

# 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, ST, MCPD, MOS, MCT, MVP VB.NET

## DAPPER (Micro ORM) dan Relationships Table Pada VB 2013

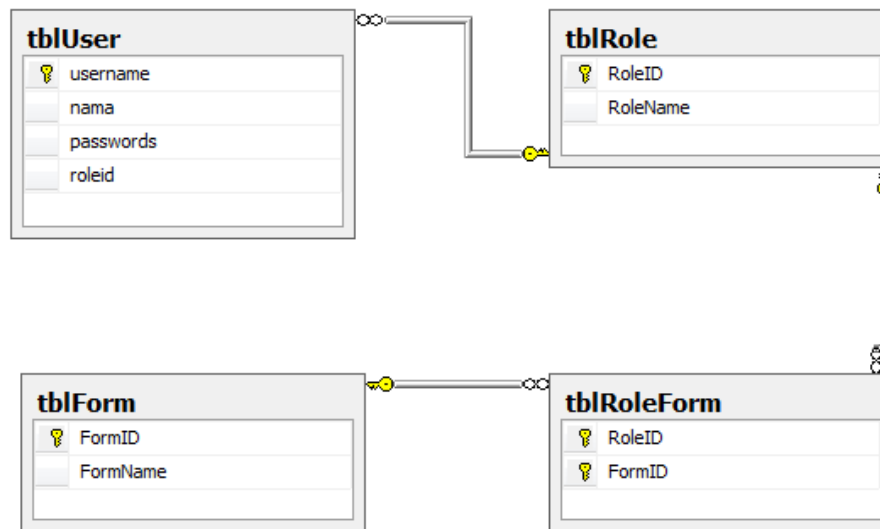
Setelah artikel pengenalan dan bagaimana penggunaan stored procedure Dapper, sekarang kita akan melanjutkan artikel bagaimana menggunakan Dapper untuk table yang berelasi. Sebaiknya untuk lebih memahami artikel ini, disarankan membaca dan mempraktekkan pada artikel sebelumnya disini (<http://junindar.blogspot.com/2014/12/pengenalan-dapper-micro-orm-pada-vb-2013.html> dan <http://junindar.blogspot.com/2015/01/dapper-micro-orm-dan-stored-procedure.html> ). Pada dua artikel sebelumnya telah dijelaskan bagaimana penggunaan Dapper dengan Visual Basic dan disertai dengan sample source code (untuk source code juga disediakan Bahasa pemrograman C#). Jika kita perhatikan pada artikel-artikel sebelumnya mengenai Dapper ini, penulis hanya menggunakan single table sebagai contoh latihan kita. Lalu

bagaimana jika kita mau menggunakan Dapper pada table yang berelasi, apakah sintak yang digunakan berbeda atau sama?

Pada artikel ini akan dijelaskan bagaimana membuat aplikasi dengan menggunakan Dapper pada table yang memiliki relasi. Diasumsikan para pembaca telah mengerti dan menyelesaikan latihan pada artikel sebelumnya (Pengenalan Dapper dan penggunaan stored procedure Dapper) sehingga akan lebih mudah dalam mengikuti dan menyelesaikan latihan pada artikel ini.


Untuk mengetahui lebih lanjut, saya akan menjelaskan bagaimana membuat aplikasi dengan menggunakan Dapper pada table berelasi .


Sebelum kita membuat aplikasi, terlebih dahulu kita buat design table yang akan diperlukan untuk membuat aplikasi. Untuk latihan kali ini, kita akan menggunakan 4 (empat) table yang saling berelasi, seperti pada gambar dibawah ini.





Buat 4 table seperti dibawah ini.

| tblForm |             |             |                                     |
|---------|-------------|-------------|-------------------------------------|
|         | Column Name | Data Type   | Allow Nulls                         |
|         | FormID      | varchar(3)  | <input type="checkbox"/>            |
|         | FormName    | varchar(50) | <input checked="" type="checkbox"/> |
|         |             |             | <input type="checkbox"/>            |

| tblRole   |             |             |                                     |
|---|-------------|-------------|-------------------------------------|
|   | Column Name | Data Type   | Allow Nulls                         |
|  | RoleID      | int         | <input type="checkbox"/>            |
|   | RoleName    | varchar(15) | <input checked="" type="checkbox"/> |
|   |             |             | <input type="checkbox"/>            |

| tblUser   |             |             |                                     |
|---|-------------|-------------|-------------------------------------|
|   | Column Name | Data Type   | Allow Nulls                         |
|  | username    | varchar(15) | <input type="checkbox"/>            |
|   | nama        | varchar(50) | <input checked="" type="checkbox"/> |
|   | passwords   | varchar(15) | <input checked="" type="checkbox"/> |
|   | roleid      | int         | <input checked="" type="checkbox"/> |
|   |             |             | <input type="checkbox"/>            |

| tblRoleForm   |             |            |                          |
|---|-------------|------------|--------------------------|
|   | Column Name | Data Type  | Allow Nulls              |
|  | RoleID      | int        | <input type="checkbox"/> |
|  | FormID      | varchar(3) | <input type="checkbox"/> |
|   |             |            | <input type="checkbox"/> |

Selanjutnya jadikan kolom “roleid” di “tblUser” menjadi foreign key yang berelasi dengan “tblRole” dimana “RoleID” menjadi primary key. Dan lakukan hal sama untuk pada “tblRoleForm pada kolom “RoleID”, sedangkan untuk kolom FormID buat relasi dengan “tblForm” dimana “FormID” sebagai primary key.

Selanjutnya isikan data pada masing-masing table diatas, seperti gambar dibawah ini.

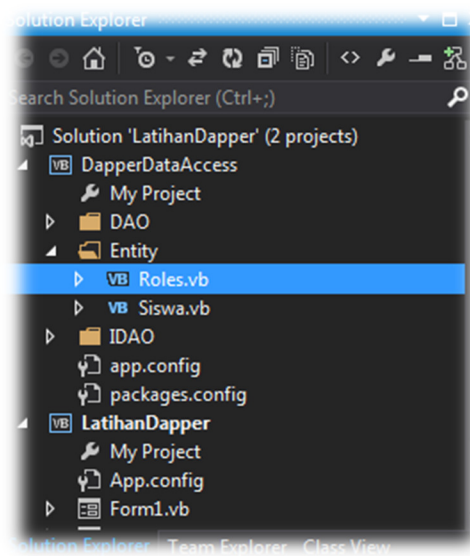
|  | RoleID | RoleName      |
|--|--------|---------------|
|  | 1      | Administrator |
|  | 2      | Approver      |
|  | 3      | Viewer        |
|  | 4      | Tester        |
|  | 11     | Operator      |

|  | FormID | FormName |
|--|--------|----------|
|  | 001    | Form A   |
|  | 002    | Form B   |
|  | 003    | Form C   |
|  | 004    | Form D   |
|  | 005    | Form E   |
|  | 006    | Form F   |
|  | 007    | Form G   |
|  | 008    | Form H   |
|  | 009    | Form I   |
|  | 010    | Form J   |

|  | username | nama     | passwords | roleid |
|--|----------|----------|-----------|--------|
|  | ahmad    | Ahmad    | 54321     | 11     |
|  | aris     | Aris     | 12345     | 2      |
|  | budi     | Budi Man | 5321      | 2      |
|  | dodo     | Dodo     | 12345     | 3      |
|  | jun      | Junindar | 12345     | 1      |
|  | lala     | Lala     | 12345     | 3      |
|  | yudi     | Yudi     | 12345     | 3      |

| RoleID | FormID |
|--------|--------|
| 1      | 001    |
| 1      | 002    |
| 1      | 003    |
| 1      | 004    |
| 1      | 005    |
| 1      | 006    |
| 1      | 007    |
| 1      | 008    |
| 1      | 009    |
| 1      | 010    |

- Buka Solution sebelumnya (Latihan Dapper). Pada project “DapperDataAccess”, tambahkan satu buah Class pada folder entity dengan nama “Roles”.



Pada Roles.vb ini kita akan membuat 4 buah Class yang mana akan merepresentasikan table yang telah dibuat sebelum nya. Jika pada table terdapat relasi, pada class-class yang akan kita buat pun akan terdapat relasi antar class yang disebut “asosiasi”. Ketikkan sintak dibawah ini pada Roles.vb

```
Public Class Roles
    Public Property RoleId() As Int16
    Public Property RoleName() As String
End Class

Public Class Forms
    Public Property FormId() As String
    Public Property FormName() As String
End Class

Public Class RoleForm
    Public Property RoleId() As Int16
    Public Property FormId() As String
End Class

Public Class Users

    Public Property Role() As Roles
    Public Property Username() As String
    Public Property Nama() As String
    Public Property Password() As String

End Class
```

Perhatikan pada class “Users” terdapat sebuah property dimana kita melakukan relasi dengan class “Roles” yang disebut asosiasi. Yang artinya sebuah class dapat mengetahui property atau method dari class lain dengan syarat harus memiliki access modifier/visibility public.

- Selanjutnya tambahkan sebuah interface pada folder “IDAO” dengan nama “IRoleDAO”. Lalu ketikkan syntax berikut pada interface yang baru dibuat.

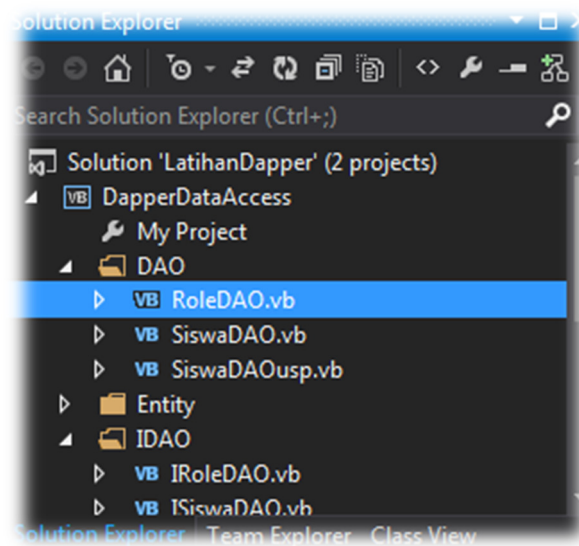
```
Public Interface IRoleDAO

    Function SelectRoleFormByRoleId(ByVal RoleName As String) As List(Of RoleForm)
    Function SelectForms() As List(Of Forms)
    Function SelectRole() As List(Of Roles)
    Function SelectUserByRoleName(ByVal RoleName As String) As List(Of Users)
    Sub DeleteFormByRoleId(ByVal RoleID As String)
    Sub InsertRoleForm(ByVal Entity As RoleForm)

End Interface
```

Untuk artikel ini kita akan menggunakan 6 method yang akan di jelaskan pada bagian selajutnya.

- Buat sebuah class pada folder DAO dengan nama “RoleDAO.vb”.



- Pada code “Class RoleDAO”, ketikkan “Implements IRoleDAO” dan enter. Maka secara otomatis akan dibuat method sesuai dengan Interface diatas.

- Masih pada class yang sama import SqlClient dan Dapper seperti dibawah.

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

- Selanjutnya ketikkan code dibawah, pada masing-masing method pada class RoleDAO.

```
Public Function SelectRoleFormByRoleId(ByVal RoleID As String) As _
    List(Of RoleForm) Implements IRoleDAO.SelectRoleFormByRoleId
    Using mycon As SqlConnection = New
        SqlConnection(My.Settings.myConDapper)
        Const strQuery As String = "SELECT * FROM tblRoleForm Where
        RoleID=@RoleID Order By FormID"
        mycon.Open()
        Return mycon.Query(Of RoleForm)(strQuery, New With
        {RoleID})
    End Using
End Function
```

Method ini berfungsi untuk mengambil data pada tblRoleForm dengan kriterianya adalah RoleID, dan hasil dari pencarian akan disimpan kedalam List of Class.

```
Public Function SelectForms() As List(Of Forms) _
    Implements IRoleDAO.SelectForms
    Using mycon As SqlConnection = New
        SqlConnection(My.Settings.myConDapper)
        Const strQuery As String = "SELECT * FROM tblForm Order By
FormID"
        mycon.Open()
        Return mycon.Query(Of Forms)(strQuery)
    End Using
End Function

Public Function SelectRole() As List(Of Roles) _
    Implements IRoleDAO.SelectRole
    Using mycon As SqlConnection = New
        SqlConnection(My.Settings.myConDapper)
        Const strQuery As String = "SELECT * FROM tblRole Order By
RoleID"
        mycon.Open()
        Return mycon.Query(Of Roles)(strQuery)
    End Using
End Function

Public Function SelectUserByRoleName(ByVal RoleName As String) _
    As List(Of Users) Implements IRoleDAO.SelectUserByRoleName
    Using mycon As SqlConnection = New
        SqlConnection(My.Settings.myConDapper)
        Const strQuery As String = "SELECT      tblUser.username,
tblUser.nama, " & _
        " tblUser.passwords as [Password], tblRole.RoleID,
tblRole.RoleName " & _
        " FROM tblUser INNER JOIN tblRole ON tblUser.roleid =
tblRole.RoleID " & _
        " Where tblRole.RoleName=@RoleName Order By
tblUser.username"
        mycon.Open()

        Return mycon.Query(Of Users, Roles, Users)(strQuery,
Function(users, roles)

users.Role = roles

Return users
Function,
With {RoleName},
splitOn:="username,RoleID")

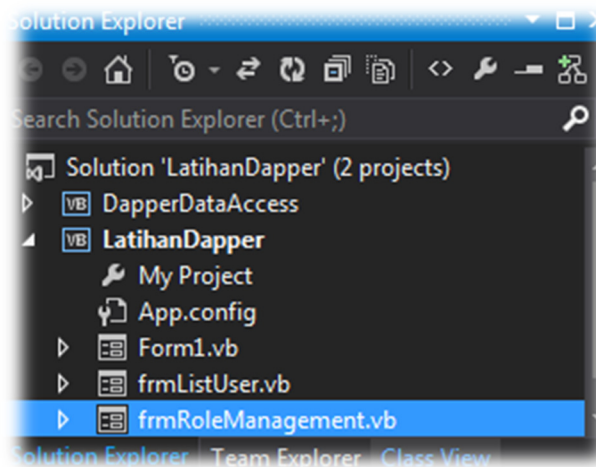
    End Using
End Function
```



```
Public Sub DeleteFormByRoleID(ByVal RoleID As String) _  
    Implements IRoleDAO.DeleteFormByRoleID  
    Using mycon As SqlConnection = New  
        SqlConnection(My.Settings.myConDapper)  
        Const strQuery As String = "Delete From tblRoleForm Where  
        RoleID=@RoleID"  
        mycon.Open()  
        mycon.Query(strQuery, New With {RoleID})  
    End Using  
End Sub  
  
Public Sub InsertRoleForm(ByVal Entity As RoleForm) _  
    Implements IRoleDAO.InsertRoleForm  
    Using mycon As SqlConnection = New  
        SqlConnection(My.Settings.myConDapper)  
        Const strQuery As String = "Insert Into tblRoleForm  
        (RoleID,FormID) " &  
        " Values (@RoleID,@FormID)"  
        mycon.Open()  
        mycon.Query(strQuery, Entity)  
    End Using  
End Sub
```

Sekarang kita telah selesai membuat Class RoleDAO dan langkah selanjutnya adalah bekerja pada layer User Interface.

- Membuat User Interface pada aplikasi.
- Tambahkan sebuah form dengan nama “frmRoleManagement”.



- Tambahkan 2 buah RadioButton pada frmRoleManagement. Dengan properties masing-masing sebagai berikut.

1. Name = rbUncheckedAll

Text = Unchecked All Node

Checked = True

2. Name = rbCheckedAll

Text = Checked All Node

- Tambahkan dua buah button, dengan masing-masing properties sebagai berikut :

1. Name = btnSave

Text = Save

2. Name = btnShow

Text = Show User List

- Lalu drag control “ImageList”, masukkan gambar (ico) pada properties images.

- Tambahkan sebuah ListBox dan ganti properties Name menjadi “lstRoles”.

- Selanjutnya tambahkan sebuah TreeView dengan properties sebagai berikut

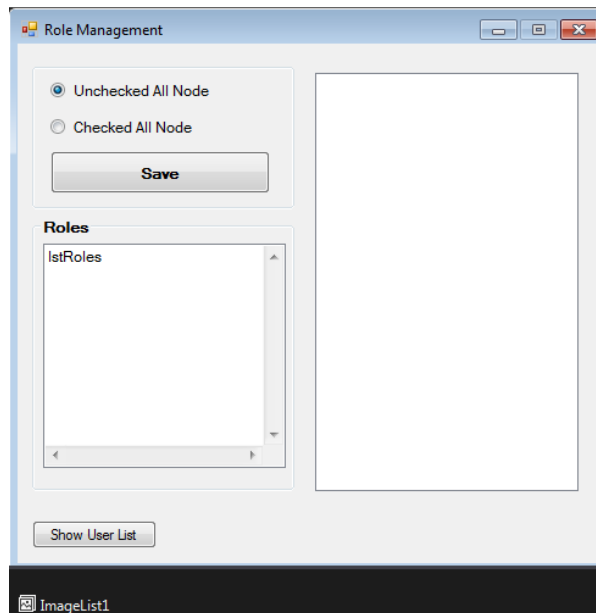
Name = tvUserRights

ImageList = ImageList1

ImageIndex = 0

SelectedImageIndex = 0

- Selanjutnya atur posisi masing-masing control seperti gambar dibawah ini.



- Import namespace GenericDataAccess.

```
Imports DapperDataAccess
```

- Selanjutnya ketikkan code seperti dibawah.

```
Dim MyAccessRole As IRoleDAO = New RoleDAO()

Private Sub PopulateRolesToListBox()
    lstRoles.Items.Clear()
    Dim RoleList As List(Of Roles)
    RoleList = MyAccessRole.SelectRole()
    For Each items In RoleList
        lstRoles.Items.Add(items.RoleID & " > " & items.RoleName)
    Next
End Sub

Private Sub AddFormNameToTree()
    Dim FormList As List(Of Forms)
    FormList = MyAccessRole.SelectForms()

    For Each items In FormList
        Dim aNode As New TreeNode(items.FormName.ToString)
        aNode.Tag = items.FormId.ToString
        tvUserRights.Nodes.Add(aNode)
    Next
End Sub
```

Method `PopulateRolesToListBox` digunakan untuk mengambil data (RoleID dan RoleName) pada `tblRole` dan ditampilkan pada `ListBox` (`lstRoles`) dengan memanggil Method “`SelectRole`” pada class `RoleDAO`. Sedangkan method `AddFormNameToTree` yaitu mengambil data pada table `Form` (`tblForm`) dan ditampilkan pada `TreeView` (`tvUserRights`). Untuk method ini memanggil “`SelectForm`” pada class `RoleDAO`.

```
Private Sub PopulateItemCheckedToTheTree _
    (ByVal RoleForm As List(Of RoleForm))
    Dim IEnum As IEnumerator = tvUserRights.Nodes.GetEnumerator
    While IEnum.MoveNext
        Dim aNode As TreeNode = DirectCast(IEnum.Current, TreeNode)
        Dim result = From o In RoleForm Where
o.FormId.Equals(aNode.Tag.ToString)

        If (result.Count > 0) Then
            aNode.Checked = True
        Else
            aNode.Checked = False
        End If
    End While
End Sub
```

```
Private Sub PopulateRoleDetails(ByVal RoleIDs As String)
    Dim RoleForm As List(Of RoleForm)
    'menampilkan data Form berdasarkan Role
    RoleForm = MyAccessRole.SelectRoleFormByRoleId(RoleIDs)
    If RoleForm.Count = 0 Then 'jika kosong
        For Each aNode As TreeNode In tvUserRights.Nodes
            'Checkbox non aktif
            aNode.Checked = False
        Next
    Else
        PopulateItemCheckedToTheTree(RoleForm)
    End If
End Sub
```

Kedua method diatas saling berkaitan, dimana pada method “PopulateRoleDetails” akan memanggil sebuah method “PopulateItemCheckedToTheTree”. Pada method “PopulateRoleDetails” digunakan untuk mencari data pada table RoleForm (tblRoleForm) dengan menggunakan RoleID sebagai kriteria. Jika hasil pencariannya tidak ditemukan (data kosong), maka seluruh Node pada TreeView (tvUserRights) akan “unchecked/false”, dan jika hasil pencarian nya tidak kosong maka akan memanggil method “PopulateItemCheckedToTheTree”.

Sedangkan pada method “PopulateItemCheckedToTheTree” digunakan untuk mengaktifkan Node pada TreeView sesuai pada hasil pencarian. Sebagai contoh jika RoleID = 1, memiliki hak akses untuk Form 1,2 dan 3 maka pada TreeView akan mengaktifkan checkbox Node (checked) untuk form tersebut aja (Form 1,2 dan 3).

```
Private Sub CheckedTreeNode(ByVal Checked As Boolean)
    Dim IEnum As IEnumerator = _
        tvUserRights.Nodes.GetEnumerator
    While IEnum.MoveNext
        Dim aNode As TreeNode = _
            DirectCast(IEnum.Current, TreeNode)
        aNode.Checked = Checked
    End While
End Sub
```

Method diatas berfungsi untuk mengaktifkan (checked) maupun menonaktifkan (unchecked) checkbox Node pada TreeView sesuai dengan nilai pada parameter “Checked”

```
Private Function TakeRoleID() As String
    Dim strID = ""

    Dim intPosisiDash = 0
    intPosisiDash =
    lstRoles.SelectedItem.ToString().IndexOf(">")
    'mengambil nilai RoleID, pada listbox yg di klik
    strID = lstRoles.SelectedItem.ToString().Substring(0,
    intPosisiDash - 1)
    Return strID

End Function
```

Jika kita kembali pada method “PopulateRolesToListBox”, seperti di jelaskan sebelumnya, method ini menampilkan data (RoleID dan RoleName) pada listbox. Dimana menggunakan karakter “>” sebagai penghubung RoleID dan RoleName. Lalu untuk method “TakeRoleId” diatas, digunakan untuk mengambil nilai RoleID nya saja, sesuai dengan item yang dipilih.

```
Private Sub InsertItemCheckedOnTheTreeToDatabase(ByVal RoleID As String)
    Dim IEnum As IEnumerator = tvUserRights.Nodes.GetEnumerator
    While IEnum.MoveNext
        Dim aNode As TreeNode = DirectCast(IEnum.Current, TreeNode)

        If aNode.Checked Then 'jika check box aktif

            Dim RoleForm As New RoleForm With {.RoleId = RoleID, _
            .FormId = aNode.Tag.ToString}

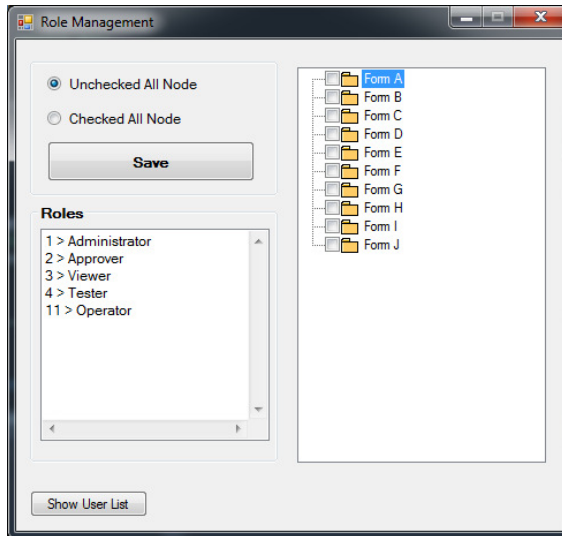
            MyAccessRole.InsertRoleForm(RoleForm) 'Insert data
        End If
    End While
End Sub
```

Method diatas berfungsi untuk melakukan penambahan data pada table RoleForm (tblRoleForm). Dimana data yang dimasukkan hanya pada Node yang checkbox nya aktif (checked).

- Dan pada event frmSiswaUsp\_Load ketikkan sintaks berikut.

```
Try
    AddFormNameToTree()
    PopulateRolesToListBox()
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

Pada saat form ini dibuka, maka dua method diatas (AddFormNameToTree dan PopulateRolesToListBox) ada di eksekusi. Dan akan mendapatkan hasil seperti dibawah.



- Selanjutnya double click pada control ListBox (lstRoles), dan ketikkan sintaks dibawah ini

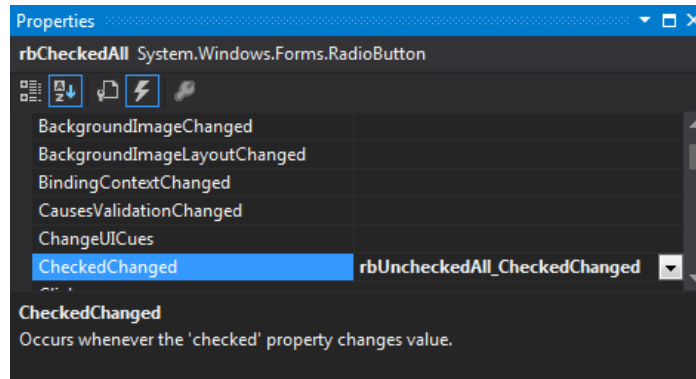
```
Try
    If lstRoles.SelectedIndex <> -1 Then
        PopulateRoleDetails(TakeRoleID)
    End If
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

Sintak diatas berfungsi, jika user memilih item pada ListBox (lstRoles), maka pada TreeView akan menampilkan/mengaktifkan Node yang sesuai dengan item yang dipilih.

- Double click pada RadioButton “rbUncheckedAll” dan ketikkan sintaks dibawah.

```
Try
    Select Case sender.name.ToString.ToUpper
        Case "RBCHECKEDALL"
            CheckedTreeNode(True)
        Case "RBUNCHECKEDALL"
            CheckedTreeNode(False)
    End Select
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

Sedangkan untuk radiobutton “rbCheckedAll” lakukan hal berikut. Pada properties, pilih tab “Events”, selanjutnya pada CheckedChanged ketikkan teks berikut “rbUncheckedAll\_CheckedChanged”



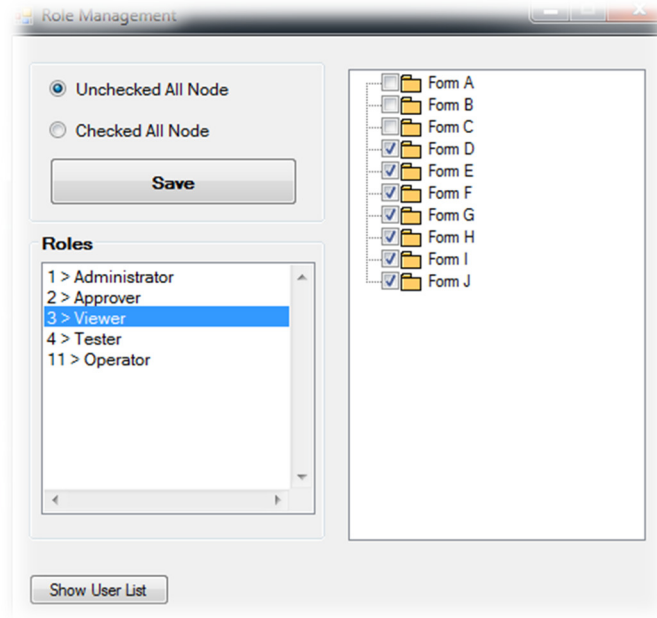
Jika dilihat pada sintaks diatas, jika user memilih radiobutton “Unchecked All Node” maka method “CheckedTreeNode” akan dijalankan dengan mengirimkan parameter yang bernilai “False”. Dan sebaliknya jika user memilih “Checked All Node” maka nilai yang dikirimkan adalah “True”

- Klik ganda button Save dan ketikkan sintaks dibawah.

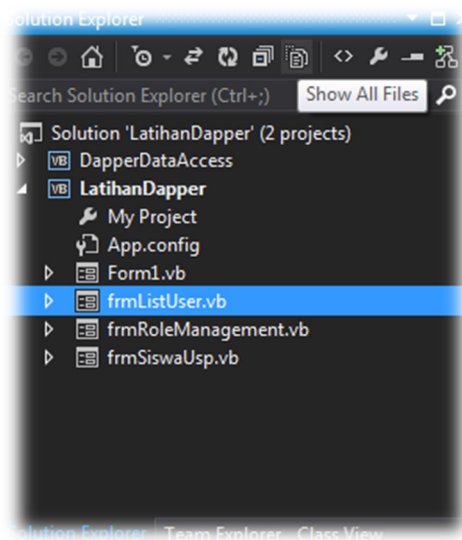
```
Try
    If lstRoles.SelectedIndex = -1 Then
        MsgBox("Pilih dahulu RoleID yang akan diupdate dari List Roles", _
            MsgBoxStyle.Information, "Latihan Dapper")
        Exit Sub
    Else
        Dim mYes_No As String = MsgBox("Apakah kamu ingin menyimpan data ?", _
            vbYesNo, "Latihan Dapper")
        If mYes_No = vbYes Then
            Dim strID = TakeRoleID()
            MyAccessRole.DeleteFormByRoleID(strID)
            InsertItemCheckedOnTheTreeToDatabase(strID)
            CheckedTreeNode(False)
            PopulateRolesToListBox()
            MsgBox("Data disimpan", MsgBoxStyle.Information, "Latihan Dapper")
        End If
    End If
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

Button ini melakukan operasi penyimpanan data pada table RoleForm. Sebelum menyimpan data, pertama kali adalah menghapus data pada table RoleForm berdasarkan RoleID (MyAccessRole.DeleteFormByRoleID(strID)). Selanjutnya menyimpan data sesuai dengan Node yang dipilih.

Jalankan program dan coba satu persatu fungsi yang ada pada form.

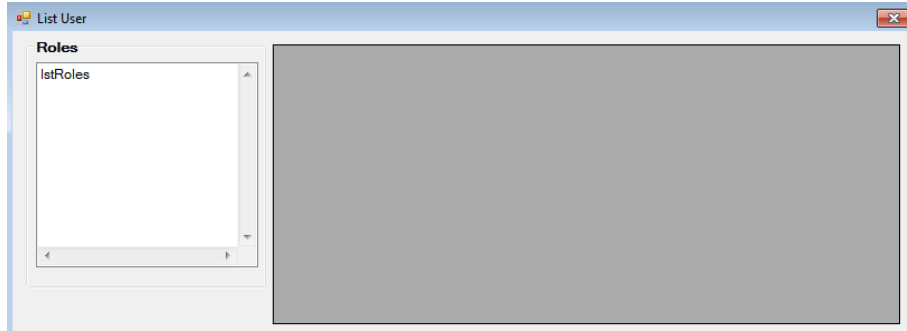


- Tambahkan sebuah form dengan nama “frmListUser”.



- Tambahkan 1 buah ListBox (lstRoles) dan DataGridView (DgvUser).
- Atur posisi 2 kontrol di atas seperti gambar dibawah.





- Import namespace GenericDataAccess
- Selanjutnya ketikkan sintaks seperti dibawah.

```
Dim MyAccessRole As IRoleDAO = New RoleDAO()

Private Sub FormatGridWithBothTableAndColumnStyles()

    DgvUser.DefaultCellStyle.ForeColor = Color.Navy
    DgvUser.RowsDefaultCellStyle.BackColor = Color.AliceBlue
    DgvUser.GridColor = Color.Blue
    DgvUser.BorderStyle = BorderStyle.Fixed3D
    DgvUser.RowHeadersBorderStyle = BorderStyle.Fixed3D
    DgvUser.AutoGenerateColumns = False

    Dim UserColumn As New DataGridViewTextBoxColumn()
    UserColumn.DataPropertyName = "Username"
    UserColumn.HeaderText = "Username"
    UserColumn.Width = 80

    Dim NamaColumn As New DataGridViewTextBoxColumn()
    NamaColumn.DataPropertyName = "Nama"
    NamaColumn.HeaderText = "Nama"
    NamaColumn.Width = 100

    Dim PasswordColumn As New DataGridViewTextBoxColumn()
    PasswordColumn.DataPropertyName = "Password"
    PasswordColumn.HeaderText = "Password"
    PasswordColumn.Width = 100

    DgvUser.Columns.Add(UserColumn)
    DgvUser.Columns.Add(NamaColumn)
    DgvUser.Columns.Add(PasswordColumn)

End Sub

Private Sub PopulateRolesToListBox()
    lstRoles.Items.Clear()
    Dim RoleList As List(Of Roles)

    RoleList = MyAccessRole.SelectRole()
    For Each items In RoleList
        lstRoles.Items.Add(items.RoleId & " > " & items.RoleName)
    Next
End Sub
```

Untuk method “FormatGridWithBothTableAndColumnStyles” digunakan sebagai pengaturan GridView, seperti warna dan kolom-kolom apa saja yang akan di tampilkan pada form.

- Lalu pada event frmListUser\_Load ketikkan sintaks berikut

```
Try
    FormatGridWithBothTableAndColumnStyles()
    PopulateRolesToListBox()
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

- Double click lstRoles dan ketikkan sintak dibawah.

```
Try
    Dim intPosisiDash As Integer
    intPosisiDash = lstRoles.SelectedItem.ToString.IndexOf(">") + 2

    Dim strName As String = _
        lstRoles.SelectedItem.ToString.Trim().Substring(intPosisiDash, _
        lstRoles.SelectedItem.ToString.Length - intPosisiDash)
    If lstRoles.SelectedIndex <> -1 Then
        DgvUser.DataSource = MyAccessRole.SelectUserByRoleName(strName)
    End If
Catch ex As Exception
    MsgBox(ex.Message)
End Try
```

Pada saat user klik item pada List Box maka pada GridView akan ditampilkan data user yang mempunyai Role sesuai dengan yang dipilih. Untuk menampilkan data pada gridview kita menggunakan RoleName sebagai kriteria. Yang menjadi pertanyaan, pada tblUser tidak terdapat kolom “RoleName” untuk pencarian. Untuk lebih memhami, buka kembali class “RoleDAO” dan lihat pada method “SelectUserByRoleName”. Untuk Query kita menggunakan InnerJoin antara table User (tblUser) dan table Role (tblRole). Sedangkan untuk menampung data kita menggunakan List (Of User), seperti kita ketahui class User memiliki asosiasi dengan class Role.

Sedangkan pada Dapper kita melakukan spliton antara primary key username dan RoleID. Seperti sintaks dibawah.

```
Return mycon.Query(Of Users, Roles, Users)(strQuery, Function(users,
roles)

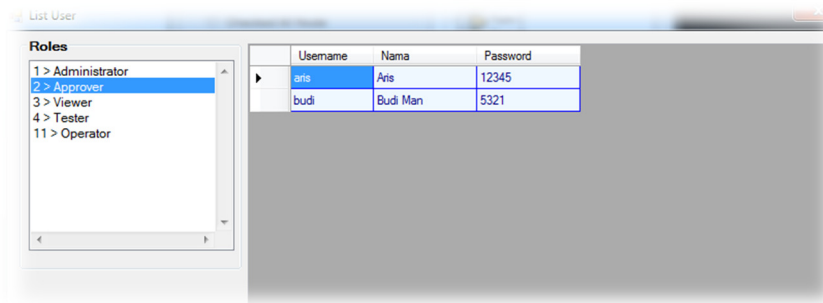
users.Role = roles

Return users
End
Function,
New With
{RoleName},
splitOn:="username,RoleID")
```

- Pada “frmRoleManagement” double click btnShow dan ketikkan sintaks berikut

```
Dim frm As New frmListUser
frm.ShowDialog()
```

- Jalankan program, lalu pada Form Role Management , klik button Show User List. Selanjutnya akan tampil form User List, pilih role pada listbox, maka data-data user yang berdasarkan Role yang dipilih akan tampil pada gridview. Seperti gambar dibawah.



Untuk Source Code Project dapat di download disini

<http://junindar.blogspot.com/2015/01/dapper-micro-orm-dan-stored-procedure.html>

Semoga artikel ini dapat menambah wawasan kita semua khusus nya penulis sendiri.

Wassalam.



## 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). Junindar mendapatkan Award Microsoft MVP VB pertanggal 1 oktober 2009 hingga saat ini. Senang mengutak-atik computer yang berkaitan dengan bahasa pemrograman. Keahlian, sedikit mengerti beberapa bahasa pemrograman seperti : VB.Net, C#, SharePoint, ASP.NET, VBA. Reporting: Crystal Report dan Report Builder. Database: MS Access, MY SQL dan SQL Server. 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 Consulting dan Project Management di Malaysia sebagai Senior Consultant. Memiliki beberapa sertifikasi dari Microsoft yaitu Microsoft Certified Professional Developer (MCPD – SharePoint 2010), MOS (Microsoft Office Specialist) dan MCT (Microsoft Certified Trainer) 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 : [junindar@gmail.com](mailto:junindar@gmail.com)*

## Referensi

1. [www.msdn.microsoft.com](http://www.msdn.microsoft.com)
2. [www.planetsourcecode.com](http://www.planetsourcecode.com)
3. [www.codeproject.com](http://www.codeproject.com)
4. [www.aspnet.com](http://www.aspnet.com)

Masih banyak lagi referensi yang ada di Intenet. Anda tinggal cari di [www.Google.com](http://www.Google.com).  
Dengan kata kunci “**tutorial VB.Net**”