

STEP BY STEP MENJADI PROGRAMMER HANDAL DENGAN VB.NET

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Junindar

Blog : <http://junindar.blog.com>

E-Mail : xx_lead@yahoo.com

junindar@gmail.com

STEP XIII

Membuat Aplikasi Database dengan menggunakan Vb.Net 2003 dan SQL Server 2000

I. Pengantar

Artikel ini saya buat, karena sangat banyak sekali Email yang masuk ke saya untuk membuat artikel Vb.Net dengan Sql Server sebagai database nya.

Setelah sekian lama akhir nya pada artikel saya yang ke 13 ini baru dapat saya realisasikan. Untuk studi kasus pada artikel ini saya ambil dari artikel pertama saya (Step 1). Oleh karena itu tata cara penulisan dan penyampaian sangat mirip dengan step 1.

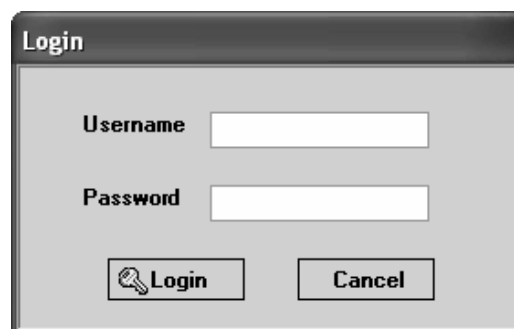
Dalam artikel ini para pembaca akan, di bimbing bagaimana membuat program aplikasi sederhana dari mulai membuat Database, Table, Stored procedure pada Sql Server 2000 sampai pada pemogramannya dengan Vb.net 2003.

II. A TIME TO MAKE PROGRAM

2.1 Desain Form

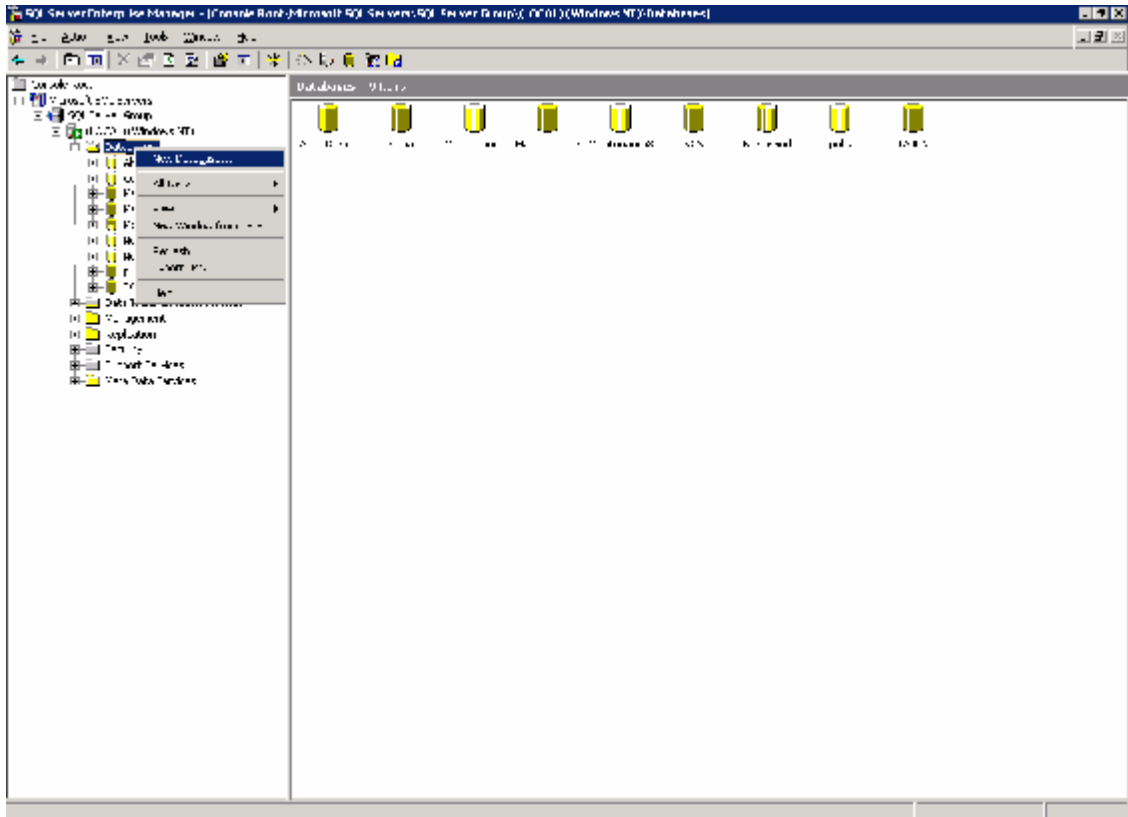


Gambar Flash Form

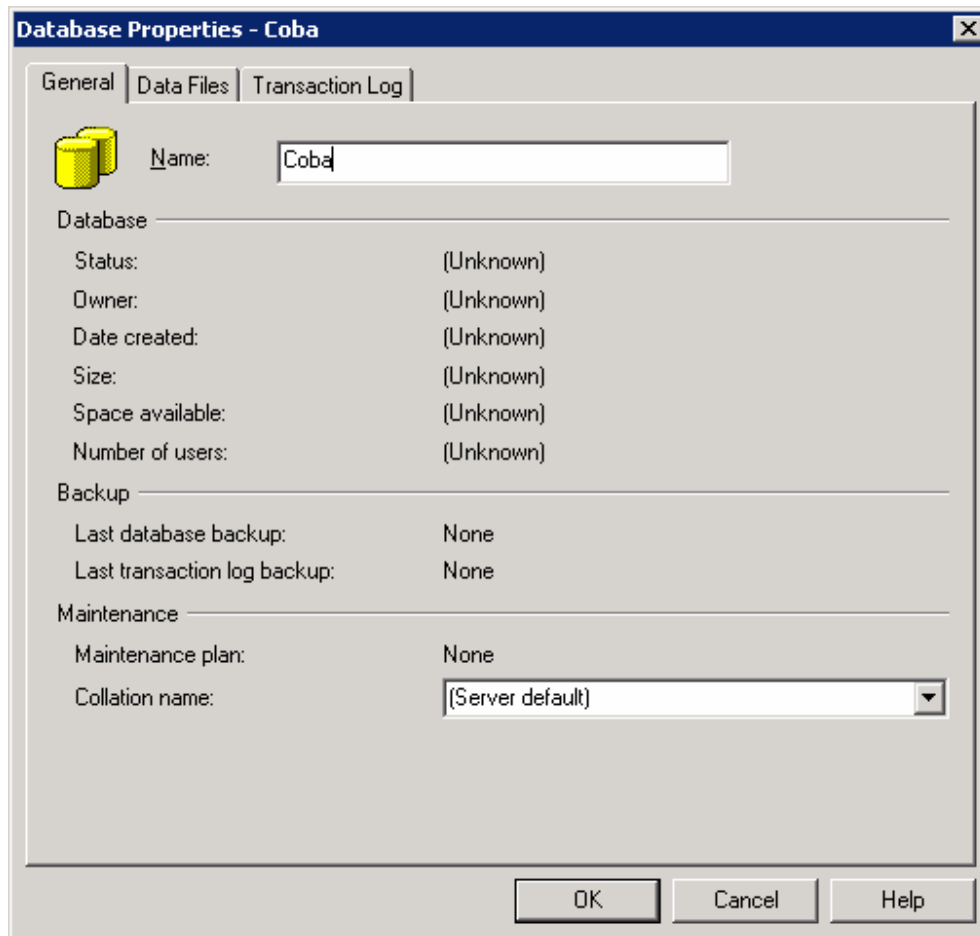


Gambar Form Login

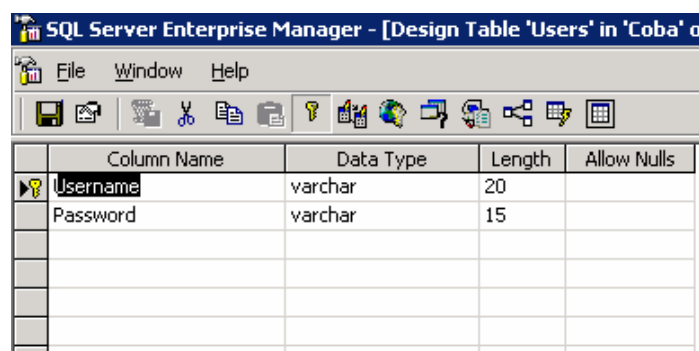
Untuk membuat program seperti diatas, mari ikuti langkah-langkah dibawah ini.
Sebelum kita masuk dalam pembuatan program, terlebih dahulu kita buat database pada Sql Server. Buatlah database dengan nama Coba. Ikuti langkah-langkah dibawah ini. Pada jendela SQL Server Enterprise Manager, klik kanan pada folder Database lalu pilih New Database.



Lalu akan muncul jendela Database properties. Ketikkan **Coba** untuk nama database yang akan kita buat. Dan klik button OK.



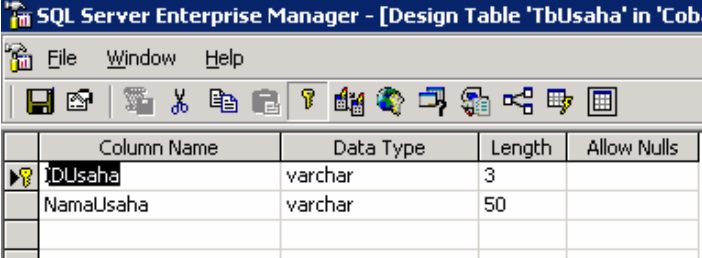
Setelah itu di bawah folder Database, akan terlihat database yang baru saja kita buat (Coba). Klik tanda + pada database coba, dan klik kanan pada root **Table** dan pilih **New Table**.



Gambar Desain Table Users

Buatlah field seperti gambar diatas, untuk membuat Primary Key, klik row Username lalu klik tanda kunci pada toolbar diatas. Setelah selesai klik Save untuk menyimpan design table dengan nama "Users". Setelah selesai buat satu lagi table dengan nama

TbUsaha, dengan nama field **IdUsaha** dan **NamaUsaha** (**IdUsaha** sebagai Primary Key nya).



The screenshot shows the 'Design Table' window for 'TbUsaha' in SQL Server Enterprise Manager. The table has two columns: 'IdUsaha' with a data type of 'varchar', length of '3', and 'Allow Nulls' set to 'No' (indicated by a key icon); and 'NamaUsaha' with a data type of 'varchar', length of '50', and 'Allow Nulls' set to 'Yes'.

Column Name	Data Type	Length	Allow Nulls
IdUsaha	varchar	3	No
NamaUsaha	varchar	50	Yes

Gambar Desain Table TbUsaha

Setelah selesai dalam pembuatan database, sekarang kita masuk dalam pembuatan program.

Buka Aplikasi Visual Studio.Net, dan pilih New Project. Ketikkan nama project sesuai dengan keinginan anda. Simpan di drive pada PC anda (Untuk drive bebas juga, sekarang jamannya hak asasi manusia, tidak ada pemaksaan).

3.1.1 Membuat StartUp Form.

Untuk pertama kali, .Net akan memberikan Default sebuah form, dengan nama Form1. Sekarang tugas kita adalah merubah tampilan Form1 tersebut menjadi seperti tampilan Flash Form seperti diatas.

Ubah properties Form1 seperti dibawah ini.

BackColor	White
FormBorderStyle	FixedToolWindow
Text	(Kosongkan)
Name	StartUp
StartPosition	CenterScreen
ControlBox	False
ShowInTaskbar	False

Setelah selesai melakukan perubahan Form1 seperti diatas. Sekarang kita tambah aksesoris pada form StartUp. Tambahkan dua (2) buah label, 1 ProgressBar dan 1 buah PictureBox. Sehingga tampilan seperti pada gambar dibawah ini.



Desain Splash Form

Untuk gambar diatas silahkan ganti dengan gambar yang anda suka dan jika anda tidak suka dengan tampilan ini, anda dapat merubah sesuai dengan keinginan anda. Setelah selesai mendesain form, tambahkan sebuah timer dan beri nama Splash_Timer.

3.1.2 Membuat Login Form.

Tambahkan sebuah Form ke dalam aplikasi. Caranya Klik Kanan Pada Project anda Pilih Add>Add Windows Form. Ganti properties Form baru tersebut seperti dibawah ini.

BackColor	LightBlue
Font+	Microsoft Sans Serif, 8.25pt, style=Bold
FormBorderStyle	Fixed3D
Text	Login
Name	FrmLogin
StartPosition	CenterScreen
ControlBox	False
ShowInTaskbar	False

Tambahkan 2 buah label, 2 buah TextBox dan 2 buah Button Ubah properties dari masing-masing component seperti dibawah ini.

Label 1

BackColor	LightBlue
Font+	Microsoft Sans Serif, 8.25pt, style=Bold
Text	Username

Label 2

BackColor	LightBlue
Font+	Microsoft Sans Serif, 8.25pt, style=Bold
Text	Password

Textbox 1

BorderStyle	Fixed3D
Name	txtUser

Textbox 2

BorderStyle	Fixed3D
Name	txtPassword
PasswordChar	*

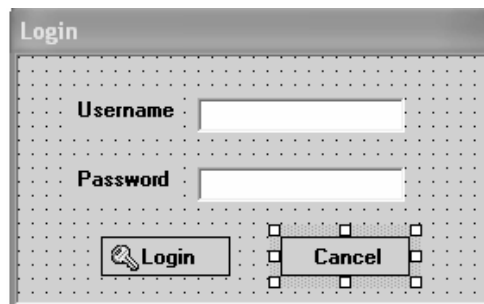
Button1

BackColor	LightBlue
Font+	Microsoft Sans Serif, 8.25pt, style=Bold
Flatstyle	Flat
Text	Login
Name	btnlogin

Button2

BackColor	LightBlue
Font+	Microsoft Sans Serif, 8.25pt, style=Bold
Flatstyle	Flat
Text	Cancel
Name	btnbatal

Setelah selesai merubah properties seperti di atas, susun dan disain form seperti dibawah ini.



Gambar Desain Form Login

3.1.3 Membuat Menu Utama

Sekarang kita akan membuat Form Utama, tapi untuk sementara jangan di modifikasi. Untuk saat ini kita ganti Form Name menjadi **MainMenu**

2.2 Source Code

2.2.1 Source Code Form Splash

Setelah kita selesai mendesain 2 buah form, sekarang waktunya kita bermain dengan source code. Pada Form splash, klik 2 kali (Double Click) component Splash_Timer . Dan ketikkan listing seperti dibawah ini.

```
Private Sub Splash_Timer_Tick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Splash_Timer.Tick

    Dim cur As New frmLogin
    ProgressBar1.Value += 4
    If ProgressBar1.Value = 100 Then
        Splash_Timer.Dispose()
        Me.Visible = False
        cur.Show()
    End If
End Sub
```

Note : anda dapat mencoba mengubah nilai value pada progressbar. Dengan menaikkan atau mengurangi nilai value maka anda melihat perubahan progressbar berjalan, apakah semakin cepat atau lambat. Itu semua tergantung pada anda.

Coba jalankan program, gmana? apakah sudah sesuai dengan yang anda inginkan ????????

2.2.2 Source Code Form Login

Sebelum kita mulai membuat listing pada form login, terlebih dahulu kita membuat koneksi pada database. Sekarang kita akan membuat koneksi dengan memanggil Class lain untuk koneksi ke database.

Sebelumnya Buat folder pada project yang anda buat (Klik kanan project>Add>New Folder) dan beri nama “**AccessData**”. Selanjutnya didalam Folder tersebut tambahkan Class dengan nama “**DataBaseConnection.vb**” (Klik kanan Folder “AccessData”>Add>Add Class). Ketikkan listing di bawah ini pada class yang baru anda buat.

```
Imports System.Data.SqlClient

Namespace AccessData

    Public Class DataBaseConnection

        Public Function open() As SqlConnection
            'Server = Nama dimana kita menginstall SQL server
            'Database = Nama database yang akan di akses
            'User ID= Account untuk SQL server, sebagai default sa (system
            administrator)
            'Password = Password untuk login ke SQL server
            Dim conect As New
            SqlConnection("Server='TESTSERVER';Database='Coba'; " & _
            " User ID='sa';Password='testserver'")

            conect.Open()
            Return conect
        End Function

        Public Function close() As SqlConnection

            Dim conect As New
            SqlConnection("Server='TESTSERVER';Database='Coba'; " & _
            " User ID='sa';Password='testserver'")
            conect.Close()
            Return conect
        End Function

    End Class
End Namespace
```

Note :Dengan membuat koneksi menggunakan Class, untuk setiap proses yang berhubungan dengan database anda cukup dengan memanggil nama Class tersebut, dengan mendeklarasikan Class tersebut.

Contoh : `Dim MyConnection As New AccessData.DatabaseConnection`

Setelah selesai membuat Class, selanjutnya kita akan membuat module untuk menyimpan Variable global yang sering kita gunakan dalam pembuatan program. Buat satu module caranya sama dengan membuat Class, dan beri nama “**module.vb**”. (Klik kanan Folder “AccessData”>Add>Add Module). Ketikkan listing di bawah ini pada module yang baru anda buat.

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

Setelah selesai sekarang kita kembali pada Form Login. Untuk pertama kali ketikkan listing dibawah ini pada baris paling atas.

```
Imports System.Data
Imports System.Data.SqlClient
```

Dan buat variable global untuk form Login ini, seperti dibawah ini.

```
#Region "deklarasi"
    Dim MyConnection As New AccessData.DataBaseConnection
    Dim f_utama As New MainMenu
    Dim objConnection As SqlConnection
    Dim objCommand As SqlCommand
    Dim objDataAdapter As SqlDataAdapter
    Dim objdataareader As SqlDataReader
    Dim strSQL As String
    Dim objDataSet As New DataSet
    Dim objDataTable As New DataTable
    Dim mProcess As String
#End Region
```

Ketikkan listing diatas dibawah `Inherits System.Windows.Forms.Form`

Selanjutnya kita buat listing untuk mencari apakah Username ada pada table Users

```
Sub CheckUser()
    objDataTable.Clear()
    strSQL = ("SELECT * FROM [Users] WHERE [UserName] " & _
        " = '" & Trim(txtuser.Text) & "' ")
    objCommand = New SqlCommand
    objCommand.Connection = MyConnection.open
    objCommand.CommandType = CommandType.Text
    objCommand.CommandText = strSQL
    objDataAdapter = New SqlDataAdapter(objCommand)
    objDataAdapter.Fill(objDataSet, "MDT_user")
    objDataTable = objDataSet.Tables("MDT_user")
    MyConnection.close()
End Sub
```

Fungsi dari listing adalah, mencari Username apakah ada pada table Users. Syntax SQL ("SELECT * FROM [Users] WHERE [UserName] = ' " & Trim(txtuser.Text) & " ' "), Maksudnya adalah menampilkan data user dengan criteria **Username**. Dan jumlah banyaknya data akan disimpan pada **ObjDataTable**. Kemudian ketikkan listing untuk menampilkan data users (**Username,Password**).

```
Sub Find_User()  
    objCommand = MyConnection.open.CreateCommand  
    objCommand.CommandText = ("Select [UserName],[Password] from  
    [Users] where [UserName]=' " & Trim(txtuser.Text) + " ' ")  
    objDataReader = objCommand.ExecuteReader  
    objDataReader.Read  
    Username = objDataReader.Item("Username")  
    Password = objDataReader.Item("Password")  
    MyConnection.close()  
End Sub
```

Untuk sekilas sepertinya listing antara **Sub CheckUser** sama dengan **Sub Find_User**. Tapi pada kenyataannya fungsi dari kedua listing ini berbeda. Untuk **Sub Find_User** disini kita menampilkan value dari **username & Password**. Jika anda sudah mengerti dengan yang saya maksud, mari lanjutkan pelajaran kita. Sekarang Double Click Button Login. Ketikkan Listing seperti dibawah ini.

```
Private Sub btnLogin_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLogin.Click

    `untuk mengaktifkan fungsi CheckUsers
    CheckUsers

    `mengharuskan user memasukkan username dan password
    If txtuser.Text.Trim() = "" And _
        txtpassword.Text.Trim() = "" Then
        Msg = MsgBox("Masukan Username dan Password", MsgBoxStyle.OKOnly, "Pesan")
        txtuser.Focus()
    ElseIf txtuser.Text = "" Then
        Msg = MsgBox("Masukan Username ", MsgBoxStyle.OKOnly, "Pesan")
        txtuser.Focus()
    ElseIf txtpassword.Text = "" Then
        Msg = MsgBox("Masukan password ", MsgBoxStyle.OKOnly, "Pesan")
        txtpassword.Focus()
    Else
        Try
            `Menghitung jumlah baris yang ada pada ObjDataTable
            If objDataTable.Rows.Count <= 0 Then
                `Jika jumlah baris = 0 / username tidak ada pada table
                Msg = MsgBox("Username tidak ada ", MsgBoxStyle.OKOnly, "Pesan")
                txtuser.Focus()
            Else
                `Jika username ada pada table / jumlah baris ObjdataTable > 0

                `Mengaktifkan fungsi mencari nilai value Username & Password
                Find_User()

                `Jika Password yang diketik pada TextBox tidak sama pada Database
                If Password <> Trim(txtPassword.Text) Then
                    Msg = MsgBox("Password salah!", MsgBoxStyle.OKOnly, "Pesan")
                    txtPassword.Focus()
                    Exit Sub
                Else
                    `Jika Sama
                    Me.Hide()
                    f_utama.ShowDialog()
                    txtuser.Text = ""
                    txtPassword.Text = ""
                End If
            End If
        Catch When Err.Number <> 0
            MsgBox("Tidak dapat melakukan proses" _
                & vbCrLf & Err.Description)
        End Try
    End If
End Sub
```

Setelah ini Double Click button **Cancel**, dan ketikkan listing seperti berikut

```
Private Sub btnBatal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles  
btnBatal.Click  
    intResponse = MessageBox.Show("Are you sure want to quit ?", Me.Text, MessageBoxButtons.YesNo,  
    MessageBoxIcon.Exclamation)  
    If intResponse = MsgBoxResult.Yes Then  
        End  
    Else  
        Exit Sub  
    End If  
End Sub
```

Sekarang anda telah selesai membuat aplikasi yang sangat sederhana. Kalau berhasil jangan senang dulu, karena masih banyak sekali rintangan yang akan kita hadapi untuk kasus-kasus berikutnya. Bagi yang gagal jangan berkecil hati, jangan ragu dan sungkan silahkan E-Mail ke saya.

Jika program yang anda buat tidak ada masalah, kita lanjutkan pelajaran kita.

Sekarang mari kita modif form utama kita yang masih polos, menjadi seperti berikut :



Ubah Properties pada form utama seperti di bawah ini.

BackColor	White
FormBorderStyle	Fixed3D
BackGroundImage	Bebas mau gambar apa
Text	MainMenu
Name	MainMenu
StartPosition	CenterScreen
Icon	Bebas juga

Anda dapat memodif form sesuai dengan keinginan anda. Dan kita tambah komponen MainMenu1 dan StatusBar1 pada Form MainMenu. Buat seperti dibawah ini.

File>>Exit



Inputan>>Input Usaha



Ketik listing seperti dibawah ini.

```
Public Sub BuatPanelStatusBar()  
    'Membuat panel dan mengatur properti teks.  
    StatusBar1.Panels.Add("Ready..")  
    StatusBar1.Panels.Add(" ")  
    StatusBar1.Panels.Add(" ")  
    'Mengatur properti panel StatusBar.  
    'Mengatur properti panel AutoSize.  
    StatusBar1.Panels(0).AutoSize = _  
        StatusBarPanelAutoSize.Spring  
    StatusBar1.Panels(1).Width = 200  
    StatusBar1.Panels(2).Width = 200  
    ' Mengatur properti panel BorderStyle.  
    StatusBar1.Panels(0).BorderStyle = _  
        StatusBarPanelBorderStyle.Raised  
    StatusBar1.Panels(1).BorderStyle = _  
        StatusBarPanelBorderStyle.Sunken  
    StatusBar1.Panels(2).BorderStyle = _  
        StatusBarPanelBorderStyle.Sunken  
    ' Mengatur perataan panel  
    StatusBar1.Panels(0).Alignment = _  
        HorizontalAlignment.Left  
    StatusBar1.Panels(1).Alignment = _  
        HorizontalAlignment.Center  
    StatusBar1.Panels(2).Alignment = _  
        HorizontalAlignment.Center  
    'Menambahkan properti teks pada panel ketiga dengan  
    'tanggal(sekarang)  
    StatusBar1.Panels(1).Text = "Selamat Datang" & " " & Username  
  
    StatusBar1.Panels(2).Text = _  
        System.DateTime.Today.ToLongDateString()  
    'Mengatur properti icon pada panel ketiga  
  
    StatusBar1.ShowPanels = True  
End Sub
```

Double_Click Form Utama

```
BuatPanelStatusBar()
```


Setelah itu Double Click menu item Exit dan ketik listing seperti dibawah ini :

```
End
```

Lalu Double Click menu item Inputan dan ketikan listing seperti dibawah ini :

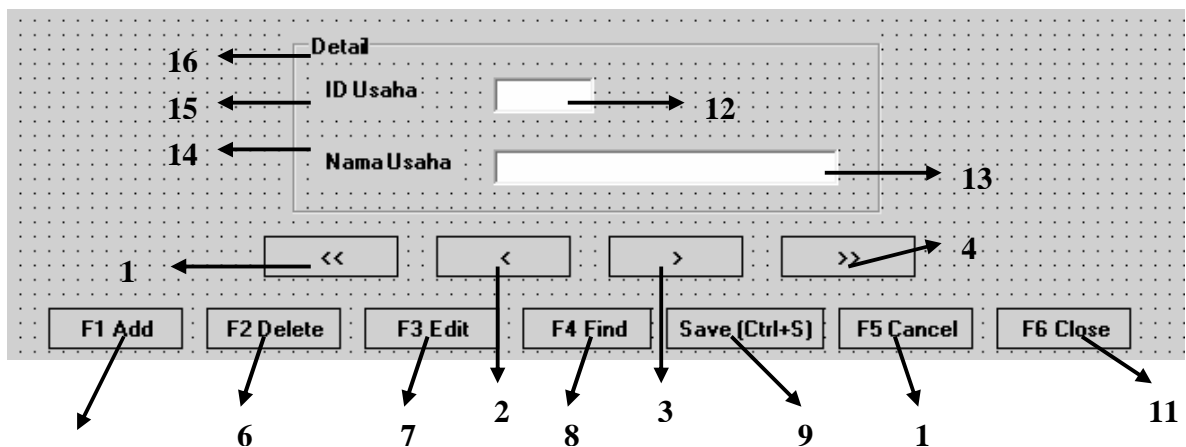
```
Dim f_usaha As New Frm_Usaha  
f_usaha.ShowDialog()  
f_usaha.Dispose()
```

Setelah selesai kita mendesain Form Utama (MainMenu), sekarang kita akan membuat Form Untuk Penginputan Usaha.

Sekarang tambahkan sebuah form baru pada project yang kita buat. Ubah properties form seperti dibawah ini.

BackColor	LightBlue
FormBorderStyle	Fixed3D
Text	Form Usaha
Name	Frm_usaha
StartPosition	CenterScreen
ControlBox	False
ShowInTaksbar	False

Dan tambahkan 11 Button, 1 GroupBox dan 2 TextBox. Susun semua komponen seperti dibawah ini.



5

Dan ubah properties masing-masing komponen seperti dibawah ini :

1.

BackColor	Transparent
FlatStyle	Flat
ForeColor	Black
Text	<<
TextAlign	MiddleCenter
Name	btnAwal
ShowInTaksbar	False

Untuk menghemat waktu dan tempat, properties setiap button hamper sama yang beda Cuma di nama dan text.

2.

Text	<
Name	btnMundur

3.

Text	>
Name	btnMaju

4.

Text	>>
Name	btnAkhir

5.

Text	F1 Add
Name	btnAdd

6.

Text	F2 Delete
Name	btnDelete

7.

Text	F3 Edit
Name	btnEdit

8.

Text	F4 Find
Name	btnCari

9.

Text	Save (Ctrl+S)
Name	btnSave

10.

Text	F5 Cancel
Name	BtnCancel

11.

Text	F6 Close
Name	btnClose

12.

BorderStyle	Fixed3D
Name	txtID
Text	"" (Kosong)

13.

BorderStyle	Fixed3D
Name	txtNama
Text	""(Kosong)

Untuk Nomor 14-16 ganti Text pada properties sesuai pada gambar.

Jika sudah selesai mendesain form tersebut, kita akan lanjutkan membuat class untuk pemrosesan data. Buat satu class didalam folder AccessData dengan nama **AccessUsaha**. Tapi sebelum itu kita melakukan coding pada Class itu, hal yang harus kita lakukan adalah dengan membuat Stored Procedures di SQL Server. Ikuti langkah-langkah dibawah ini. Pada jendela Enterprise Manager klik tanda + pada database Coba, lalu di bawah nya akan terlihat Stored Procedures, klik kanan lalu **New Stored Procedure**. Setelah itu akan muncul jendela Stored Procedure. Dan ketikkan sintaks seperti dibawah ini. Untuk menyimpannya klik button **OK**, sedangkan untuk mengecek syntak dari SQL nya klik button **Check Syntax**.

```
CREATE PROCEDURE [SpInsertUsaha]
    (@IDUsaha [varchar](3),
     @NamaUsaha [varchar](50))

AS INSERT INTO [TbUsaha]
    ([IDUsaha], [NamaUsaha])

VALUES
    ( @IDUsaha, @NamaUsaha)

GO
```

Lalu buat 2 Stored Procedure dan ketikkan sintaks dibawah untuk masing-masing Stored Procedure.

```
CREATE PROCEDURE [SpUpdateUsaha]
    (@IDUsaha [varchar],
     @NamaUsaha [varchar](50))

AS UPDATE [TbUsaha]

SET [NamaUsaha] = @NamaUsaha
WHERE
    ([IDUsaha] = @IDUsaha)

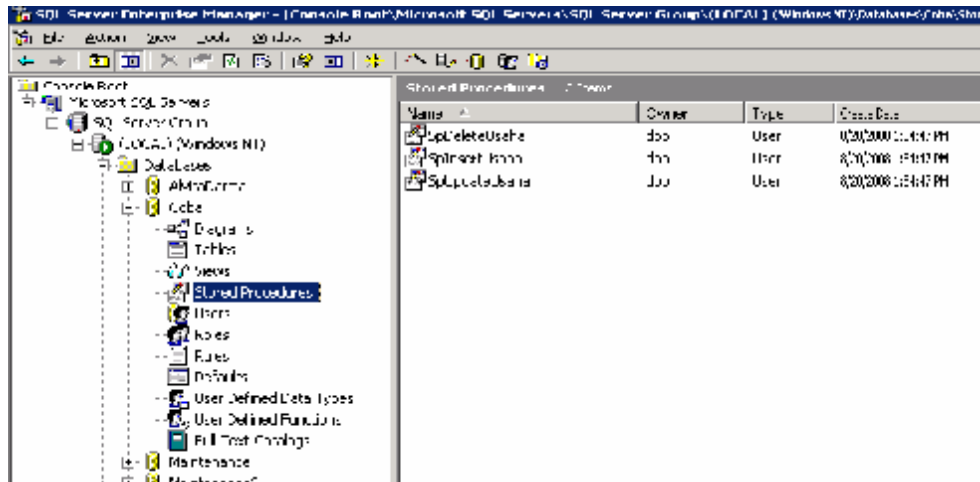
GO
```

```
CREATE PROCEDURE [SpDeleteUsaha]
    (@IDUsaha [varchar])

AS DELETE [TbUsaha]
WHERE
    ([IDUsaha] = @IDUsaha)

GO
```

Setelah melakukan hal diatas dengan benar maka kita akan memiliki 3 Stored Procedures, seperti pada gambar dibawah ini.



Dan kita kembali pada jendela Vb.Net, pada Class **AccessUsaha**, ketikkan sintaks berikut.

```
Imports System.Data
Imports System.Data.SqlClient

Namespace AccessData
    Public Class AccessUsaha
        Public Function AccessUsahaInsert(ByVal IDUsaha As String, _
        ByVal NamaUsaha As String) As SqlDataReader

            Dim MyConnection As DataBaseConnection = New AccessData.DataBaseConnection
            Dim myCommand As SqlCommand = New SqlCommand("SpInsertUsaha", MyConnection.open)

            myCommand.CommandType = CommandType.StoredProcedure

            Dim parameterIdUsaha As SqlParameter = New SqlParameter("@IDUsaha",
            SqlDbType.VarChar, 3)
            parameterIdUsaha.Value = IDUsaha
            Dim parameterNamaUsaha As SqlParameter = New SqlParameter("@NamaUsaha",
            SqlDbType.VarChar, 50)
            parameterNamaUsaha.Value = NamaUsaha

            With myCommand.Parameters
                .Add(parameterIdUsaha)
                .Add(parameterNamaUsaha)
            End With
            Dim result As SqlDataReader =
            myCommand.ExecuteReader(CommandBehavior.CloseConnection)
            Return result
        End Function
    End Class
End Namespace
```

```
Public Function AccessUsahaUpdate(ByVal IDUsaha As String, ByVal NamaUsaha As
String)

    Dim MyConnection As DataBaseConnection = New AccessData.DataBaseConnection
    Dim myCommand As SqlCommand = New SqlCommand("SpUpdateUsaha",
MyConnection.open)

    myCommand.CommandType = CommandType.StoredProcedure

    Dim parameterIdUsaha As SqlParameter = New SqlParameter("@IDUsaha",
SqlDbType.VarChar, 3)
    parameterIdUsaha.Value = IDUsaha

    Dim parameterNamaUsaha As SqlParameter = New SqlParameter("@NamaUsaha",
SqlDbType.VarChar, 50)
    parameterNamaUsaha.Value = NamaUsaha

    With myCommand.Parameters
        .Add(parameterIdUsaha)
        .Add(parameterNamaUsaha)
    End With
    Dim result As SqlDataReader =
myCommand.ExecuteReader(CommandBehavior.CloseConnection)
    Return result
End Function

Public Function AccessUsahaDelete(ByVal IDUsaha As String)

    Dim MyConnection As DataBaseConnection = New AccessData.DataBaseConnection
    Dim myCommand As SqlCommand = New SqlCommand("SpDeleteUsaha",
MyConnection.open)

    myCommand.CommandType = CommandType.StoredProcedure

    Dim parameterIdUsaha As SqlParameter = New SqlParameter("@IDUsaha",
SqlDbType.VarChar, 3)
    parameterIdUsaha.Value = IDUsaha

    With myCommand.Parameters
        .Add(parameterIdUsaha)
    End With
    Dim result As SqlDataReader =
myCommand.ExecuteReader(CommandBehavior.CloseConnection)
    Return result
End Function

End Class
End Namespace
```

Jika sudah selesai, mari kita lanjutkan membuat program di dalam Form Usaha.

Dan buat variable global form seperti dibawah ini.

```
#Region "deklarasi"  
    Dim MyConnection As New AccessData.DataBaseConnection  
    Dim objConnection As SqlConnection  
    Dim objCommand As SqlCommand  
    Dim objDataAdapter As SqlDataAdapter  
    Dim objdatareader As SqlDataReader  
    Dim strSQL As String  
    Dim objDataSet As New DataSet  
    Dim objDataTable As New DataTable  
    Public mProcess As String  
    Dim mPosition As Integer  
#End Region
```

Lalu ketikkan fungsi-fungsi berikut :

```
Private Sub clear()  
    'Mengkosongkan isi TextBox  
    txtID.Text = ""  
    txtNama.Text = ""  
End Sub
```

```
Private Sub enabled_off()  
    'Menonaktifkan Textbox  
    txtID.Enabled = False  
    txtNama.Enabled = False  
End Sub
```

```
Private Sub enabled_on()  
    'Mengaktifkan TextBox  
    txtID.Enabled = True  
    txtNama.Enabled = True  
End Sub
```

```
Sub ViewData()  
  
    'Menampilkan Data IDUsaha dan Nama Usaha dengan Criteria "IDUsaha"  
    pada masing-masing TextBox  
    objCommand = MyConnection.open.CreateCommand  
    objCommand.CommandText = "select * from TbUsaha " & _  
    " Where IDUsaha = '" & Trim(txtID.Text) & "'" & _  
    objdatareader = objCommand.ExecuteReader  
    While objdatareader.Read  
        txtID.Text = objdatareader.Item("IDUsaha").ToString  
        txtNama.Text = objdatareader.Item("NamaUsaha").ToString  
    End While  
    MyConnection.close  
End Sub
```

```
Sub Displaydata()  
    Try  
        'Menyimpan jumlah baris Tabel "TbUsaha" pada Objdatatable  
        objDataTable.Clear()  
        StrSQL = "select * from TbUsaha"  
        objCommand = New SqlCommand  
        objCommand.Connection = MyConnection.open  
        objCommand.CommandType = CommandType.Text  
        objCommand.CommandText = StrSQL  
        objDataAdapter = New SqlDataAdapter(objCommand)  
        objDataAdapter.Fill(objDataSet, "MDT_Usaha")  
        MyConnection.close()  
        objDataTable = objDataSet.Tables("MDT_Usaha")  
    Catch ex As Exception  
        MsgBox("Tidak dapat membuat hubungan ke database" _  
            & vbCrLf & Err.Description)  
        MyConnection.close()  
    End Try  
End Sub
```

```
Sub displaydata2()  
    Try  
        'Menampilkan Data "IDUsaha" dan "NamaUsaha"  
        berdasarkan posisi Baris yang dicari  
        StrSQL = "select * from TbUsaha"  
        objCommand = New SqlCommand  
        objCommand.Connection = MyConnection.open  
        objCommand.CommandType = CommandType.Text  
        objCommand.CommandText = StrSQL  
        objDataAdapter = New SqlDataAdapter(objCommand)  
        objDataAdapter.Fill(objDataSet, "MDT_Usaha")  
        MyConnection.close()  
        objDataTable = objDataSet.Tables("MDT_Usaha")  
        mPosition = BindingContext(objDataTable).Position  
        With objDataTable.Rows(mPosition)  
            txtID.Text = .Item("IDUsaha")  
            txtNama.Text = .Item("NamaUsaha")  
        End With  
    Catch When Err.Number <> 0  
        MsgBox("Tidak dapat membuat hubungan ke database" _  
            & vbCrLf & Err.Description)  
        MyConnection.close()  
    End Try  
End Sub
```



```
Sub cari()  
  Try  
    'untuk mengecek apakah IDUsaha  
    yang diinputkan ada pada Tabel "TbUsaha"  
    objDataTable.Clear()  
    strSQL = "select * from [TbUsaha] where " & _  
    " IDUsaha = '" & Trim(txtID.Text) + "'"  
    objCommand = New SqlCommand  
    objCommand.Connection = MyConnection.open  
    objCommand.CommandType = CommandType.Text  
    objCommand.CommandText = strSQL  
    objDataAdapter = New SqlDataAdapter(objCommand)  
    objDataAdapter.Fill(objDataSet, "MDT_Usaha")  
    MyConnection.close()  
    objDataTable = objDataSet.Tables("MDT_Usaha")  
  Catch When Err.Number <> 0  
    MsgBox("Tidak dapat membuat hubungan ke database" _  
    & vbCrLf & Err.Description)  
    MyConnection.close()  
  End Try  
End Sub
```

Double Click **"btnclose"** dan ketik seperti dibawah ini :

```
objDataTable.Clear()  
Me.Close()
```

Double Click **"btncancel"** dan ketik seperti dibawah ini :

```
enabled_off()  
clear()  
mProcess = ""  
btncancel.Enabled = False  
btncancel.Text = "Save"
```

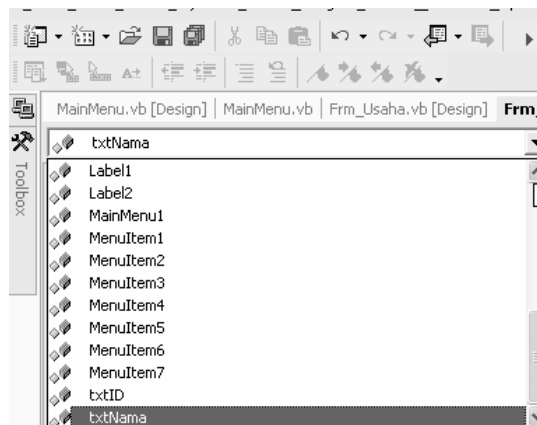
Double Click **"btnadd"** dan ketik seperti dibawah ini :

```
mProcess = "ADD"  
btncancel.Text = "Save Add"  
enabled_on()  
btncancel.Enabled = True  
txtID.Focus()  
clear()
```

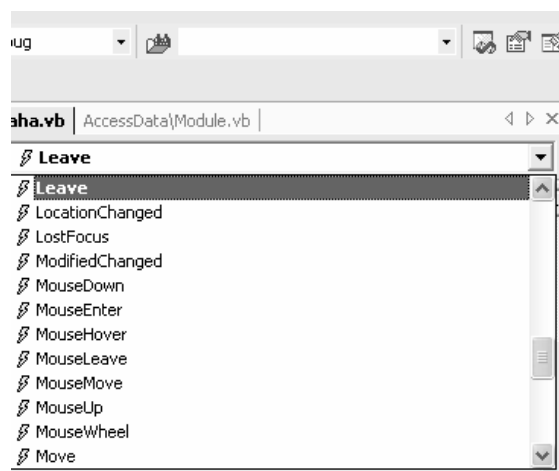
Double Click **"btnEdit"** dan ketik seperti dibawah ini :

```
If mProcess <> "View" Or txtNama.Text = "" Then
    MsgBox("Lakukan pencarian terlebih dahulu, sebelum proses edit")
Else
    enabled_on()
    txtNama.Focus()
    mProcess = "EDIT"
    btnsave.Text = "Save Edit"
    btnsave.Enabled = True
End If
```

Pilih ClassName = **txtNama**, Seperti dibawah ini



Dan Pilih Method Name = **Leave**. Untuk selanjutnya penulis hanya menyebutkan ClassName dan MethodName saja, untuk visual tidak akan diberikan.



Lalu ketik seperti dibawah ini.

```
btnsave.Focus()
```

Double Click Form Usaha, dan ketik seperti berikut :

```
Try
    strSQL = "Select * From TbUsaha"
    objCommand = New SqlCommand
    objCommand.Connection = MySqlConnection.open
    objCommand.CommandType = CommandType.Text
    objCommand.CommandText = strSQL
    objDataAdapter = New SqlDataAdapter(objCommand)
    objDataAdapter.Fill(objDataSet, "MDT_Usaha")
    MySqlConnection.close()
    objDataTable = objDataSet.Tables("MDT_Usaha")
Catch When Err.Number <> 0
    MsgBox("Tidak dapat membuat hubungan ke database" _
    & vbCrLf & Err.Description)
End Try
clear()
enabled_off()
```

Double Click btncari, dan ketik seperti berikut :

```
mProcess = "View"
txtID.Enabled = True
txtID.Focus()
```

Double Click btnSave, dan ketik seperti berikut :

```
If btnsave.Text = "Save Add" Then
    Try
        cari()
        If objDataTable.Rows.Count >= 1 Then
            MsgBox(" Kode " & "" & txtID.Text & "" & " sudah ada")
            txtID.Focus()
        ElseIf txtID.Text = "" Or txtNama.Text = "" Then
            MsgBox("Isikan data yang mau di Input terlebih dahulu")
            txtID.Focus()
        Else
            Dim mYes_No As String = MsgBox("Anda yakin data mau disimpan ?", vbYesNo)
            If mYes_No = vbYes Then
                myconnectionUsaha.AccessUsahaInsert(txtID.Text, txtNama.Text)
                enabled_off()
                btnsave.Enabled = False
                btnsave.Text = "Save"
                btnadd.Focus()
                Displaydata()
                clear()
            Else
                enabled_off()
                clear()
                btnadd.Focus()
            End If
        End If
    Catch When Err.Number <> 0
        MsgBox("Tidak dapat melakukan proses simpan" _
            & vbCrLf & Err.Description)
    End Try
Else
    Try
        cari()
        If objDataTable.Rows.Count = 0 Then
            MsgBox(" Kode : " & "" & txtID.Text & "" & " tidak ada pada table")
        Else
            Dim mYes_No As String = MsgBox("Anda yakin data mau diupdate ?", vbYesNo)
            If mYes_No = vbYes Then
                myconnectionUsaha.AccessUsahaUpdate(txtID.Text, txtNama.Text)
                enabled_off()
                btnsave.Text = "Save"
                btnsave.Enabled = False
                btnadd.Focus()
                clear()
            Else
                enabled_off()
                clear()
                btnadd.Focus()
            End If
        End If
    Catch When Err.Number <> 0
        MsgBox("Tidak dapat melakukan proses update" _
            & vbCrLf & Err.Description)
    End Try
End If
objDataTable.Clear()
```

Pilih ClassName = **txtNama**
Dan Pilih Method Name = **KeyUp**

```
If e.KeyCode = Keys.Enter Then  
    btnsave.PerformClick()  
ElseIf e.KeyCode = Keys.Escape Then  
    btncancel.PerformClick()  
End If
```

Pilih ClassName = **txtNama**, Seperti dibawah ini
Dan Pilih Method Name = **KeyUp**

```
If mProcess = "View" Then  
    Try  
        cari()  
        If objDataTable.Rows.Count = 0 Then  
            MsgBox("Data yang dicari tidak ada")  
        Else  
            ViewData()  
            btndelete.Focus()  
        End If  
    Catch When Err.Number <> 0  
        MsgBox("Tidak dapat melakukan proses pencarian" _  
            & vbCrLf & Err.Description)  
        MyConnection.close()  
    End Try  
End If
```

Double_Click **btnakhir**

```
BindingContext(objDataTable).Position = objDataTable.Rows.Count - 1  
displaydata2()
```

Double_Click **btnmaju**

```
BindingContext(objDataTable).Position += 1  
displaydata2()
```

Double_Click **btnmundur**

```
BindingContext(objDataTable).Position -= 1  
displaydata2()
```

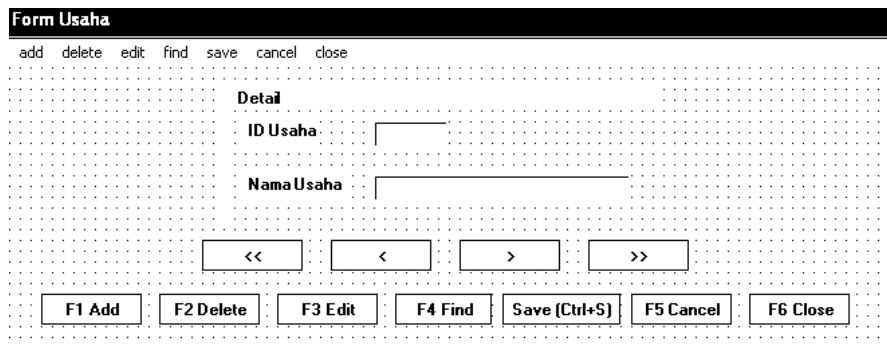
Double_Click **btnawal**

```
BindingContext(objDataTable).Position -= 1  
displaydata2()
```

Sekarang coba anda jalankan program, apakah program sudah berjalan dengan semestinya, jika belum bisa seperti biasa.....

Tentu anda akan merasa heran, kenapa pada button Add, Edit dll ada semacam shortcut. Tetapi jika digunakan shortcut-shortcut tersebut tidak bekerja. Mungkin merupakan suatu kelemahan pada VB.Net tidak menyediakan shortcut untuk component button, tidak seperti Delphi yang menyediakan shortcut untuk button . Tapi jika anda seorang programmer yang memanfaatkan kelebihan dari bahasa pemrograman, masalah itu semua akan bisa diatasi. Ok, sekarang penulis akan memberikan solusi gimana cara membuat shortcut untuk menjalankan perintah pada suatu button.

Tambahkan komponen MainMenu pada Form Usaha. Lalu buat seperti berikut :



Pada masing-masing Properties MenuItem, buat **Visible=False**. Dan shortcut sesuai dengan button. Contoh : Untuk Button dengan Text : **F1 Add**, pada menu item add properties>Shortcut = F1. Dan selanjutnya..Buat sendiri ya..... ???????. Lalu coba Double_Click menu item add, dan ketik listing seperti dibawah.

```
btnadd.PerformClick()
```

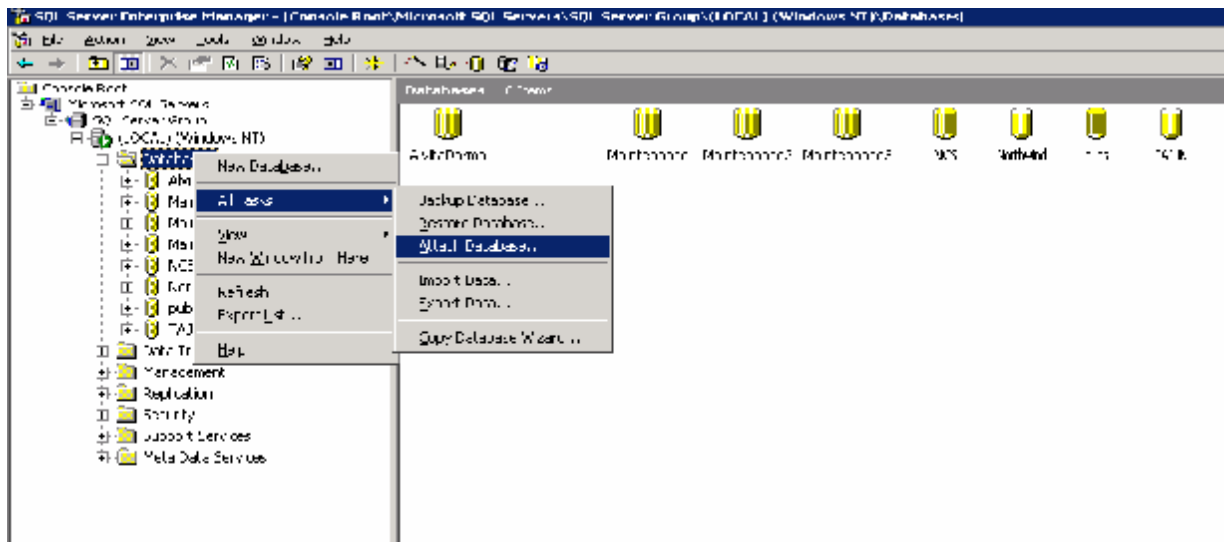
Untuk semua menu item buat listing seperti diatas, hanya ubah button apa yang akan diproses. Coba jalankan program dan gunakan shortcut. Apa sudah jalan ??????

Kalau berhasil, berarti anda sudah bisa proses dasar dari database (Select, Insert, Delete & Update). Dengan begitu anda dapat mengembangkan program ini sesuai dengan keinginan anda.

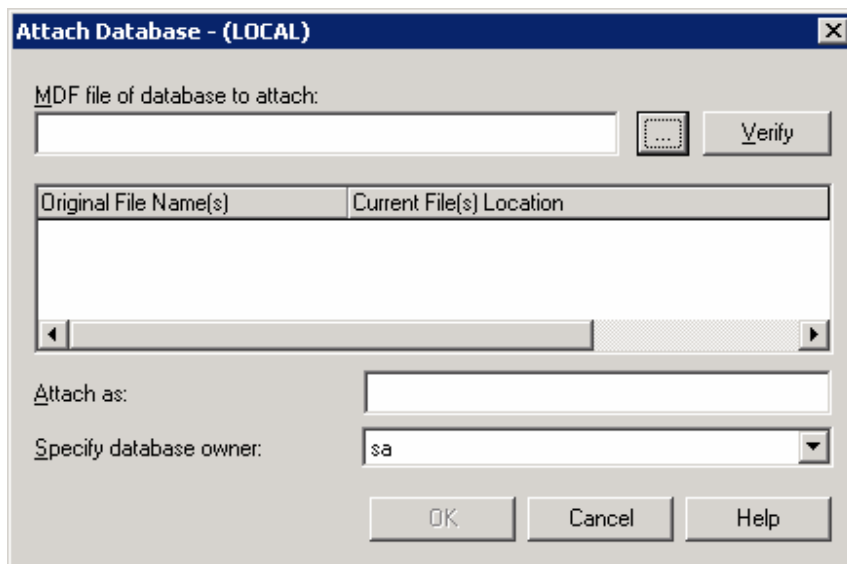
Untuk mendapatkan source code lengkap dapat di download pada bog saya

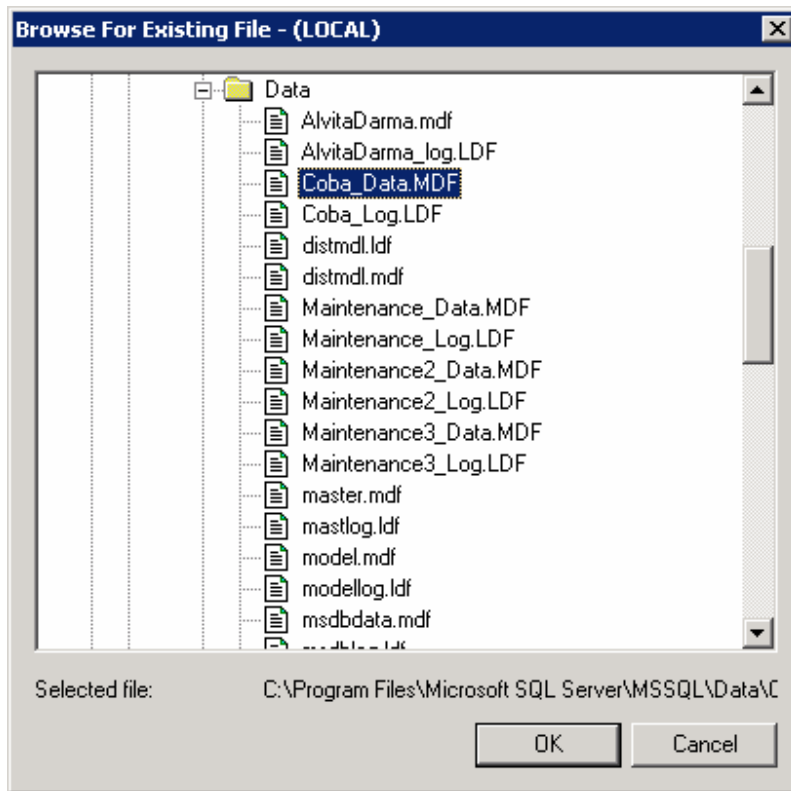
<http://junindar.blog.com> . Untuk menjalankan program yang didownload. Unzip file

tersebut. Lalu copy 2 file yang berada pada folder DB ke C:\Program Files\Microsoft SQL Server\MSSQL\Data. Lalu buka Enterprise Manager, klik kanan pada Folder Database All Tasks >> Attach Database

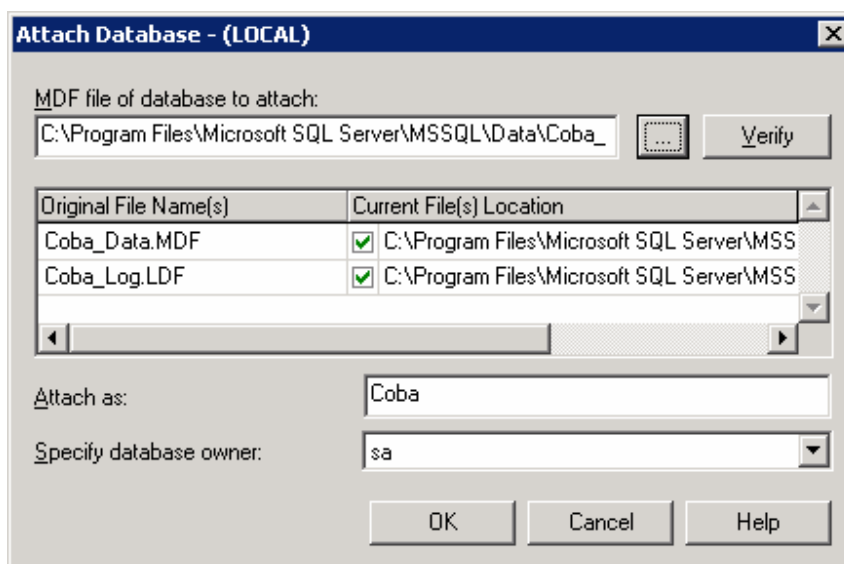


Lalu akan muncul jendela seperti dibawah. Dan klik button **Browse** disamping button **Verify**





Akan muncul jendela seperti diatas, lalu cari file pada path dimana kita menyimpan file DB (C:\Program Files\Microsoft SQL Server\MSSQL\Data). Setelah ketemu klik file Coba_Data.MDF dan klik button **OK**. Maka jendela Attach Database akan terisi sendiri seperti gambar dibawah ini.



Dan klik OK, maka didalam Folder Database di Enterprise Manager akan terdapat database baru dengan nama Coba. Setelah langkah diatas berhasil maka program yang di download juga dapat dijalankan. Username :jun Password :123

Penutup

Maaf kepada para pembaca, kalau artikel ini baru saat ini dapat dirilis. Mudah-mudahan artikel ini dapat sedikit menjawab dari persoalan yang dihadapi para pembaca.

Referensi

1. www.msdn.microsoft.com
2. www.planetsourcecode.com
3. www.codeproject.com
4. www.aspnet.com

Masih banyak lagi referensi yang ada di Intenet. Anda tinggal cari di www.Google.com. Dengan kata kunci “**tutorial VB.Net**”



Sinopsis buku “Panduan Lengkap Menjadi Programmer”

Setelah mengetahui apa itu Microsoft.NET Framework, Microsoft Visual Basic.NET, ADO.NET, Database dan Microsoft Access, dan fungsi-fungsi serta dasar penggunaannya, ternyata menjadi programmer VB.NET itu mudah! Sebab, dari dasar pengetahuan tersebut, kita sudah bisa membuat suatu program yang kita inginkan. Misalnya, membuat aplikasi toko, seperti form penjualan dan pembelian. Visual Basic.NET (VB.NET) merupakan program generasi selanjutnya dari Visual Basic. VB.NET memungkinkan kita untuk membangun aplikasi database client atau server dengan performa tinggi.

Umumnya, para programmer banyak menggunakan VB.NET untuk membuat aplikasi atau program yang mereka butuhkan. Pasalnya, program ini memiliki kelebihan dan kemudahan tertentu dibandingkan dengan program-program lainnya yang sejenis.

Selanjutnya, langkah apa saja yang mesti diketahui dan dilakukan agar kita bisa menjadi programmer VB.NET? Mediakita menerbitkan buku *Panduan Lengkap Menjadi Programmer* yang ditulis oleh **Junindar**. Buku ini berisi tutorial pembuatan program, pembuatan laporan, sampai dengan melakukan *deployment* program. Melalui buku ini, Anda bisa menjadi seorang programmer VB.NET yang handal. Sebab, buku ini merupakan buku yang komplit untuk mengantar seseorang dari tidak tahu apa-apa tentang VB.NET menjadi seorang programmer.

Step By Step Menjadi Programmer Handal Dengan VB.Net XIII
Junindar

Memang, saat ini, banyak sekali buku-buku tentang pemrograman VB.NET, tetapi sangat jarang yang menjelaskan secara detail tentang fungsi-fungsi dan pemrograman di dalam VB.NET. Di buku ini, secara terperinci Anda akan dibimbing bagaimana membuat database, membuat program (*create program*), tampilan login, menu, form, laporan, sampai dengan membuat file setup (*installer*).

Selain itu, buku ini juga dilengkapi CD berisikan *source code* pembuatan program sehingga lebih memudahkan Anda dalam berlatih.

Biografi Penulis.



Junindar Lahir di Tanjung Pinang, 21 Juni 1982. Menyelesaikan Program S1 pada jurusan Teknik Informatika di Sekolah Tinggi Sains dan Teknologi Indonesia (ST-INTEN-Bandung). Senang mengutak-atik computer yang berkaitan dengan bahasa pemrograman. Keahlian, sedikit mengerti beberapa bahasa pemrograman seperti : VB.Net, ASP.NET, VBA. Reporting : Crystal Report (Under Program). Database : Access, paradox dan SQL Server 2000. Simulation / Modeling Packages: Visio Enterprise, Rational Rose dan Power Designer. Dan senang bermain gitar, karena untuk bisa menjadi pemain gitar dan seorang programmer sama-sama membutuhkan seni. Pada saat ini bekerja di salah satu Perusahaan Swasta di BATAM sebagai Software Development. Mempunyai moto hidup : **“Jauh lebih baik menjadi Orang Bodoh yang giat belajar, dari pada orang Pintar yang tidak pernah mengimplementasikan ilmunya”**.

*Kritik dan saran kirim ke : xx_lead@yahoo.com
junindar@gmail.com*