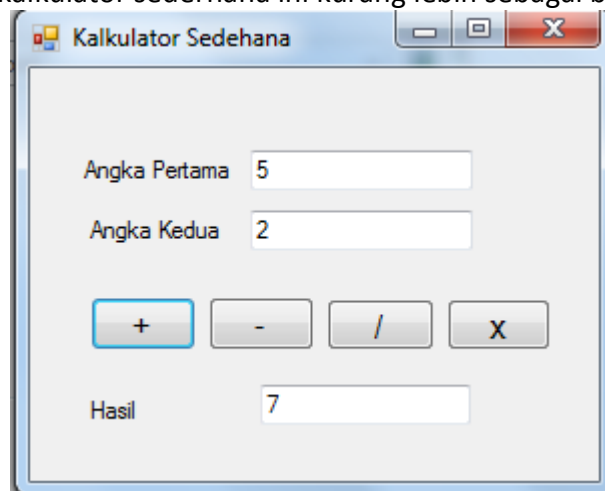
<>	Tidak sama dengan
<	Kurang dari
>	Lebih dari
<=	Kurang dari atau sama dengan
>=	Lebih dari atau sama dengan

- **Derajat Operator**

Operator	Operasi
Not	Tertinggi
* / \ mod and	
+ - or xor	
= <> <= >=	Terendah

### 5. Praktikum I (Poin: 30)

- Buatlah Form baru dengan nama frmkalkulator
- Buatlah sebuah kalkulator sederhana yang bisa melakukan operasi penambahan, pengurangan, pembagian dan perkalian antara dua buah bilangan yang diinputkan oleh user.
- Hasil program kalkulator sederhana ini kurang lebih sebagai berikut:



- Properties name diatas adalah:  
txtAngka1, txtAngka2, txtHasil, btnTambah, btnKurang, btnBagi, btnKali

## Listing Program

```
Public Class frmkalkulator
    Private Sub btnTambah_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnTambah.Click
        Dim Angka1, Angka2, Hasil As Double
        Angka1 = Val(txtAngka1.Text)
        Angka2 = Val(txtAngka2.Text)
        Hasil = Angka1 + Angka2
        txthasil.Text = Hasil
    End Sub

    Private Sub btnkali_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnkali.Click
        Dim Angka1, Angka2, Hasil As Double
        Angka1 = Val(txtAngka1.Text)
        Angka2 = Val(txtAngka2.Text)
        Hasil = Angka1 * Angka2
        txthasil.Text = Hasil
    End Sub

    Private Sub btnBagi_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnBagi.Click
        Dim Angka1, Angka2, Hasil As Double
        Angka1 = Val(txtAngka1.Text)
        Angka2 = Val(txtAngka2.Text)
        Hasil = Angka1 / Angka2
        txthasil.Text = Hasil
    End Sub

    Private Sub btnKurang_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnKurang.Click
        Dim Angka1, Angka2, Hasil As Double
        Angka1 = Val(txtAngka1.Text)
        Angka2 = Val(txtAngka2.Text)
        Hasil = Angka1 - Angka2
        txthasil.Text = Hasil
    End Sub
End Class
```

## 6. Praktikum II (Poin: 30)

- Buatlah Form baru dengan nama frmHitungNilai
- Buatlah sebuah Hitung Nilai yang bisa melakukan operasi operator aritmatika yang inputkan oleh user.
- Hasil program Hitung Nilai ini kurang lebih sebagai berikut:

Program Hitung Nilai	
Tatap Muka	89
Mid Test	78
Final Test	87
<input type="button" value="Hitung"/>	
Nilai Akhir	85

- Ketentuan :

$$\text{Nilai Akhir} = \text{Tatap Muka} + \text{MidTest} + \text{Final Test} / 3$$

**POKOK BAHASAN:**

- Statement If...Then
- Statement Select Case

**TUJUAN BELAJAR:**

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami penggunaan statement
2. Memahami penggunaan Select Case

**Statement/Pernyataan If...Then**

Pada beberapa kasus terkadang kita menginginkan komputer melakukan suatu pernyataan tertentu bila suatu kondisi terpenuhi. Dalam Visual Basic .NET perintah percabangan/pemilihan keputusan dapat dilakukan dengan statemen **If...Then** dan **Select Case**.

Ada beberapa jenis pernyataan If...Then

**1. Statemen If...Then**

Statemen ini digunakan untuk melakukan aksi setelah melakukan pengujian terhadap suatu kondisi. Pernyataan dalam blok statemen hanya akan dilaksanakan ketika kondisi pengesanan/pengujian bernilai benar.

Statement If...Then memiliki beberapa sintaks/cara penulisan sesuai dengan jumlah pernyataan yang akan dieksekusi.

**a. If...Then dengan Kondisi dan Pernyataan Tunggal**

```
If <kondisi> Then <Pernyataan>
```

**Contoh :**

```
If Nilai >= 60 Then Keterangan = "Lulus"
```

**b. If...Then dengan Pernyataan Jamak**

```
If <Kondisi> Then  
<Pernyataan_1>  
<Pernyataan_2>  
..  
<Pernyataan_n>  
End If
```

**Contoh :**

```
If Nilai >= 60 Then  
    Keterangan = "Lulus"  
    Ucapan = "Selamat"  
End If
```

c. If...Then dengan 2 kondisi.

```
If <Kondisi> Then
    <Pernyataan_Jika_Kondisi_Benar>
Else
    <Pernyataan_Jika_Kondisi_Salah>
End If
```

**Contoh:**

```
If Nilai >= 60 Then
    Keterangan = "Lulus"
    Ucapan = "Selamat"
Else
    Keterangan = "Tidak Lulus"
    Ucapan = "Jangan Bersedih"
End If
```

d. If...Then dengan kondisi jamak.

```
If <Kondisi_1> Then
    <Pernyataan>
ElseIf <Kondisi_2> Then
    <Pernyataan>
...
ElseIf <Kondisi_n> Then
    <Pernyataan>
Else
    <Pernyataan>
End If
```

**Contoh:**

```
If Nilai >= 85 Then
    Keterangan = "Lulus, Sangat Memuaskan"
ElseIf Nilai >= 70 Then
    Keterangan = "Lulus, Memuaskan"
ElseIf Nilai >=60 Then
    Keterangan = "Lulus, Cukup Memuaskan"
Else
    Keterangan = "Tidak Lulus, Silahkan Mengulang"
End If
```

## 2. Statemen Select Case

Sama halnya seperti statemen If...Then, Select Case juga mengerjakan suatu blok statemen berdasarkan uji nilai ekspresi. Perbedaannya adalah pada tata cara penulisan dan pengelompokan nilai dari variabel/kondisi.

**Sintaks :**



```
Select Case <Variabel penguji>  
  Case <Nilai_1>  
    <Pernyataan_1>  
  Case <Nilai_2>  
    <Pernyataan_2>  
  Case Else  
    <Pernyataan_n>
```

End Select

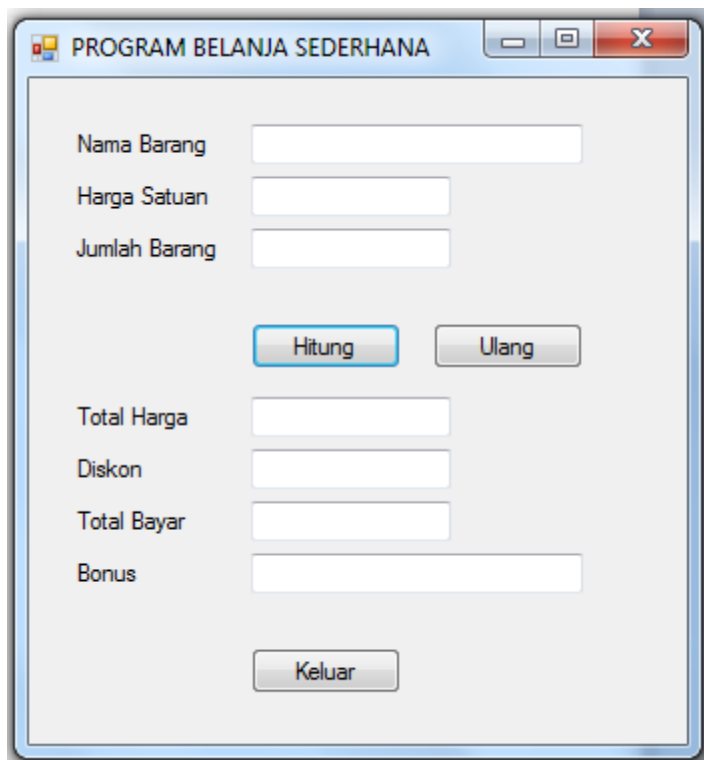
**Contoh:**

```
Select Case Nilai  
  Case "A"  
    Keterangan = "Sangat Memuaskan"  
  Case "B"  
    Keterangan = "Memuaskan"  
  Case "C"  
    Keterangan = "Cukup"  
  Case Else  
    Keterangan = "Kurang"
```

End Select

### 3. Praktikum I (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar



The image shows a screenshot of a Windows application window titled "PROGRAM BELANJA SEDERHANA". The window contains a form with the following elements:

- Input field for "Nama Barang"
- Input field for "Harga Satuan"
- Input field for "Jumlah Barang"
- Two buttons: "Hitung" and "Ulang"
- Input field for "Total Harga"
- Input field for "Diskon"
- Input field for "Total Bayar"
- Input field for "Bonus"
- A button labeled "Keluar" at the bottom center.

Pada program diatas perhitungan dilakukan dengan kriteria sebagai berikut:

**Total Harga = Harga Satuan x Jumlah Barang**

Diskon dan Bonus, didapat dengan ketentuan:

Total Harga	Diskon	Bonus
>=500 ribu	20% x Total Harga	Tas Pinggang
200 ribu – 500 ribu	15% x Total Harga	Payung
100 ribu – 200 ribu	10% x Total Harga	Kaos
50 ribu – 100 ribu	5% x Total Harga	Cangkir
< 50 ribu	0%	Tidak Ada

**Total Bayar = Total Harga – Diskon**

### Kode Program

```
Public Class ProgramBelanja
    Private Sub btnUlang_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnUlang.Click
        'Membersihkan isi Textbox
        txtNama.Text = ""
        txtHarga.Text = ""
        txtJumlah.Text = ""
        txtTotal.Text = ""
        txtDiskon.Text = ""
        txtBayar.Text = ""
        txtBonus.Text = ""
        txtNama.Focus()
    End Sub
    Private Sub btnHitung_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnHitung.Click
        'Deklarasi Variabel
        Dim Harga, Jumlah As Integer
        Dim Total, Diskon, Bayar As Double
        Dim Bonus As String
        'Deklarasi Input Harga dan Jumlah
        Harga = txtHarga.Text
        Jumlah = txtJumlah.Text
        'Perhitungan Total Harga
        Total = Harga * Jumlah
        'Penentuan Diskon dan Bonus
        If Total >= 500000 Then
            Diskon = 0.2 * Total
            Bonus = "Tas Pinggang"
        ElseIf Total >= 200000 Then
            Diskon = 0.15 * Total
            Bonus = "Payung"
        ElseIf Total >= 100000 Then
            Diskon = 0.1 * Total
            Bonus = "Kaos"
        ElseIf Total >= 50000 Then
            Diskon = 0.05 * Total
            Bonus = "Pena"
        Else
            Diskon = 0
            Bonus = "Tidak Ada"
        End If
        'Perhitungan Total Bayar
        Bayar = Total - Diskon
        'Deklarasi Keluaran Total Harga, Diskon, Total Bayar, dan Bonus
        txtTotal.Text = Total
        txtDiskon.Text = Diskon
        txtBayar.Text = Bayar
        txtBonus.Text = Bonus
    End Sub

    Private Sub btnKeluar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles btnKeluar.Click
        End
    End Sub
End Class
```

#### 4. Praktikum I (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar

The screenshot shows a Windows application window titled "Order Makanan". It features a light gray background with a white border. At the top, there are standard window controls (minimize, maximize, close). The main content area contains the following elements:

- A dropdown menu with "Nasi Uduk" selected, followed by the label "Harga" and the price "Rp. 3500".
- A checked checkbox with the label "Termasuk dengan minuman".
- Another dropdown menu with "Es Jeruk" selected, followed by the label "Harga" and the price "Rp. 3000".
- A rounded rectangular box titled "Menu Pesanan" containing:
  - A label "Berapa porsi makanan" next to a text box containing the number "1".
  - A label "Berapa porsi minuman" next to a text box containing the number "1".
  - A "Hitung" button.
  - The total price "Rp. 6500" displayed to the right of the button.

Kode Program:

```
Public Class Ordermakanan
    Public hrgmakanan, hrgminuman As Integer

    Private Sub ComboBox1_SelectedIndexChanged(ByVal sender As System.Object,
        Select Case ComboBox1.Text
            Case "Nasi Uduk"
                Label2.Text = "Rp. 3500"
                hrgmakanan = 3500
            Case "Bakso"
                Label2.Text = "Rp. 6000"
                hrgmakanan = 6000
            Case "Mie Ayam"
                Label2.Text = "Rp. 7000"
                hrgmakanan = 7000
        End Select
    End Sub
End Class
```

```

1 Private Sub ComboBox2_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.
    Select Case ComboBox2.Text
        Case "Es Jeruk"
            Label4.Text = "Rp. 3000"
            hrgminuman = 3000
        Case "Jus Alpukat"
            Label4.Text = "Rp. 7000"
            hrgminuman = 7000
        Case "Teh Manis"
            Label4.Text = "Rp. 4000"
            hrgminuman = 4000
    End Select
End Sub

```

---

```

1 Public Sub New()

    ' This call is required by the designer.
    InitializeComponent()
    ComboBox2.Visible = False
    Label3.Visible = False
    ' Add any initialization after the InitializeComponent() call.

End Sub
1 Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object, ByVal e As System.
    If CheckBox1.Enabled = False Then
        ComboBox2.Visible = False
    End If

    If CheckBox1.Enabled = True Then
        ComboBox2.Visible = True
        Label3.Visible = True
    End If
End Sub

```

---

```

1 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim porsimkn, porsimmm, jumlah As Integer
    porsimkn = TextBox1.Text
    porsimmm = TextBox2.Text
    jumlah = (hrgmakanan * porsimkn) + (hrgminuman * porsimmm)
    Label7.Text = "Rp. " & jumlah.ToString
End Sub
End Class

```

---

# 4

## Perulangan

### POKOK BAHASAN:

- For...Next
- Do...Loop

### TUJUAN BELAJAR:

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami penggunaan For...Next
2. Memahami penggunaan Do...Loop

Proses perulangan dalam pemrograman dilakukan untuk mengerjakan suatu proses operasi secara bertahap demi tahap dengan nilai variabel yang menaik atau menurun.

Dalam Visual Basic .NET proses perulangan dapat dilakukan dengan beberapa statemen, diantaranya adalah statemen For...Next dan Do...Loop.

#### 1. For...Next

Statemen ini akan mengulangi suatu blok pernyataan sebanyak jumlah yang ditentukan. Statemen ini digunakan jika banyaknya jumlah perulangan sudah diketahui.

##### Sintaks:

```
For <Variabel_Pengulang> = NilaiAwal To NilaiAkhir [Step Tingkat]
<Pernyataan_1>
...
<Pernyataan_n>
Next <Variabel_Pengulang>
```

Statemen ini digunakan untuk kondisi yang mempunyai nilai berurutan dan variable yang mempunyai nilai numerik.

Default untuk Step adalah 1, jadi untuk perulangan dengan urutan menaik 1, nilai step tidak perlu ditulis. Sedangkan untuk perulangan menurun (Nilai awal > Nilai Akhir), nilai step diawali dengan tanda minus(-).

Misalnya : For i = 10 To 1 Step -1.

##### Contoh:

Untuk mencetak angka 1 sampai 10 secara berurutan pada objek ListBox dapat dilakukan dengan memberi listing program sebagai berikut:

```
For i = 1 To 10
List1.AddItem i
Next i
```

#### 2. Do...Loop

Statemen ini mengulang blok statemen bila kondisi benar atau sampai kondisi menjadi benar. Bila tidak ada perintah keluar, proses perulangan (*loop*) akan terus berlangsung. Statemen ini digunakan untuk kondisi yang mempunyai nilai tidak pasti dan tidak berurutan.

Statemen ini memiliki dua buah bentuk logika.

**a. Statemen Do...Loop...While**

Statemen ini akan mengerjakan pernyataan dalam blok statemen ketika kondisi bernilai benar, dan akan berhenti ketika kondisi sudah bernilai salah.

**Sintaks:**

<b>Do While &lt;Kondisi&gt;</b> <Pernyataan_1> ... <Pernyataan_n> <b>Loop</b>	<i>atau</i>	<b>Do</b> <Pernyataan_1> ... <Pernyataan_n> <b>Loop While &lt;Kondisi&gt;</b>
---	-------------	---

**Contoh:**

Untuk mencetak angka 1 sampai 10 secara berurutan pada objek ListBox dapat dilakukan dengan memberi listing program sebagai berikut:

```
i = 1
Do While i <= 10
    List1.AddItem i
    i = i + 1
Loop
```

**b. Statemen Do...Loop...Until**

Statemen ini akan mengerjakan pernyataan dalam blok statemen ketika kondisi bernilai salah, dan akan berhenti ketika kondisi mencapai nilai benar.

**Sintaks:**

<b>Do Until &lt;Kondisi&gt;</b> <Pernyataan_1> ... <Pernyataan_n> <b>Loop</b>	<i>atau</i>	<b>Do</b> <Pernyataan_1> ... <Pernyataan_n> <b>Loop Until &lt;Kondisi&gt;</b>
---	-------------	---

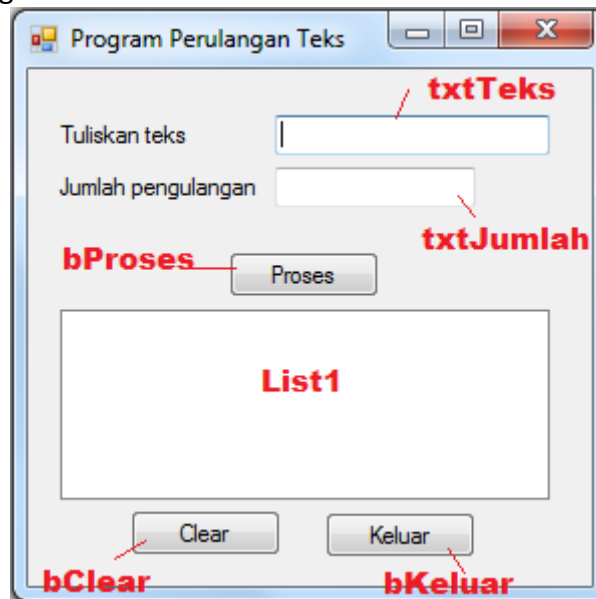
**Contoh:**

Untuk mencetak angka 1 sampai 10 secara berurutan pada objek ListBox dapat dilakukan dengan memberi listing program sebagai berikut:

```
i = 1
Do
    List1.AddItem i
    i = i + 1
Loop Until i > 10
```

### 3. Praktikum I (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar



#### Kode Program:

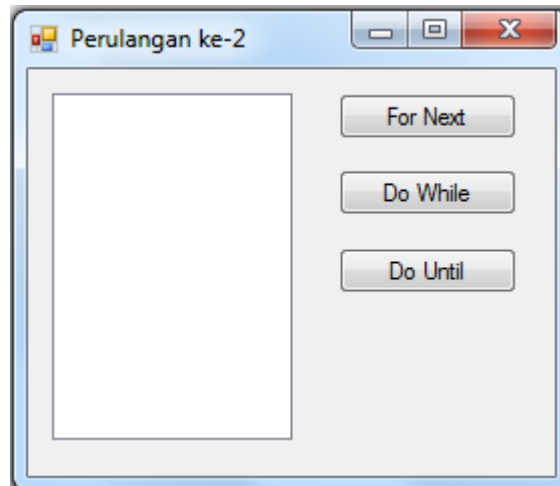
```
Public Class Perulangan
    Private Sub bKeluar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bKeluar.Click
        End
    End Sub

    Private Sub bProses_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bProses.Click
        Dim teks As String
        Dim jumlah As Integer
        teks = txtTeks.Text
        jumlah = txtJumlah.Text
        For i = 1 To jumlah
            List1.Items.Add(teks)
        Next i
    End Sub

    Private Sub bClear_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bClear.Click
        txtTeks.Text = ""
        txtJumlah.Text = ""
        List1.Items.Clear()
    End Sub
End Class
```

#### 4. Praktikum II (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar



#### Kode Program:

```
Public Class Perulangan2
    Private Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click
        ListBox1.Items.Clear()
        For i = 1 To 10
            ListBox1.Items.Add("For Next : " & i)
        Next
    End Sub

    Private Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click
        ListBox1.Items.Clear()
        Dim i As Integer = 0
        Do While i <= 10
            ListBox1.Items.Add("Do While : " & i)
            i = i + 1
        Loop
    End Sub

    Private Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button3.Click
        ListBox1.Items.Clear()
        Dim i As Integer = 0
        Do Until i > 10
            ListBox1.Items.Add("Do Until : " & i)
            i = i + 1
        Loop
    End Sub
End Class
```



**POKOK BAHASAN:**

- Array satu dimensi
- Array multi dimensi

**TUJUAN BELAJAR:**

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami dan mengenal Array (array satu dimensi dan array multi dimensi)
2. Dapat membuat aplikasi sederhana dengan menggunakan array satu dimensi dan array multi dimensi

**Dasar Teori**

Array adalah sekumpulan data yang memiliki tipe yang sama, sejumlah tetap, serta disusun secara terstruktur dan disimpan dalam satu variabel yang sama, dan diurutkan dengan index.

**1. Array Satu Dimensi**

Suatu array yang nilai dan ukurannya yang sudah ditentukan terlebih dahulu, dan memiliki satu dimensi.

Bentuk Umum :

**Dim Array[indeks] As Tipe\_Data**

Contoh :

```
Dim Arr(2) As String
Arr(0) = TxtNim.Text
Arr(1) = TxtNama.Text
Arr(2) = TxtProdi.Text
```

**2. Array Multi Dimensi**

Suatu array yang fungsinya hampir sama dengan array satu dimensi hanya saja pada array multi dimensi ini mewakili nilai table yang terdiri dari informasi yang diatur dalam baris dan kolom. Untuk mendefinisikan elemen table tertentu, kita harus menentukan dua indeks, pertama mengidentifikasi elemen baris dan yang mengidentifikasi elemen kolom. Array multidimensi memiliki lebih dari dua dimensi.

Bentuk Umum :

**Dim Array[indeks,Indeks] As Tipe\_Data**  
**Dim Array[indeks,indeks,indeks] As Tipe\_Data**

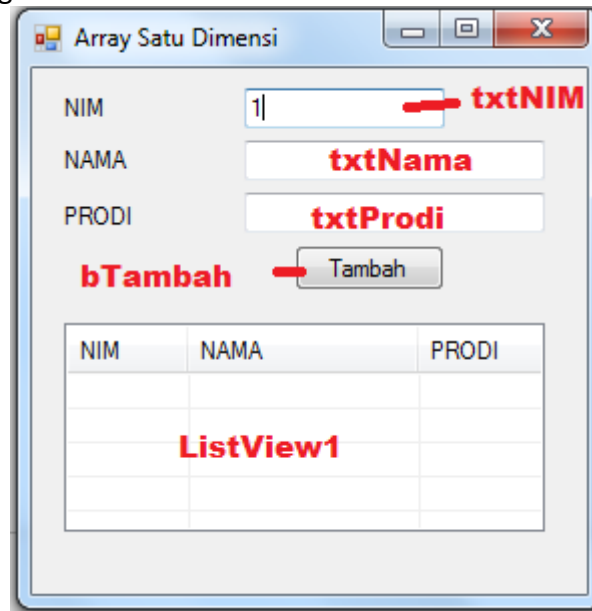
Contoh :

```
Dim Arr(3, 1) As String
Arr(0, 0) = "NIM"
Arr(0, 1) = "NAMA"
Arr(1, 0) = "JENIS KELAMIN"
Arr(1, 1) = "PRODI"
```

Arr(2, 0) = "Laki-laki"  
 Arr(2, 1) = "Perempuan"  
 Arr(3, 0) = "Manajemen Informatika"  
 Arr(3, 1) = "Teknik Informatika"

### 3. Praktikum I Array Satu Dimensi (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar



#### Kode Program:

```

Public Class Array1dimensi
    Private Sub Array1dimensi_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        ListView1.GridLines = True
        ListView1.View = View.Details
        ListView1.Columns.Add("NIM")
        ListView1.Columns.Add("NAMA", 115)
        ListView1.Columns.Add("PRODI", 110)
        txtNIM.Text = 1
        txtNama.Focus()
    End Sub

    Private Sub bTambah_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bTambah.Click
        Dim Arr(2) As String
        Arr(0) = txtNIM.Text
        Arr(1) = txtNama.Text
        Arr(2) = txtProdi.Text

        Dim listitem As ListViewItem
        listitem = New ListViewItem
        listitem = ListView1.Items.Add(Arr(0))
        listitem.SubItems.Add(Arr(1))
        listitem.SubItems.Add(Arr(2))

        txtNIM.Text = txtNIM.Text + 1
        txtNama.Text = ""
        txtProdi.Text = ""
        txtNama.Focus()
    End Sub
End Class
  
```

#### 4. Praktikum II Array Multi Dimensi (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar

NIM	NAMA	JENIS KE

#### Kode Program:

```
Public Class ArrayMultiDimensi
| Private Sub ArrayMultiDimensi_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
    Dim Arr(3, 1) As String
    Arr(0, 0) = "NIM"
    Arr(0, 1) = "NAMA"
    Arr(1, 0) = "JENIS KELAMIN"
    Arr(1, 1) = "PRODI"
    Arr(2, 0) = "Laki-laki"
    Arr(2, 1) = "Perempuan"
    Arr(3, 0) = "Sistem Informasi"
    Arr(3, 1) = "Komputerisasi Akuntansi"

    ListView1.GridLines = True
    ListView1.View = View.Details

    For Baris = 0 To 1
        For Kolom = 0 To 1
            ListView1.Columns.Add(Arr(Baris, Kolom), 100)
        Next Kolom
    Next Baris

    For Baris = 2 To 2
        For Kolom = 0 To 1
            cbKelamin.Items.Add(Arr(Baris, Kolom))
        Next Kolom
    Next Baris
End Class
```

```

        For Baris = 3 To 3
            For Kolom = 0 To 1
                cbProdi.Items.Add(Arr(Baris, Kolom))
            Next Kolom
        Next Baris
    End Sub
} Private Sub bProses_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bProses.Click
    Dim Arr(3) As String
    Arr(0) = txtNIM.Text
    Arr(1) = txtNama.Text
    Arr(2) = cbKelamin.Text
    Arr(3) = cbProdi.Text

    Dim listitem As ListViewItem
    listitem = New ListViewItem
    listitem = ListView1.Items.Add(Arr(0))
    listitem.SubItems.Add(Arr(1))
    listitem.SubItems.Add(Arr(2))
    listitem.SubItems.Add(Arr(3))
    txtNIM.Text = txtNIM.Text + 1
    txtNama.Text = ""
    cbKelamin.Text = ""
    cbProdi.Text = ""
    txtNama.Focus()
End Sub
End Class

```

# 6

## Procedure, Function dan Module

### POKOK BAHASAN:

- Procedure
- Function
- Module

### TUJUAN BELAJAR:

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami dan mengenal Procedure, Function dan Module
2. Dapat membuat aplikasi sederhana dengan menggunakan Procedure, Function dan Module

### Dasar Teori

#### 1. Procedure

Sejauh ini, anda telah menuliskan sub procedure yang dibuat secara otomatis pada saat menggunakan event . Sub procedure merupakan blok kode yang mempunyai nama dan berisi perintah yang dapat dipanggil suatu waktu. Artinya, hanya sekali dibuat yang kemudian di panggil berkali-kali sesuai dengan kebutuhan.

Bentuk Umum :

```
[Private | Public] Sub subname[(argumentlist)]  
Statements  
End sub
```

Sub Procedure dapat dipanggil dengan perintah :

```
[call] subname[(argumentlist)]
```

Keterangan :

Sub procedure (*Procedure*) sebaiknya menggunakan kata kerja (verb)

Pemanggilan procedure yang masih dalam satu kelas (class) atau modul (module) dapat dilakukan dengan kata kunci **Me**.

Procedure dapat menerima argument melalui nilai (by value) atau juga reference (by reference). Jika menggunakan by value, nilai pada argument tidak berubah. Sedangkan jika menggunakan by reference, nilai argument yang digunakan dapat berubah sesuai dengan perintah dalam procedure.

Secara default, argument dimasukkan melalui nilai (by value) dapat menggunakan kata kunci ByVal dan ByRef untuk By Reference.

Syntaks argument :

```
[ByVal | ByRef] variabelname as type
```

Contoh :

1. Procedure dengan argument di passing sebagai value

```
Sub HitungVal(byVal as Decimal)
```

```
  A+=1
```

```
  textAakhir.text=a
```

```
end sub
```

2. Procedure dengan argument di passing sebagai value

```
Sub HitungRef(byRef as Decimal)
```

```
  A+=1
```

```
  textAakhir.text=a
```

```
end sub
```

3. pemanggilan procedure

```
HitungVal(textAwal.Text)
```

## 2. Function (Fungsi)

Fungsi mempunyai banyak kemiripan dengan sub procedure. Bedanya, fungsi selalu mengembalikan nilai (return value)

Sintaks fungsi dituliskan sebagai berikut :

```
[Private | Public] function functionname[(argumentlist)] [As type]
```

```
Statement
```

```
Functionname==expression | {return expression}
```

```
End function
```

Keterangan :

- Perbedaan antara fungsi dengan procedure hanyalah pada pengembalian nilai saja (return value). Sebagai contoh :  
Menggunakan argument berupa nilai (by value) dan reference (by reference)
- Pengembalian nilai dapat dilakukan dengan menggunakan operator assignment "=" pada fungsi atau biasanya dengan perintah return.

### Contoh :

```
Private Function cekData() as Boolean
```

```
  if IsNumeric(TextAwal.text) then
```

```
    return true
```

```
  else
```

```
    return false
```

```
  endif
```

```
end function
```

### Pemanggilan fungsi

```
if cekData()=true then
```

```
  HitungRef(textAwal.text)
```

```
else
```

```
  MessageBox.Show("Data yang dimasukkan bukan angka","Kesalahan data",
```

```
MessageBoxButtons.OK,  
    MessageBoxIcon.Error)  
endif
```

### 3. Modul (Module)

Modul merupakan bagian yang sengaja dipisahkan untuk memudahkan pemrograman. Dalam modul dapat dimasukkan procedure dan fungsi dan kemudian digunakan oleh beberapa form.

- Umumnya modul dideklarasikan secara public (public) agar dapat digunakan di dalam kode dimanapun
- Modul dapat diisi dengan main procedure yang akan dijalankan pertama kali saat program dimulai
- Modul merupakan salah satu argumentasi code reuse yang bertujuan agar program lebih efisien.

Sintaks pembuatan module sebagai berikut :

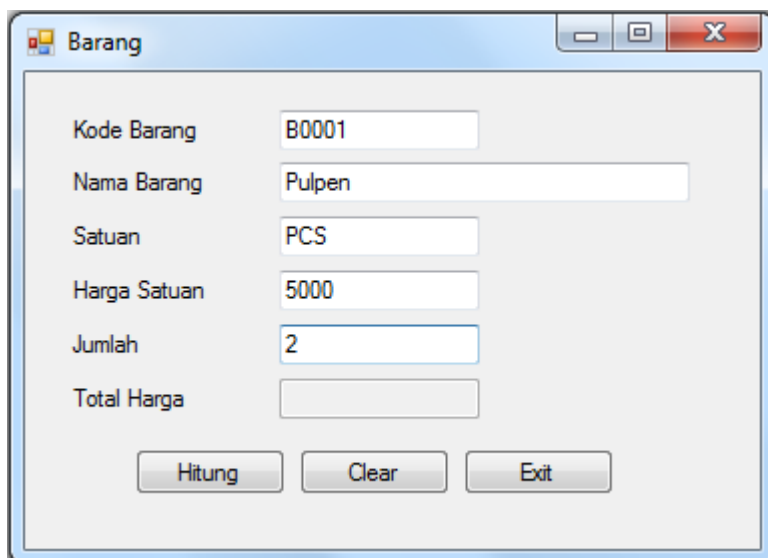
```
Module moduleName  
    Statement  
End Module
```

Contoh:

```
Module Module  
    Public intResponse, RoleId, Msg As Integer  
    Public Username, Password, Nama As String  
End Module
```

### 1. Praktikum I Procedure dan Function (Poin: 30)

Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar



The image shows a screenshot of a Windows application window titled "Barang". The window contains a form with the following fields and buttons:

Kode Barang	<input type="text" value="B0001"/>
Nama Barang	<input type="text" value="Pulpen"/>
Satuan	<input type="text" value="PCS"/>
Harga Satuan	<input type="text" value="5000"/>
Jumlah	<input type="text" value="2"/>
Total Harga	<input type="text"/>

At the bottom of the form, there are three buttons: "Hitung", "Clear", and "Exit".

#### Properties Name

txtKodeBrg, txtNamaBrg, txtSatuan, txtHrgSatuan, txtJumlah, txtTotal, bHitung, bClear, bExit

## Kode Program

```
Public Class ProcedureFunction
    Sub Bersih()
        txtKodeBrg.Text = ""
        txtNamaBrg.Text = ""
        txtSatuan.Text = ""
        txtHrgsatuan.Text = ""
        txtJumlah.Text = ""
        txtTotal.Text = ""
        TxtKodeBrg.Focus()
        txtTotal.Enabled = False
    End Sub

    Private Sub ProcedureFunction_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        Call Bersih()
    End Sub

    Private Sub bClear_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bClear.Click
        Call Bersih()
    End Sub

    Private Function Hitung()
        Dim Jumlah, HrgSatuan, Total As Integer
        Jumlah = txtJumlah.Text
        HrgSatuan = TxtHrgsatuan.Text
        Total = Jumlah * HrgSatuan
        txtTotal.Text = Total
        Return Total
    End Function

    Sub CekDataKosong()
        If txtKodeBrg.Text = "" Then
            MessageBox.Show("Kode Barang harus diisi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            txtKodeBrg.Focus()
        ElseIf txtNamaBrg.Text = "" Then
            MessageBox.Show("Nama Barang harus diisi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            txtNamaBrg.Focus()
        ElseIf txtSatuan.Text = "" Then
            MessageBox.Show("Satuan Barang harus diisi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            txtSatuan.Focus()
        ElseIf txtHrgsatuan.Text = "" Then
            MessageBox.Show("Harga Satuan Barang harus diisi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            txtHrgsatuan.Focus()
        ElseIf txtJumlah.Text = "" Then
            MessageBox.Show("Jumlah Barang harus diisi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            txtJumlah.Focus()
        Else
            Call Hitung()
        End If
    End Sub

    Private Sub bHitung_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bHitung.Click
        Call CekDataKosong()
    End Sub

    Private Sub bExit_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles bExit.Click
        Dim Tutup As String
        Tutup = MessageBox.Show("Yakin tutup form ini ?", "Konfirmasi",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question)
        If Tutup = MsgBoxResult.Yes Then
            End
        Else
            Exit Sub
        End If
    End Sub
End Class
```



**POKOK BAHASAN:**

- Database

**TUJUAN BELAJAR:**

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami dan mengenal database
2. Dapat membuat database Ms.Access dan SQL Server
3. Dapat membuat aplikasi database dengan akses data lewat komponen ADO.Net

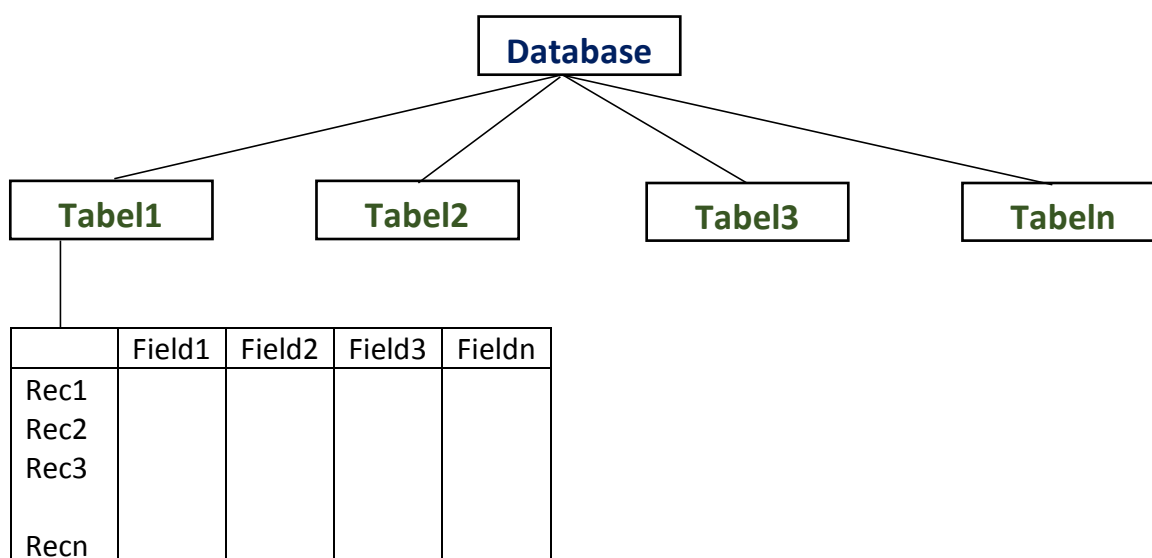
**1. Dasar Teori**

Database adalah informasi yang tersimpan dan tersusun rapi di dalam suatu tempat, dan dapat dengan mudah dimanipulasi seperti menambah data, menghapus, mencari, mengatur informasi yang kita butuhkan

**2. Komponen Database**

Komponen terbesar dalam suatu Sistem Database adalah Database itu sendiri. Data dalam database dapat dikelompokkan dalam beberapa tabel. **Tabel** merupakan kumpulan data sejenis. Sebuah tabel biasanya terdiri dari beberapa field (kolom) dan record (baris), sebuah **field** menerangkan sebuah informasi dalam tentang identitas data dalam tabel, sedangkan **record** menerangkan sebuah data dalam tabel secara lengkap.

Hirarki dari sebuah database dapat dilihat pada gambar 7.1



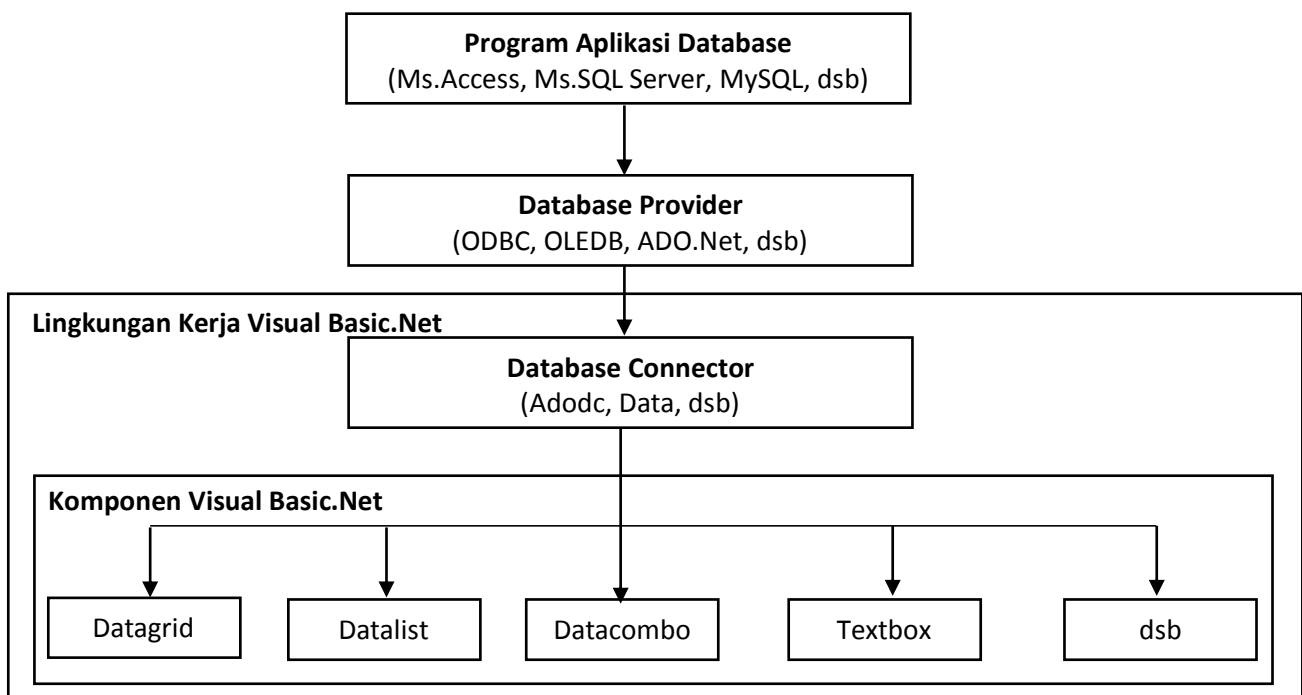
Gambar 7.1 Hirarki Database

### 3. Mengenal Database Management System

*Database Management System* atau yang biasa disingkat dengan DBMS merupakan perangkat lunak atau program komputer yang dirancang secara khusus untuk memudahkan pengelolaan database. Salah satu macam DBMS yang populer dewasa ini berupa RDBMS (*Relational DataBase Management System*), yang menggunakan model basis data relasional atau dalam bentuk tabel-tabel yang saling terhubung. Microsoft Access, Microsoft SQL Server dan MySQL merupakan contoh produk RDBMS.

### 4. Konsep Dasar Pemrograman Database

Pemrograman Database (*Database Programming*) merupakan suatu bentuk pemrograman alternatif untuk pengolahan database. Dengan pemrograman database kita dapat secara leluasa mengatur tampilan dan alur kerja sebuah database dengan lebih baik. Visual BASIC.Net merupakan salah satu bahasa pemrograman yang telah mendukung pemrograman database. Visual BASIC.Net dapat dihubungkan dengan program aplikasi pengolah data lain seperti Access, MySQL, SQL Server dan sebagainya. Alur kerja pemrograman database dalam Visual BASIC.Net dapat dijelaskan melalui gambar 7.2.



Gambar 7.2 Alur kerja pemrograman database dalam Visual Basic.Net

### 5. Koneksi Visual Basic.Net dengan Database (DBMS).

Untuk dapat menghubungkan Visual Basic.Net dengan database, kita akan menggunakan komponen ADO Data Control (ADODC). Komponen ini dapat dihubungkan dengan beberapa komponen yang digunakan untuk mengakses data seperti textbox, datagrid, dsb.

#### Data Provider

Data provider bertanggung jawab untuk menyediakan dan menghubungkan koneksi ke database. NET Framework saat ini dilengkapi dengan dua DataProvider yaitu:

- SQL Data Provider yang dirancang hanya untuk bekerja dengan SQL Server

- Daya Provider OLEDB yang memungkinkan untuk terhubung ke database jenis lain seperti Access, MySQL dan Oracle.

Setiap Data Provider terdiri dari kelas komponen berikut:

- Object Connection yang menyediakan koneksi ke database
- Object Command yang digunakan untuk mengeksekusi perintah
- Object DataReader yang menyediakan fungsi forward-only, read-only, recordset.

## 1. Praktikum I Database (Poin: 30)

### Membuat Database (Microsoft Access)

Langkah-langkah dalam membuat database adalah sebagai berikut :

1. Buatlah database dengan nama **JualBuku.accdb** lalu simpan kedalam folder **Project Anda > bin > debug** kemudian rancanglah struktur tabel seperti berikut ini :

Nama Tabel : **Jenis**

Field	Type	Size	Keterangan
KodeJenis	Text	2	Primary Key (PK)
Jenis	Text	50	

2. Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar

The image shows a Windows application window titled "Data Jenis Buku". Inside the window, there are two text input fields: "Kode Jenis" (labeled TextBox1) and "Nama Jenis" (labeled TextBox2). Below these fields are four buttons: "Simpan", "Ubah", "Hapus", and "Batal". Underneath the buttons is another text input field labeled "TextBox3". At the bottom of the form is a large gray rectangular area labeled "DataGridView1", which is currently empty.

3. Tambahkan sebuah module (klik kanan **Project > Add > Module**) kemudian tuliskan kode program dibawah ini :

```
Imports System.Data.OleDb
Module Module1
    Public Conn As OleDbConnection
    Public da As OleDbDataAdapter
    Public ds As DataSet
    Public cmd As OleDbCommand
    Public rd As OleDbDataReader
    Public Str As String
    Public Sub Koneksi()
        Str = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" & Application.StartupPath &
            "\JualBuku.accdb"
        Conn = New OleDbConnection(Str)
        If Conn.State = ConnectionState.Closed Then
            Conn.Open()
        End If
    End Sub
End Module
```

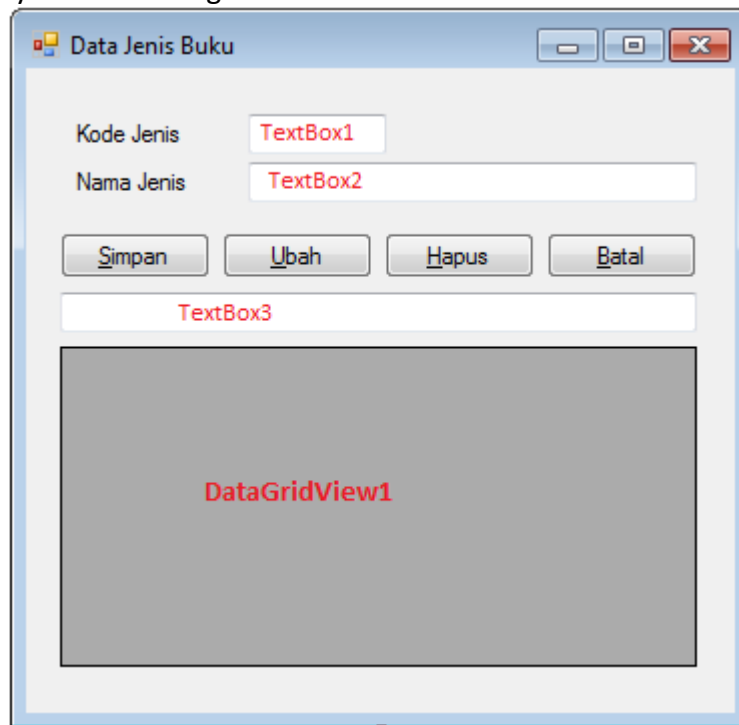
4. Kemudian tuliskan kode program pada Form Jenis dibawah ini :

```
Imports System.Data.OleDb
Public Class DataJenis
    Sub Kosong()
        TextBox1.Clear()
        TextBox2.Clear()
        TextBox1.Focus()
    End Sub
    Sub Isi()
        TextBox2.Clear()
        TextBox2.Focus()
    End Sub
    Sub TampilJenis()
        da = New OleDbDataAdapter("Select * From Jenis", Conn)
        ds = New DataSet
        ds.Clear()
        da.Fill(ds, "Jenis")
        DataGridView1.DataSource = ds.Tables("Jenis")
        DataGridView1.Refresh()
    End Sub
    Sub AturGrid()
        DataGridView1.Columns(0).Width = 60
        DataGridView1.Columns(1).Width = 200
        DataGridView1.Columns(0).HeaderText = "KODE JENIS"
        DataGridView1.Columns(1).HeaderText = "NAMA JENIS"
    End Sub
    Private Sub DataJenis_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        Call Koneksi()
        Call TampilJenis()
        Call Kosong()
        Call AturGrid()
    End Sub
End Class
```

## 2. Praktikum II Database (Poin: 30)

Mengembangkan Form pada praktikum I mengenai pembuatan form Data Jenis Buku dengan menambahkan fungsi Simpan, Ubah, Hapus, dan Batal.

Langkah-langkahnya adalah sebagai berikut:



Tambahkan kode program berikut ini:

### 5. Kode program button1 untuk **Simpan**

```
Private Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click
    If TextBox1.Text = "" Or TextBox2.Text = "" Then
        MsgBox("Data belum lengkap..!")
        TextBox1.Focus()
        Exit Sub
    Else
        cmd = New OleDbCommand("Select * From Jenis where KodeJenis='" & TextBox1.Text & "'", Conn)
        rd = cmd.ExecuteReader
        rd.Read()
        If Not rd.HasRows Then
            Dim Simpan As String = "insert into Jenis(KodeJenis,Jenis)values " & _
                "(" & TextBox1.Text & "','" & TextBox2.Text & "'"
            cmd = New OleDbCommand(Simpan, Conn)
            cmd.ExecuteNonQuery()
            MsgBox("Simpan data sukses...!", MsgBoxStyle.Information, "Perhatian")
        End If
        Call TampilJenis()
        Call Kosong()
        TextBox1.Focus()
    End If
End Sub
```

### 6. Kode program TextBox2 menentukan maksimum pengisian data Jenis Buku dan merubah huruf kecil menjadi huruf kapital jika menekan tombol Enter.

```
Private Sub TextBox2_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles TextBox2.
    TextBox2.MaxLength = 50
    If e.KeyChar = Chr(13) Then
        TextBox2.Text = UCase(TextBox2.Text)
    End If
End Sub
```

7. Kode program DataGridView1 pada saat Tabel DataGridView1 diklik maka data akan langsung terisi ke TextBox1 dan TextBox2.

```
Private Sub DataGridView1_CellContentClick(ByVal sender As Object, ByVal e As System.Windows.Forms.DataGridView
    Dim i As Integer
    i = Me.DataGridView1.CurrentRow.Index
    With DataGridView1.Rows.Item(i)
        Me.TextBox1.Text = .Cells(0).Value
        Me.TextBox2.Text = .Cells(1).Value
    End With
End Sub
```

8. Kode program button2 untuk **Ubah**

```
Private Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click
    If TextBox1.Text = "" Then
        MsgBox("Kode Jenis belum diisi")
        TextBox1.Focus()
        Exit Sub
    Else
        Dim Ubah As String = "Update Jenis set " & _
            "Jenis='" & TextBox2.Text & "' " & _
            "where KodeJenis='" & TextBox1.Text & "'"
        cmd = New OleDbCommand(Ubah, Conn)
        cmd.ExecuteNonQuery()
        MsgBox("Ubah data sukses..!", MsgBoxStyle.Information, "Perhatian")
        Call TampilJenis()
        Call Kosong()
        TextBox1.Focus()
    End If
End Sub
```

9. Kode program button3 untuk **Hapus**

```
Private Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button3.Click
    If TextBox1.Text = "" Then
        MsgBox("Kode Buku belum diisi")
        TextBox1.Focus()
        Exit Sub
    Else
        If MessageBox.Show("Yakin akan menghapus Data Jenis " & TextBox1.Text &
            " ?", "", MessageBoxButtons.YesNo) = Windows.Forms.DialogResult.Yes Then
            cmd = New OleDbCommand("Delete * From Jenis where KodeJenis='" & TextBox1.Text & "'", Conn)
            cmd.ExecuteNonQuery()
            Call Kosong()
            Call TampilJenis()
        Else
            Call Kosong()
        End If
    End If
End Sub
```

10. Kode program button4 untuk **Batal**

```
Private Sub Button4_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button4.Click
    Call Kosong()
End Sub
```

11. Kode program TextBox1\_KeyPress berfungsi sebagai validasi Kode Jenis jika kode jenis yang telah ada didalam Tabel Jenis maka TextBox1 akan secara otomatis muncul data yang telah ada.

```
Private Sub TextBox1_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
    TextBox1.MaxLength = 2
    If e.KeyChar = Chr(13) Then
        cmd = New OleDbCommand("Select * From Jenis where KodeJenis='" & TextBox1.Text & "'", Conn)
        rd = cmd.ExecuteReader
        rd.Read()
        If rd.HasRows = True Then
            TextBox2.Text = rd.Item(1)
            TextBox2.Focus()
        Else
            Call Isi()
            TextBox2.Focus()
        End If
    End If
End Sub
```

12. Kode program TextBox3 berfungsi sebagai TextBox pencarian data yang telah tersimpan di dalam Tabel Jenis.

```
Private Sub TextBox3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles TextBox3.Click
    cmd = New OleDbCommand("Select * From Jenis where KodeJenis like '%" & TextBox3.Text & "'", Conn)
    rd = cmd.ExecuteReader
    rd.Read()
    If rd.HasRows Then
        da = New OleDbDataAdapter("Select * From Jenis where KodeJenis like '%" & TextBox3.Text & "'", Conn)
        ds = New DataSet
        da.Fill(ds, "Dapat")
        DataGridView1.DataSource = ds.Tables("Dapat")
        DataGridView1.ReadOnly = True
    Else
        MsgBox("Data tidak ditemukan")
    End If
End Sub
Class
```

### 3. Tugas (Poin 50)

Buatlah Form yang dapat menyimpan data Mahasiswa, sebagai berikut:

1. NIM
2. Nama
3. Alamat
4. No Telepon/HP
5. email

Kirim Project anda dalam bentuk file WinRar/WinZip ke email: **updi.fery@gmail.com** dengan Subject/Topik: **TUGAS VB.NET Nama\_MHS STMIK IP** (Paling lambat Tanggal.28 Mei 2017)

# 8

## Relasi Antar Tabel

### POKOK BAHASAN:

- Relasi Antar Tabel

### TUJUAN BELAJAR:

Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami dan mengenal Relasi antar Tabel
2. Dapat membuat aplikasi database dengan akses data relasi antar tabel

### Dasar Teori

Merupakan hubungan yang terjadi pada suatu tabel dengan tabel yang lainnya, yang berfungsi untuk mengatur operasi suatu database. Hubungan yang dapat dibentuk dapat mencakupi 3 (tiga) macam hubungan yaitu :

#### 1. One-To-One (1 – 1)

Mempunyai pengertian “Setiap baris data pada tabel pertama dihubungkan hanya ke satu baris data pada tabel ke dua”

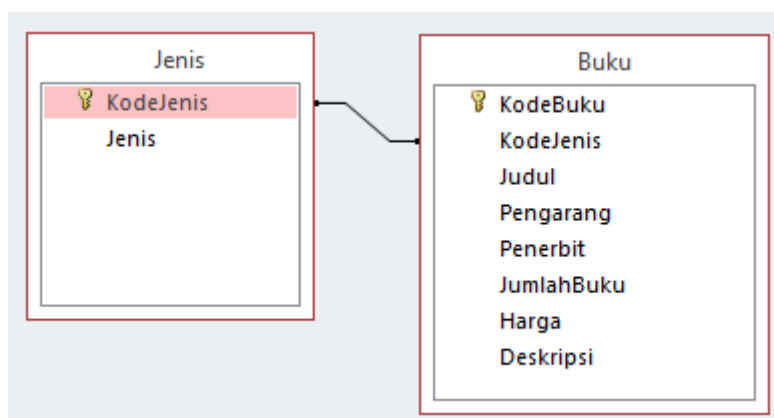
#### 2. One-To-Many (1 – M)

Mempunyai pengertian “Setiap baris data dari tabel pertama dapat dihubungkan ke satu baris atau lebih data pada tabel ke dua”

#### 3. Many-To-Many ( M–N )

Mempunyai pengertian “Satu baris atau lebih data pada tabel pertama bisa dihubungkan ke satu atau lebih baris data pada tabel ke dua “

**Contoh :** Relasi antar Tabel antara tabel Jenis dengan Tabel Buku



Gambar 8.1 Rancangan ERD (Entity Relationship Diagram)

### Menambahkan Tabel baru ke dalam Database (Microsoft Access)

Langkah-langkah dalam menambahkan tabel ke dalam database adalah sebagai berikut :

13. Buka database dengan nama **JualBuku.accdb** lalu klik menu Create > Table kemudian rancanglah struktur tabel seperti berikut ini :



Nama Tabel : **Buku**

Field	Type	Size	Keterangan
KodeBuku	Text	3	Primary Key (PK)
KodeJenis	Text	2	
Judul	Text	100	
Pengarang	Text	50	
Penerbit	Text	50	
JumlahBuku	Number		
Harga	Number		
Deskripsi	Text	255	

14. Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar

The screenshot shows a Windows form titled "Data Buku". The form has a standard Windows title bar with minimize, maximize, and close buttons. The main area contains the following controls:

- Kode Buku:** A text box labeled "TextBox1".
- Jenis Buku:** A dropdown menu labeled "ComboBox1".
- Kode Buku:** A text box labeled "TextBox8" (likely a duplicate of the first one).
- Judul:** A text box labeled "TextBox2".
- Pengarang:** A text box labeled "TextBox3".
- Penerbit:** A text box labeled "TextBox4".
- Deskripsi:** A text area labeled "TextBox7".
- Jumlah Buku:** A text box labeled "TextBox5".
- Harga:** A text box labeled "TextBox6".

Below the input fields, there are five buttons arranged horizontally, each with a number above it:

- 1 Simpan
- 2 Ubah
- 3 Hapus
- 4 Batal
- 5 Tutup

At the bottom of the form, there is a text box labeled "TextBox9" and a large gray rectangular area labeled "DataGridView1".

15. Kemudian tuliskan kode program pada Form Data Buku dibawah ini :

```
Imports System.Data.OleDb
Public Class DataBuku
    Sub Kosong()
        TextBox1.Clear()
        ComboBox1.Text = ""
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        TextBox7.Clear()
        TextBox8.Clear()
        TextBox1.Focus()
    End Sub

    Sub Isi()
        ComboBox1.Text = ""
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        TextBox7.Clear()
        ComboBox1.Focus()
    End Sub

    Sub TampilBuku()
        da = New OleDbDataAdapter("Select * From Buku", Conn)
        ds = New DataSet
        ds.Clear()
        da.Fill(ds, "Buku")
        DataGridView1.DataSource = ds.Tables("Buku")
        DataGridView1.Refresh()
    End Sub

    Sub TampilJenis()
        cmd = New OleDbCommand("Select KodeJenis From Jenis", Conn)
        rd = cmd.ExecuteReader
        Do While rd.Read
            ComboBox1.Items.Add(rd.Item(0))
        Loop
    End Sub

    Sub AturGrid()
        'Untuk mengatur luas Columns pada DataGridView
        DataGridView1.Columns(0).Width = 60
        DataGridView1.Columns(1).Width = 50
        DataGridView1.Columns(2).Width = 300
        DataGridView1.Columns(3).Width = 100
        DataGridView1.Columns(4).Width = 100
        DataGridView1.Columns(5).Width = 100
        DataGridView1.Columns(6).Width = 100
        DataGridView1.Columns(7).Width = 300

        'Untuk menampilkan Judul Header pada DataGridView
        DataGridView1.Columns(0).HeaderText = "KODE BARANG"
        DataGridView1.Columns(1).HeaderText = "KODE JENIS"
        DataGridView1.Columns(2).HeaderText = "JUDUL"
        DataGridView1.Columns(3).HeaderText = "PENGARANG"
        DataGridView1.Columns(4).HeaderText = "PENERBIT"
        DataGridView1.Columns(5).HeaderText = "JUMLAH"
        DataGridView1.Columns(6).HeaderText = "HARGA"
        DataGridView1.Columns(7).HeaderText = "DESKRIPSI"
    End Sub
End Class
```

```

Private Sub DataBuku_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
    Call Koneksi()
    Call TampilJenis()
    Call TampilBuku()
    Call Kosong()
    Call AturGrid()
End Sub

Private Sub ComboBox1_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles
cmd = New OleDbCommand("Select * From Jenis where KodeJenis='" & ComboBox1.Text & "'", Conn)
rd = cmd.ExecuteReader
rd.Read()
If rd.HasRows = True Then
    TextBox8.Text = rd.Item(1)
Else
    MsgBox("Kode jenis ini tidak terdaftar")
End If
End Sub

Private Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click
If TextBox1.Text = "" Or ComboBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or
TextBox4.Text = "" Or TextBox5.Text = "" Or TextBox6.Text = "" Then
    MsgBox("Data belum lengkap..!")
    TextBox1.Focus()
    Exit Sub
Else
    cmd = New OleDbCommand("Select * From Buku where KodeBuku='" & TextBox1.Text & "'", Conn)
    rd = cmd.ExecuteReader
    rd.Read()
    If Not rd.HasRows Then
        Dim Simpan As String =
            "insert into Buku(KodeBuku,KodeJenis,Judul,Pengarang,Penerbit,JumlahBuku,Harga,Deskripsi)values " & _
            "(" & TextBox1.Text & "','" & ComboBox1.Text & "','" &
            TextBox2.Text & "','" & TextBox3.Text & "','" &
            TextBox4.Text & "','" & TextBox5.Text & "','" &
            TextBox6.Text & "','" & TextBox7.Text & "'"
        cmd = New OleDbCommand(Simpan, Conn)
        cmd.ExecuteNonQuery()
        MsgBox("Simpan data sukses...!", MsgBoxStyle.Information, "Perhatian")
    End If
    Call TampilBuku()
    Call Kosong()
    TextBox1.Focus()
End If
End Sub

Private Sub Button4_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button4.Click
    Call Kosong()
End Sub

Private Sub Button5_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button5.Click
    Me.Close()
End Sub
End Class

```

### Tugas (Poin 50)

Lengkapilah program diatas dengan menambah fungsi button **Ubah, Hapus, Pencarian data** pada TextBox9 dan fungsi **validasi**.

### Praktikum 1 (Poin 30)

#### Menambahkan Tabel baru ke dalam Database (Microsoft Access)

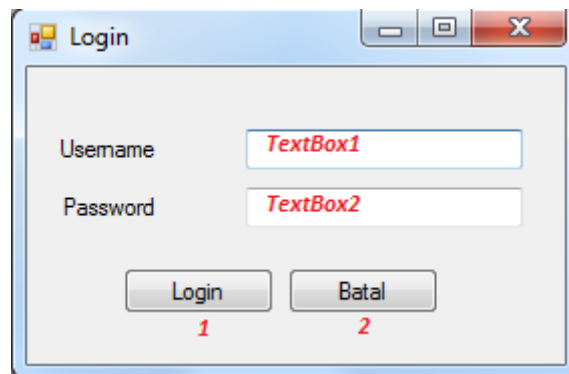
Langkah-langkah dalam menambahkan tabel ke dalam database adalah sebagai berikut :

1. Buka database dengan nama **JualBuku.accdb** lalu klik menu Create > Table kemudian rancanglah struktur tabel seperti berikut ini :

Nama Tabel : **Pemakai**

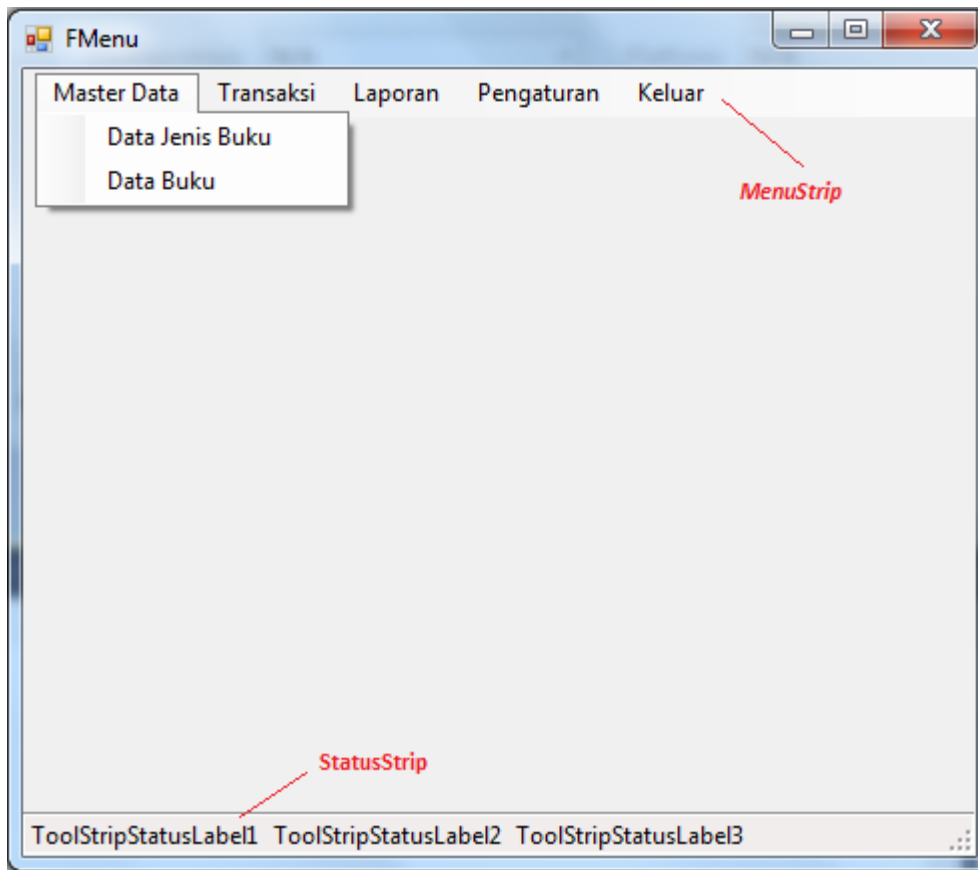
Field	Type	Size	Keterangan
Kode_Pmk	Text	10	Primary Key (PK)
Nama_Pmk	Text	10	
Status_Pmk	Text	15	
Password_Pmk	Text	10	

2. Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar (nama form: Login)

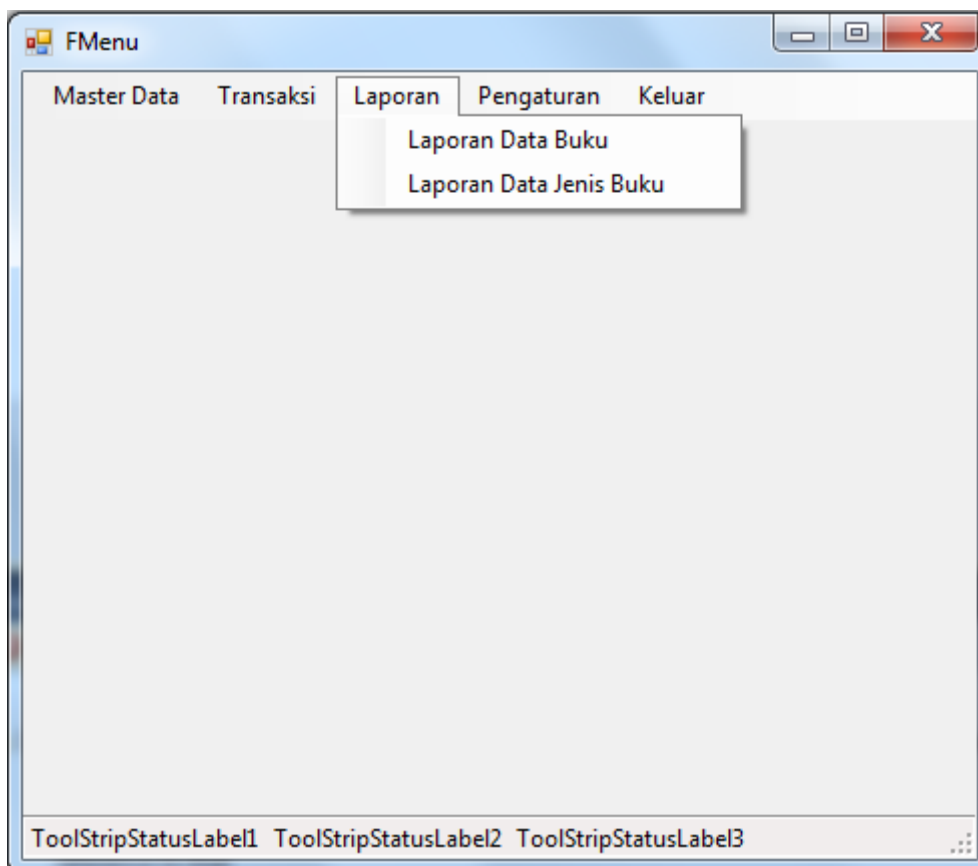


Gambar 1

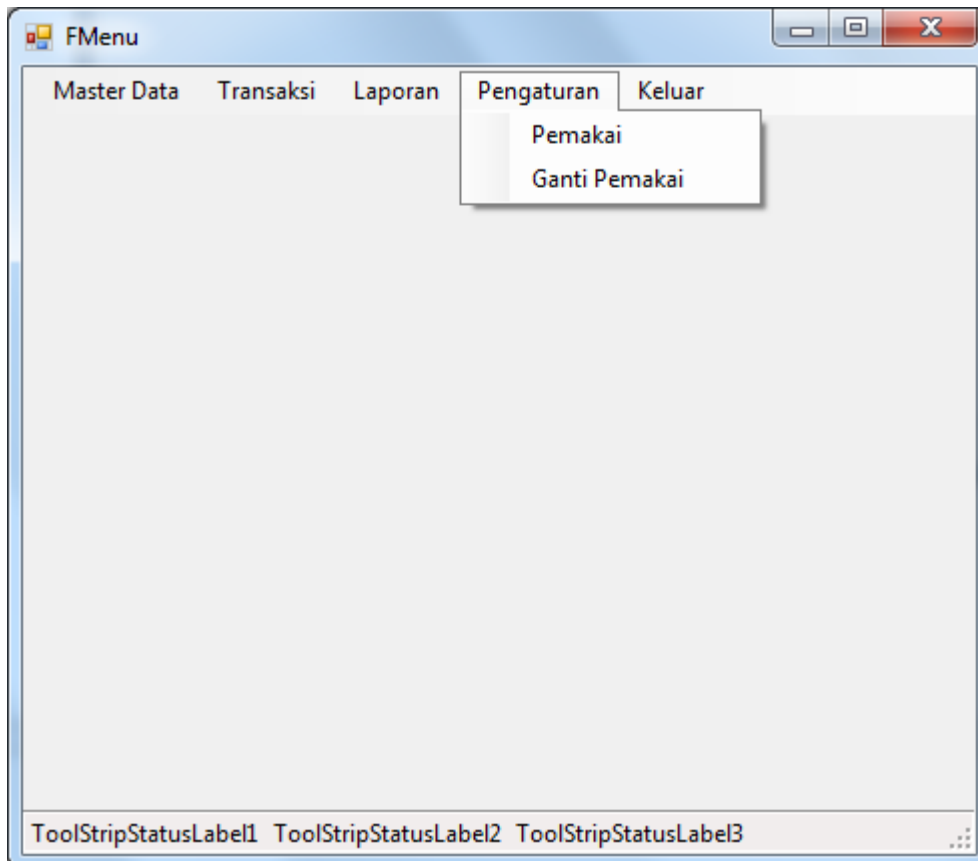
3. Buatlah sebuah form baru pada Visual BASIC .NET, desain tampilan form sehingga didapat tampilan seperti pada gambar (nama form: Fmenu)



Gambar 2



Gambar 3



Gambar 4

4. Kemudian tuliskan kode program pada Form **Login** dibawah ini :

```
Imports System.Data.OleDb
Public Class Login
    Private Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button1.Click
        Koneksi()
        cmd = New OleDbCommand("select * from Pemakai where Nama_Pmk='" &
            TextBox1.Text & "' and Password_Pmk='" &
            TextBox2.Text & "'", Conn)
        rd = cmd.ExecuteReader
        rd.Read()
        If rd.HasRows Then
            Me.Visible = False
            FMenu.Show()
            FMenu.ToolStripStatusLabel1.Text = rd.GetString(0)
            FMenu.ToolStripStatusLabel2.Text = rd.GetString(1)
            FMenu.ToolStripStatusLabel3.Text = rd.GetString(2)
            If FMenu.ToolStripStatusLabel3.Text <> "ADMINISTRATOR" Then
                FMenu.PemakaiToolStripMenuItem.Enabled = False
            Else
                FMenu.PemakaiToolStripMenuItem.Enabled = True
            End If
        Else
            MsgBox("login salah, periksa kembali Nama Pemakai dan password Anda...!!")
            TextBox1.Focus()
        End If
    End Sub
End Class
```

```

Private Sub TextBox1_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs)
    If e.KeyChar = Chr(13) Then TextBox2.Focus()
End Sub

Private Sub TextBox2_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs)
    If e.KeyChar = Chr(13) Then Button1.Focus()
End Sub

Private Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click
    TextBox1.Clear()
    TextBox2.Clear()
End Sub
End Class

```

5. Kemudian tuliskan kode program pada Form **FMenu** dibawah ini :

```

Public Class FMenu
    Private Sub KeluarToolStripMenuItem1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
        End
    End Sub

    Private Sub DataJenisBukuToolStripMenuItem_Click(ByVal sender As Object, ByVal e As System.EventArgs) H
        DataJenis.Show()
    End Sub

    Private Sub DataBukuToolStripMenuItem_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handle
        DataBuku.Show()
    End Sub
End Class

```

**POKOK BAHASAN:**

- Membuat Laporan

**TUJUAN BELAJAR:**

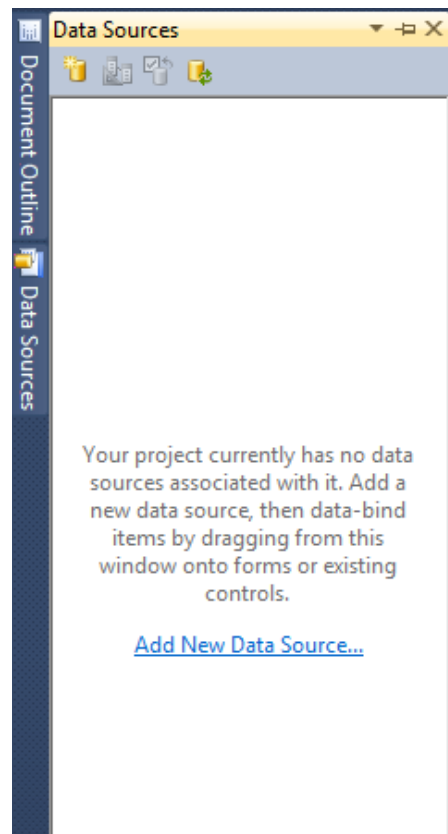
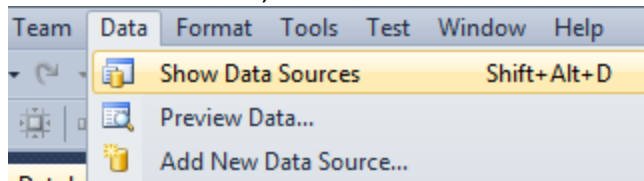
Setelah mempelajari bab ini, diharapkan mampu:

1. Memahami dan mengenal membuat laporan
2. Dapat membuat laporan dengan DataSet dan Report Viewer

**Dasar Teori****Membuat Laporan dengan DataSet**

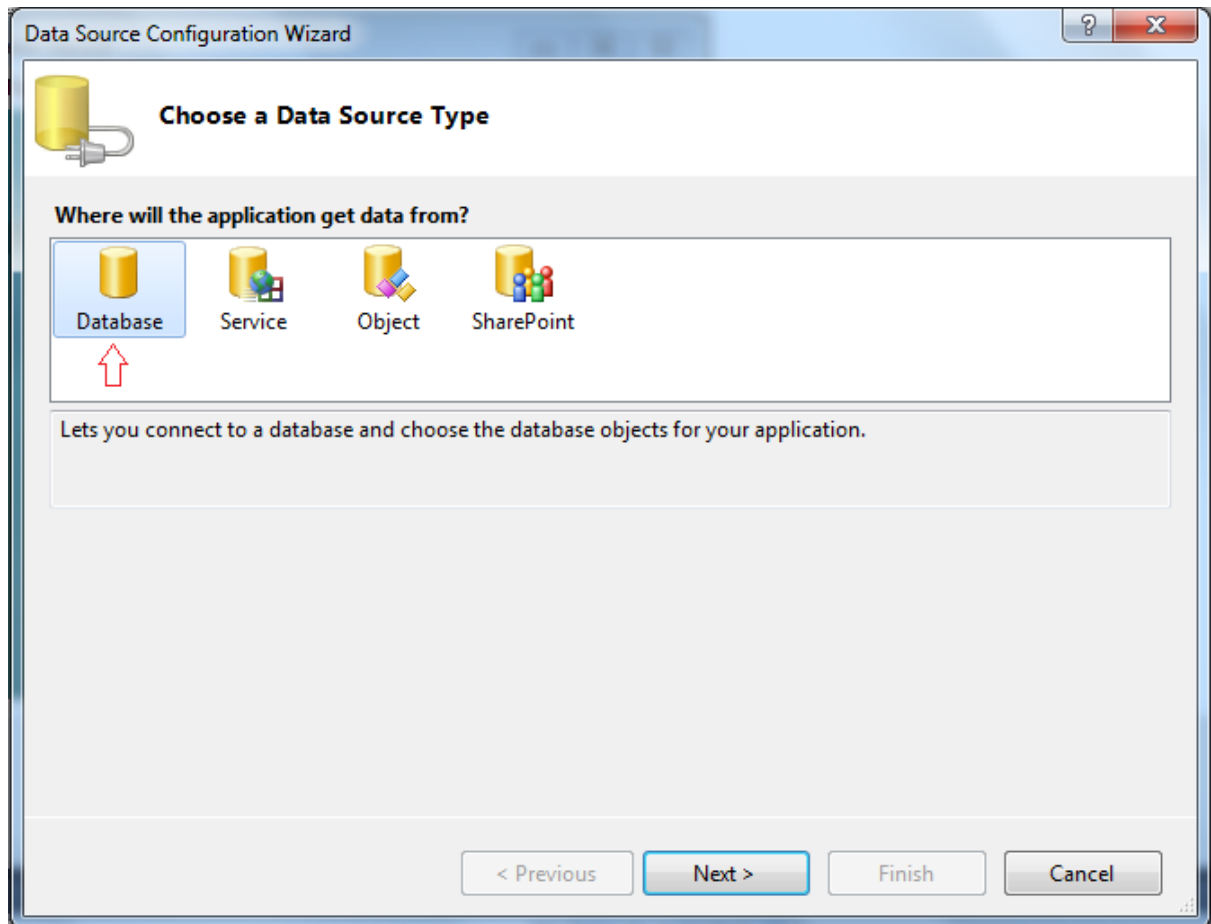
Menghubungkan DataSet dengan Data Sources ke database langkah-langkahnya sebagai berikut:

1. Klik Tab menu **Data**, lalu klik **Show Data Source** .

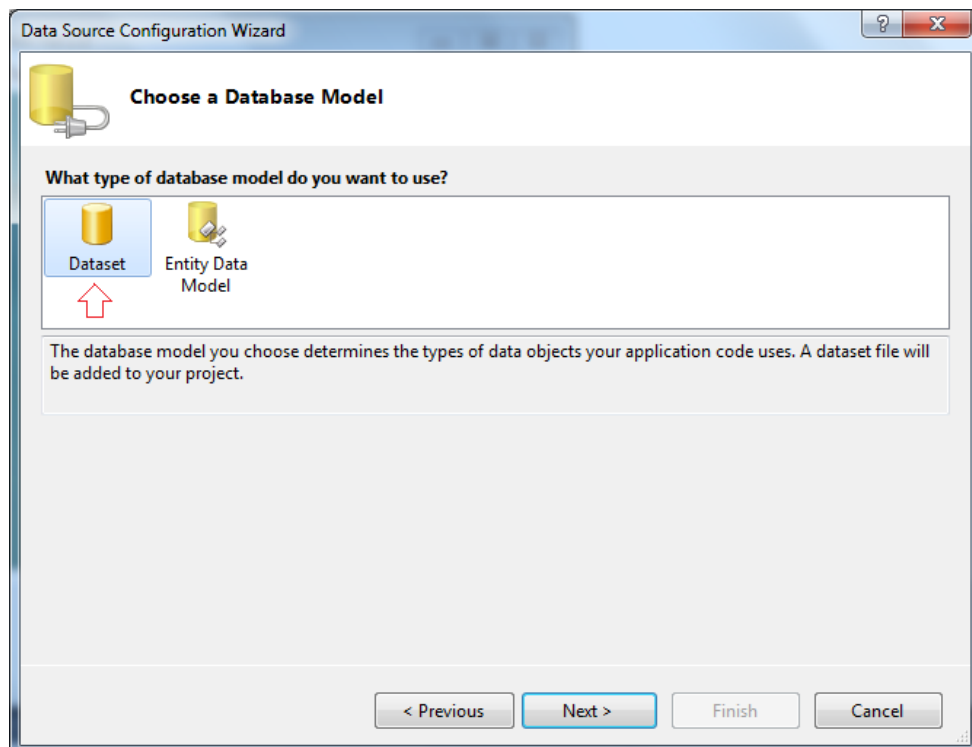


2. Klik **Add New Data Source**

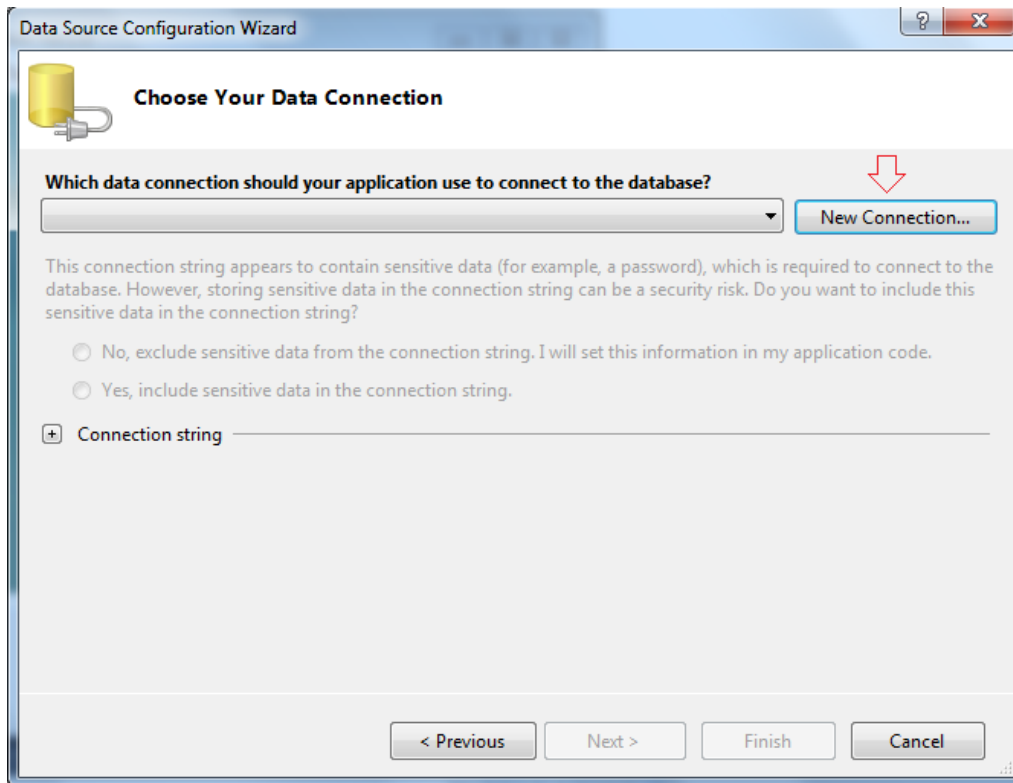




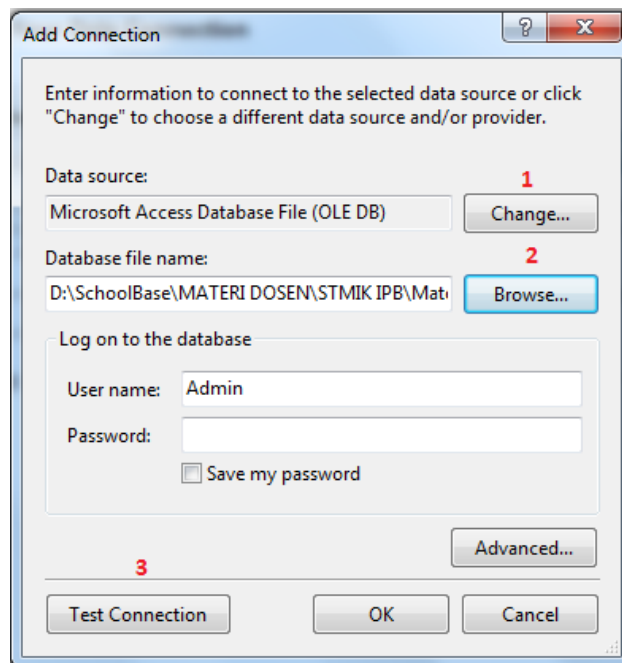
3. Pilih **Database**, Klik **Next**

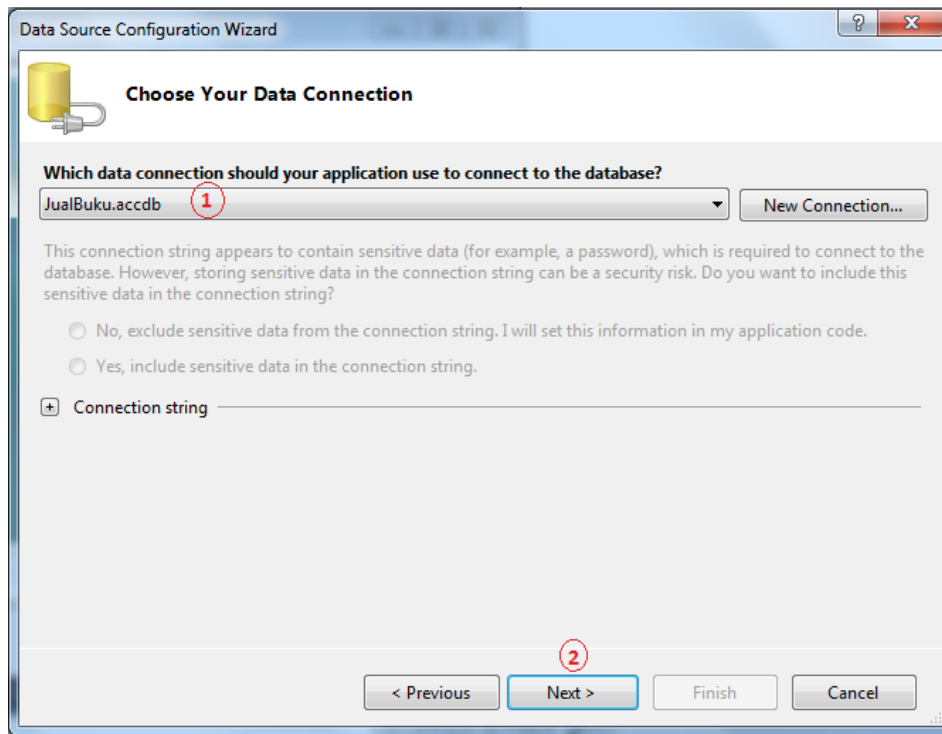


4. Pilih **Dataset**, Klik **Next**

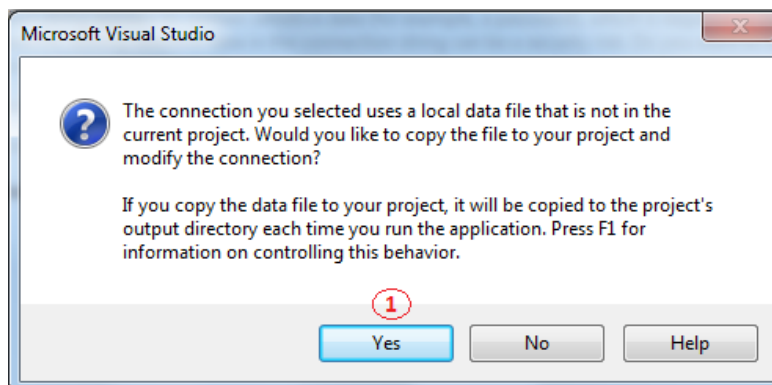


5. Klik **New Connection**, Pilih **Microsoft Access Data File**

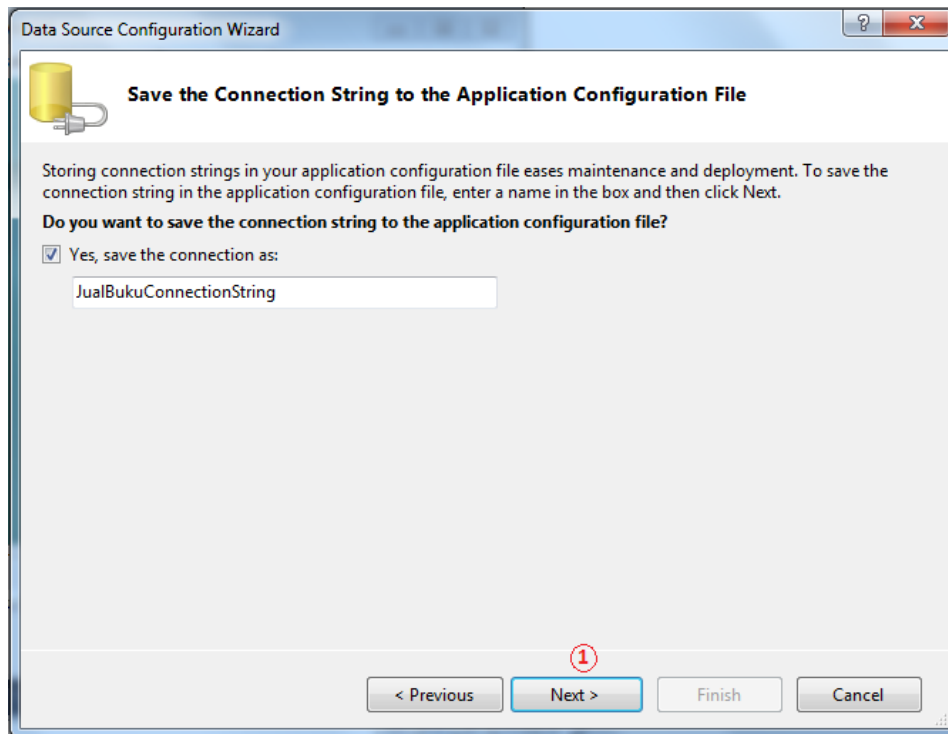




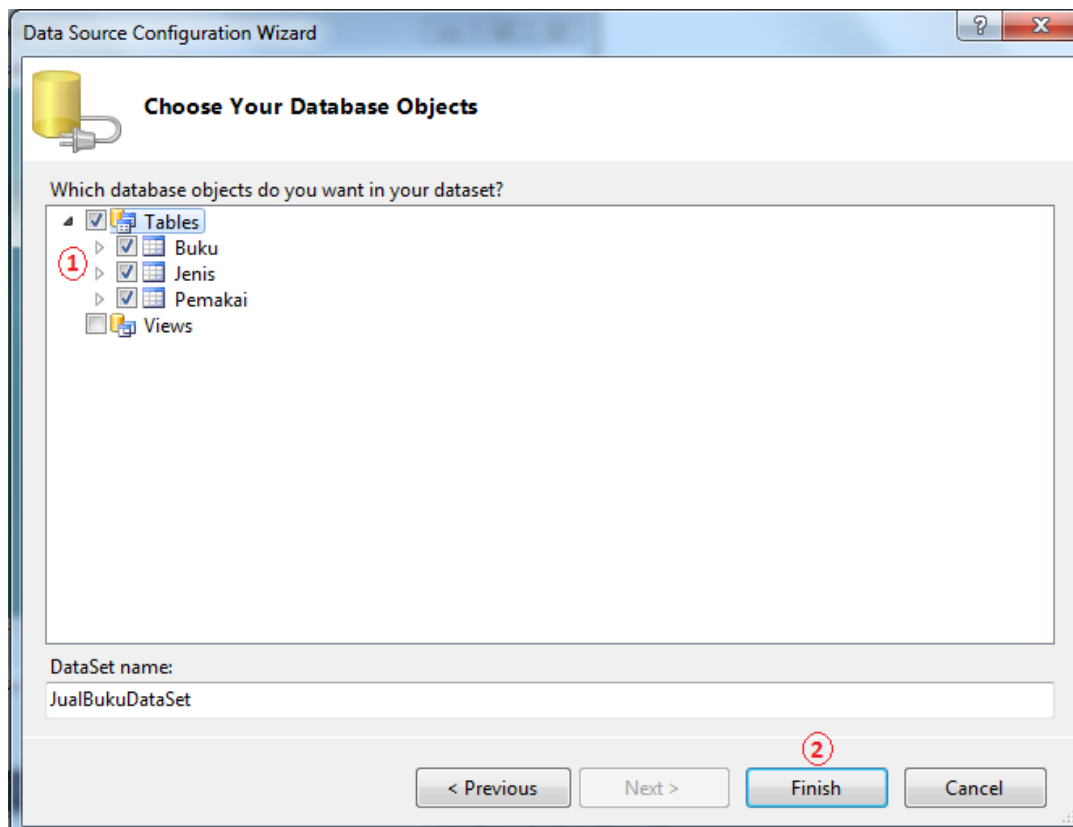
6. Klik **Next**, pada kotak dialog configuration wizard, klik **next**



7. Klik **Yes**
8. Klik **Next**

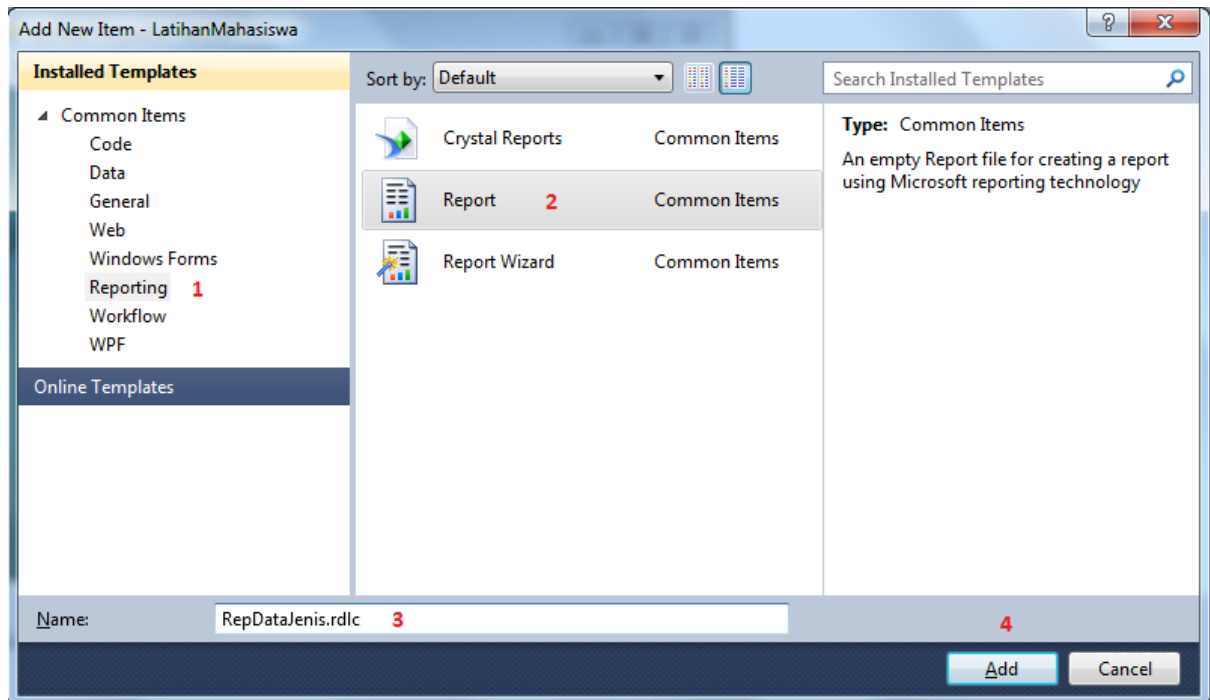


9. Beri tanda cek pada **table**, klik **finish**

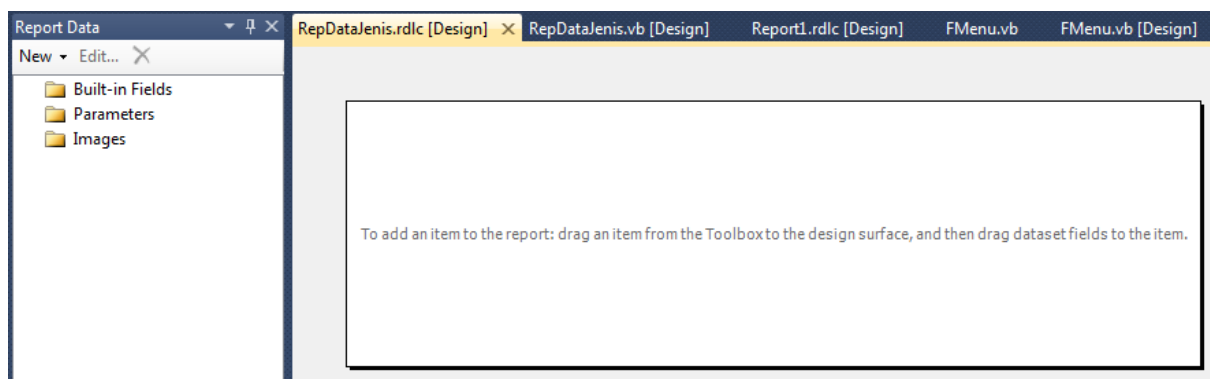


## Membuat Rancangan Design Laporan dengan Report

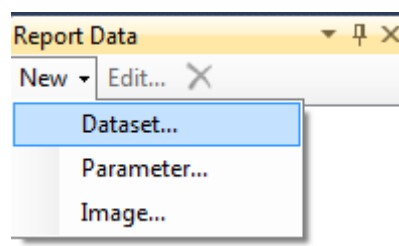
1. Membuat Form Design Report, langkah-langkahnya sebagai berikut:
2. Klik Kanan **Project** > **Add** > **Windows Form** > Pilih **Reporting** > Klik **Report** > Beri Name **RepDataJenis.rdlc** > **Add**.



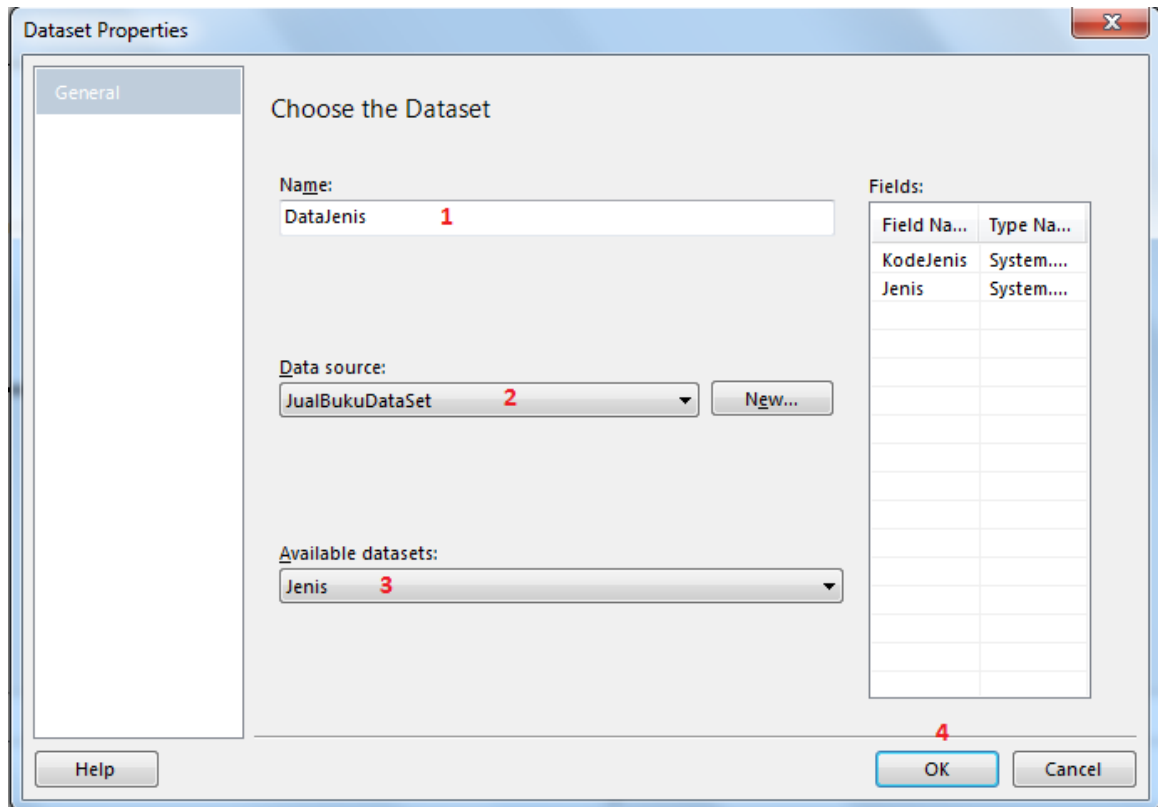
3. Akan tampil sebagai berikut:



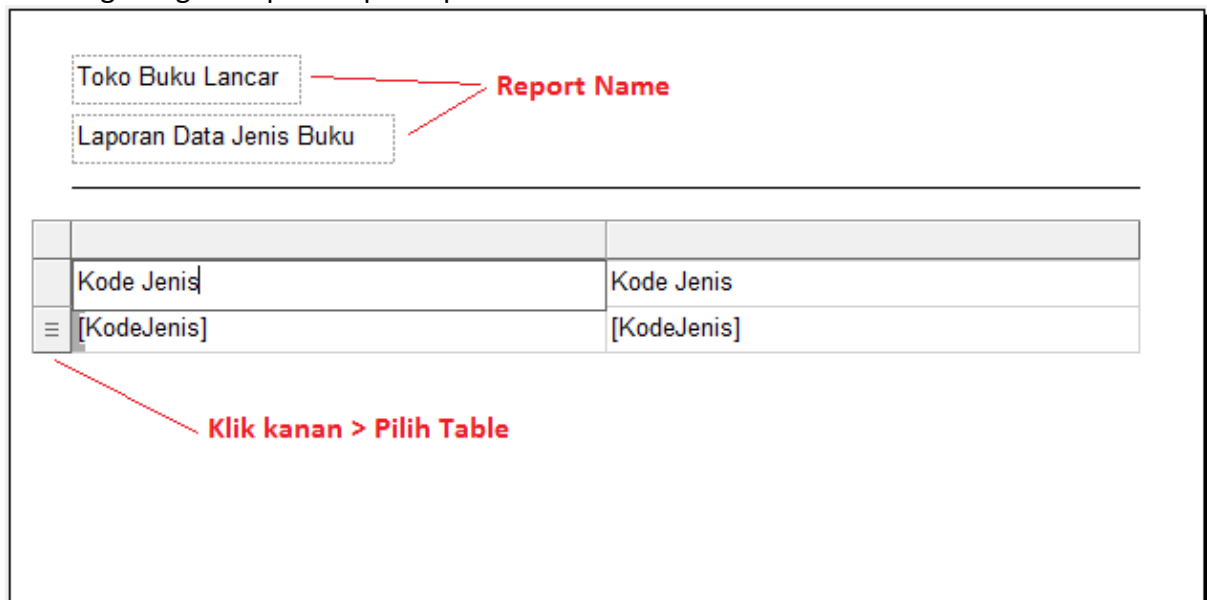
4. Klik **New, DataSet**



5. Atur seperti gambar dibawah ini:

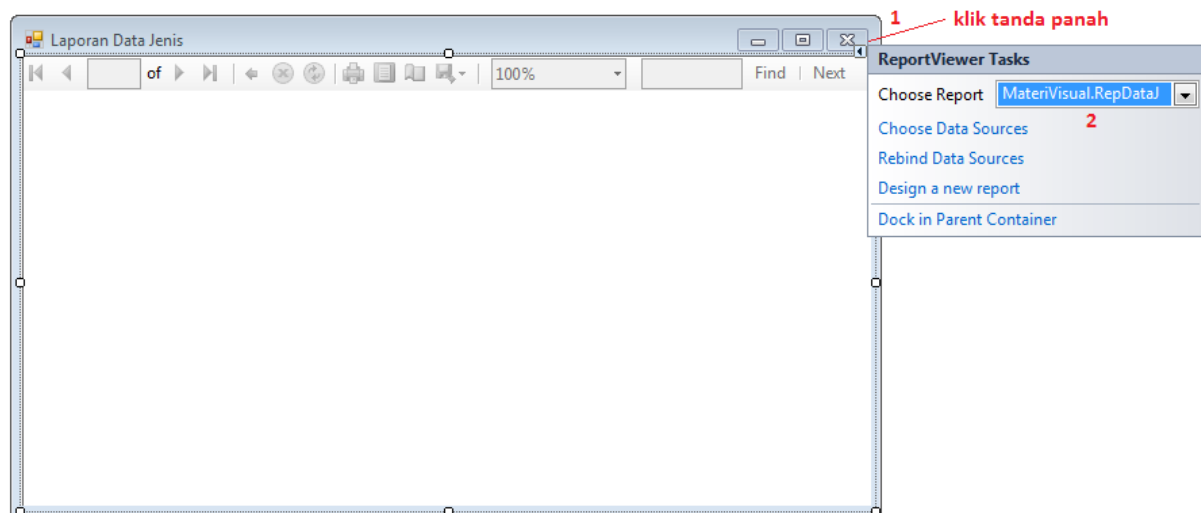


6. Rancang design tampilan rapot seperti berikut:



## Membuat Rancangan Form untuk memunculkan Laporan

1. Buat Form baru klik kanan **Project > Add > Windows Form**.
2. Rancang design form laporan dengan menggunakan **ReportViewer**



3. Setelah merancang tampilan form untuk laporan langkah selanjutnya adalah menambahkan kode program ke dalam menu utama dengan cara memanggil nama Form laporan Contoh : `RepDataJenis.Show()`