

BAB XI

Praktikum Membangun Web Application (9): Kereta Belanja

I. Tujuan Praktikum

Dapat mengetahui cara membuat program secara OOP.

II. Bahan & Alat

Hardware berupa PC/Laptop dengan perangkat lunak SQL Server 2008 pada Microsoft Visual Studio .NET 2008 serta modul praktikum.

III. Metode yang digunakan

Melanjutkan Web Application Penjualan Online dengan membuat Halaman Kereta Belanja dan membuat Class Item Kereta dan Kereta.

IV. Landasan Teori

a. Dasar OOP

Pemrograman berorientasi obyek (OOP) merupakan metode pemrograman dimana pengembang harus mendefinisikan tipe data dari struktur data dan juga tipe dari operasi yang dapat diaplikasikan ke struktur data. Dengan demikian struktur data menjadi obyek yang memiliki data dan fungsi.

Kemampuan utama dari pemrograman berorientasi obyek (OOP), antara lain:

1. Pemrograman OOP menekankan pada data daripada prosedur karena data diperlakukan sebagai elemen yang penting dan tidak boleh mengalir secara bebas dalam program.
2. Data disembunyikan dari akses program oleh fungsi-fungsi eksternal
3. Obyek dapat berkomunikasi satu dengan yang lain melalui function.
4. Data baru dan function dapat dengan mudah ditambahkan pada saat dibutuhkan
5. Konsep pemrogramannya mengikuti pendekatan button up.

Beberapa keunggulan menggunakan pemrograman OOP dibandingkan dengan pemrograman terstruktur, antara lain:

1. Maintainability

Maintainability (kemampuan untuk mudah dikelola) karena Visual Basic.NET menyediakan struktur modular yang jelas untuk program. Maksudnya jika terjadi kesalahan pada program yang besar maka solusinya adalah memecah program tersebut menjadi modul-modul kecil (modularitas). Modularitas melekat pada OOP

karena class-class yang ada merupakan suatu modul sehingga class-class yang memiliki fungsionalitas sama akan disimpan pada namespace yang sama pula.

2. Extensibility

Extensibility (kemampuan untuk dapat diperluas) yaitu class-class yang ada dapat diperluas dengan menambahkan method atau data, atau mengubah perilaku dari method.

3. Reusability

Reusability (kemampuan untuk dapat digunakan kembali) yaitu kode-kode program yang sudah pernah ditulis (class) dapat digunakan kembali bagi mereka yang memerlukan fungsionalitas yang sama yang disediakan oleh kode.

b. Komponen OOP

1. Obyek

Elemen dasar dari konsep pemrograman OOP adalah obyek. Obyek merupakan dasar dari entitas run-time dalam suatu sistem berorientasi obyek. Ketika suatu program dijalankan, obyek berinteraksi satu dan lainnya dengan saling mengirimkan pesan (message).

2. Class

Class adalah suatu koleksi dari obyek-obyek yang bertipe sama. Class merupakan cetakan biru (blue print) atau template atau spesifikasi dari obyek. Dengan kata lain, class adalah representasi abstrak dari obyek, sedangkan obyek adalah representasi nyata dari class atau perwujudan (instance) dari suatu class.

Suatu class dapat berisi property, field, method dan event dari suatu obyek. Gabungan dari property, field, method dan event umumnya disebut dengan members dari class. Suatu class dapat dideklarasikan dengan menggunakan kata kunci "Class".

3. Method

Method adalah suatu tindakan yang dilakukan oleh obyek. Sedangkan event adalah pemberitahuan yang diterima oleh obyek atau dikirimkan ke obyek atau aplikasi lain.

4. Field

Field adalah informasi atau atribut yang terdapat dalam suatu obyek. Bentuk field mirip dengan variabel yaitu dapat dibaca dan diset secara langsung.

5. Property

Property adalah informasi (atribut) yang dapat disimpan dalam suatu obyek. Property dapat dideklarasikan dengan menggunakan variabel public atau method property.

Public property digunakan untuk mengidentifikasi dirinya sebagai field. Pendeklarasian seperti ini tidak memiliki kemampuan untuk melakukan validasi pembacaan dan penulisan nilai property.

Method property memungkinkan pengguna untuk mengontrol operasi baca dan tulis dari property dalam suatu class.

Dalam deklarasi property digunakan kata kunci Property yang diikuti oleh nama property-nya. Terdapat dua bagian dalam property yaitu Get dan Set.

Get digunakan untuk mendefinisikan semua property yang membolehkan akses ke nilai data, sedangkan Set digunakan untuk mendefinisikan semua property yang membolehkan mengubah nilai data.

V. Bahan Praktikum

- a. Buka kembali Katalog_Online
- b. Tambahkan class baru pada folder App_Code yaitu class ItemKereta
- c. Buat program-program berikut pada class ItemKereta

Deklarasikan Field yang akan digunakan

```
Private FKdBrng As String
Private FNmBrng As String
Private FHrgBrng As Long
Private FQty As Integer
```

Deklarasikan Property yang akan digunakan

```
Public Property PKdBrG() As String
    Get
        Return FKdBrG
    End Get
    Set(ByVal value As String)
        FKdBrG = value
    End Set
End Property

Public Property PNmBrG() As String
    Get
        Return FNmBrG
    End Get
    Set(ByVal value As String)
        FNmBrG = value
    End Set
End Property

Public Property PHrgBrG() As Long
    Get
        Return FHrgBrG
    End Get
    Set(ByVal value As Long)
        FHrgBrG = value
    End Set
End Property

Public Property PQty() As Integer
    Get
        Return FQty
    End Get
    Set(ByVal value As Integer)
        FQty = value
    End Set
End Property

Public ReadOnly Property PSubtotal() As Double
    Get
        Return FHrgBrG * FQty
    End Get
End Property
```

Dilanjutkan dengan membuat Konstruktor untuk data baru.

```
Public Sub New(ByVal PKdBrG As String, ByVal PNmBrG As String, ByVal PHrgBrG As Long, ByVal PQty As Integer)
    Me.PKdBrG = PKdBrG
    Me.PNmBrG = PNmBrG
    Me.PHrgBrG = PHrgBrG
    Me.PQty = PQty
End Sub
```

- d. Tambahkan class baru pada folder App_Code yaitu class Kereta
- e. Buat program-program berikut pada class Kereta

Deklarasikan Imports yang dibutuhkan

```
Imports Microsoft.VisualBasic
Imports System.Collections.Generic
Public Class Kereta
```

Buat method-method yang dibutuhkan

```
Private PkeretaBelanja As List(Of ItemKereta)

Public Sub New()
    PkeretaBelanja = New List(Of ItemKereta) ()
End Sub

Public Function GetItems() As List(Of ItemKereta)
    Return PkeretaBelanja
End Function
```

```
Public Sub AddItems(ByVal KdBrng As String, ByVal NmBrng As String, ByVal HrgBrng As Long, ByVal Qty As Integer)
    Dim barang_ada As Boolean = False
    For Each item As ItemKereta In PkeretaBelanja
        If item.PKdBrng = KdBrng Then
            item.PQty += 1
            barang_ada = True
        End If
    Next

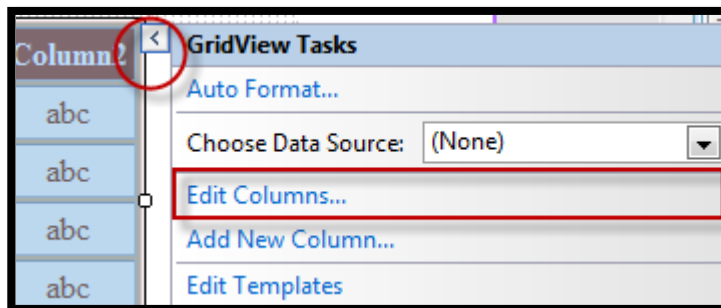
    If Not barang_ada Then
        Dim item As New ItemKereta(KdBrng, NmBrng, HrgBrng, 1)
        PkeretaBelanja.Add(item)
    End If
End Sub
```

```
Public Sub UpdateQty(ByVal index As Integer, ByVal Qty As Integer)
    Dim item As ItemKereta
    item = PkeretaBelanja(index)
    item.PQty = Qty
End Sub

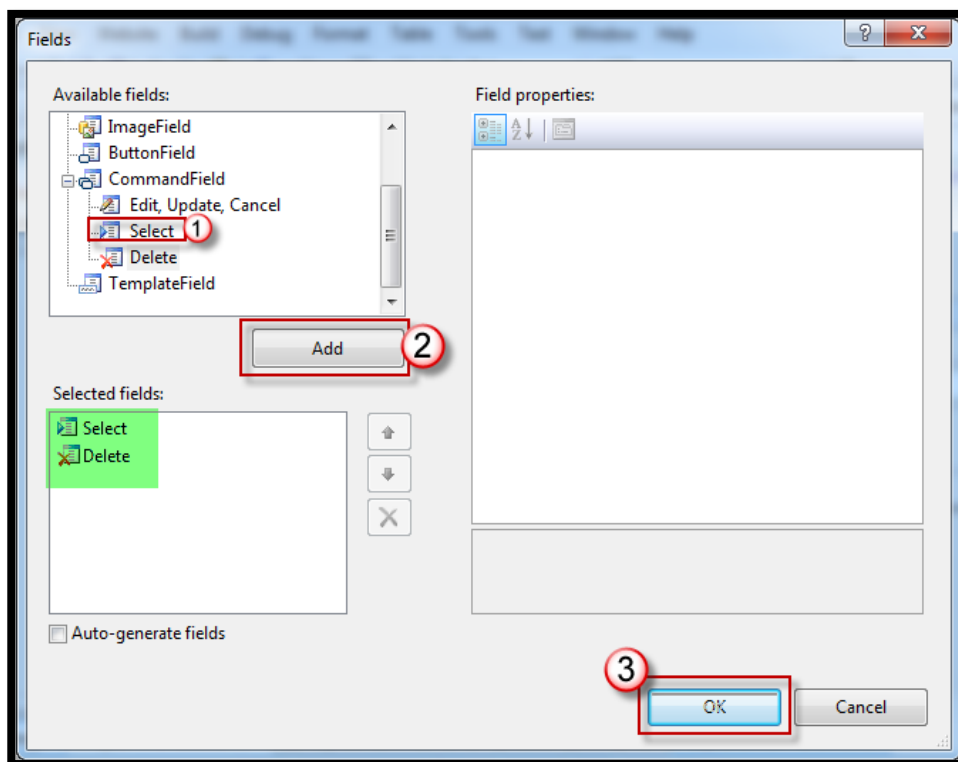
Public Sub DeleteItem(ByVal index As Integer)
    PkeretaBelanja.RemoveAt(index)
End Sub
```

```
Public ReadOnly Property Count() As Integer
    Get
        Return PkeretaBelanja.Count
    End Get
End Property
```

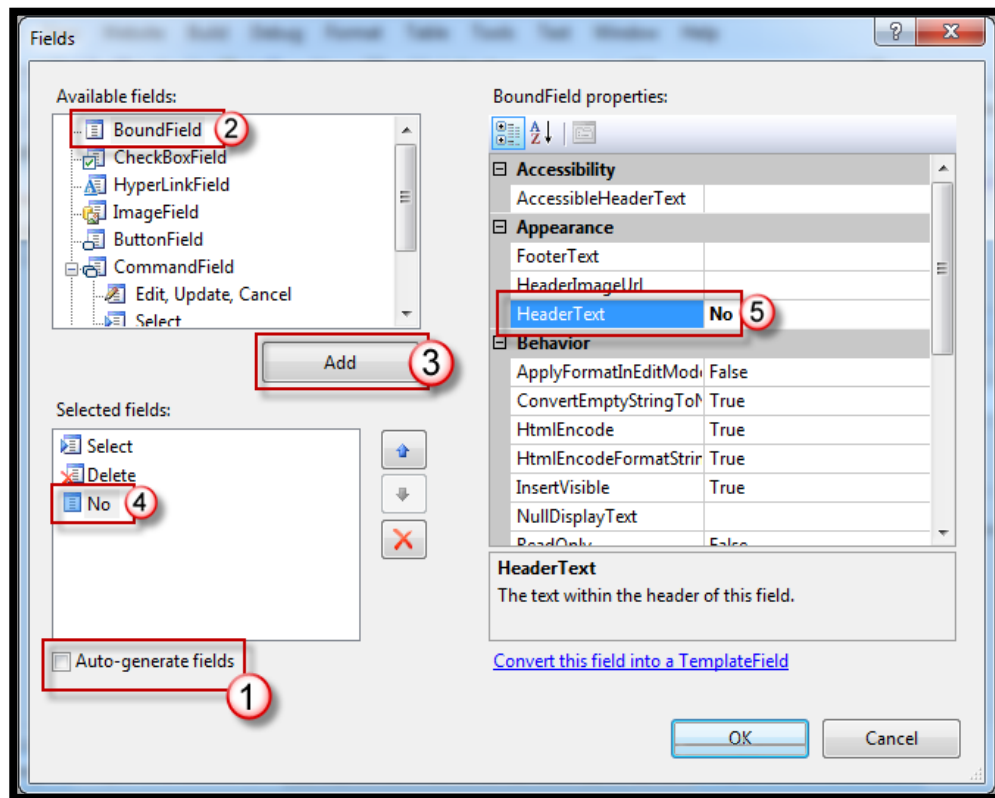
- f. Buka halaman ListKeretaBelanja.aspx kemudian Desain GridView sebagai berikut:
Pilih anak panah pada pojok kanan atas dari GridView kemudian pilih Edit Columns...



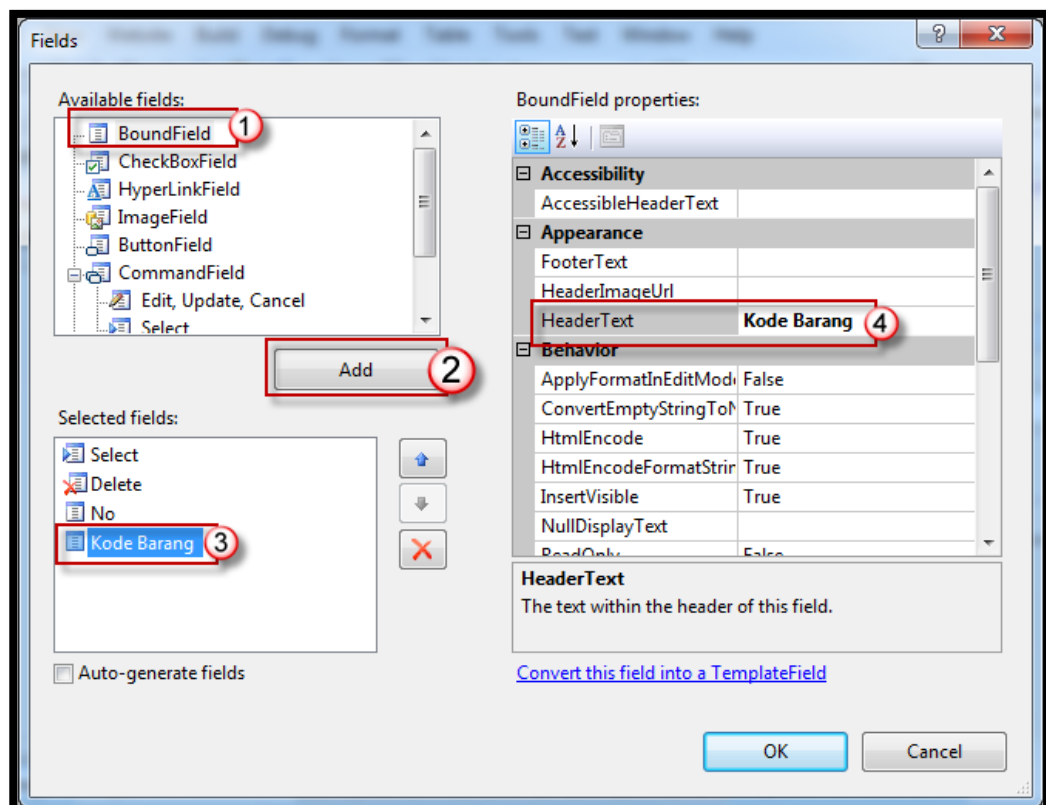
Pilih Command Field Select kemudian klik tombol Add, lanjutkan juga untuk Delete. Jika sudah selesai klik tombol OK.



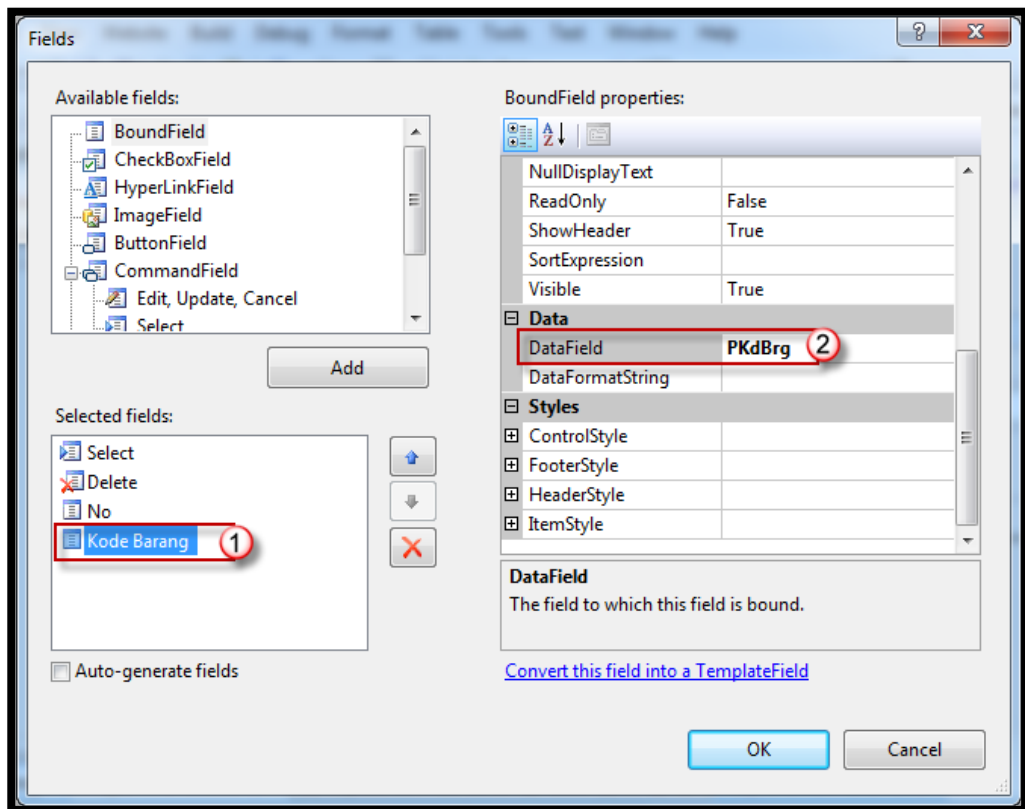
Hilangkan tanda Cek List pada Auto-generate fields. Kemudian pilih BoundField kemudian klik tombol Add. Kemudian ubah Properties HeaderText dengan No.



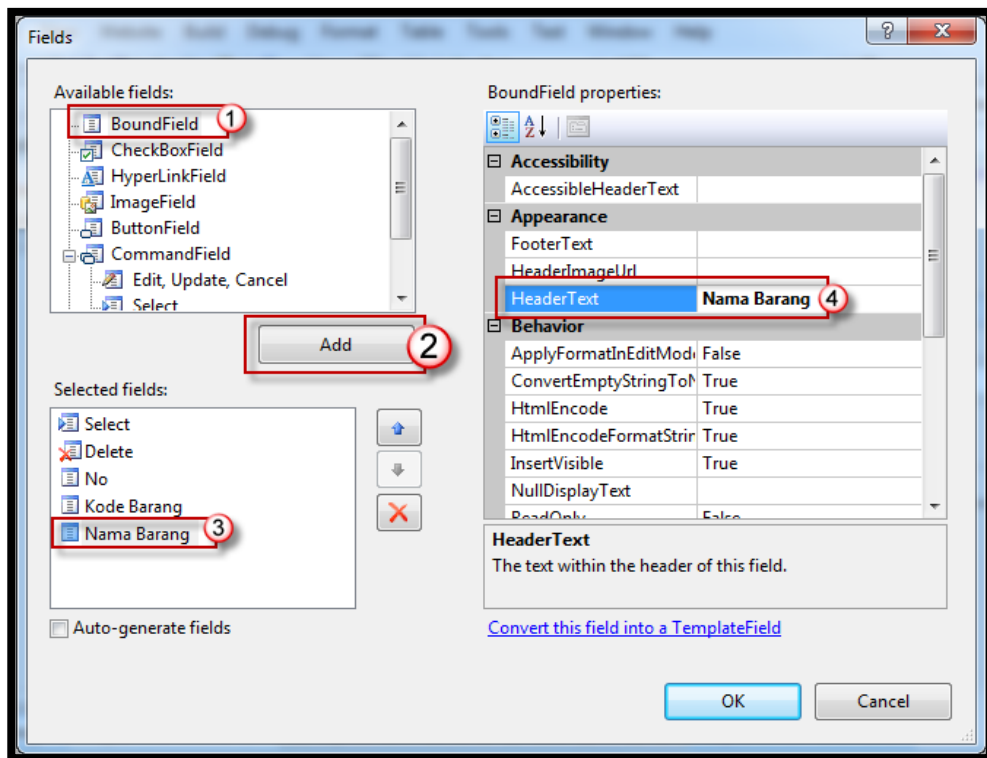
Pilih kembali fields BoundField kemudian klik tombol OK ubah properties HeaderText menjadi Kode Barang.



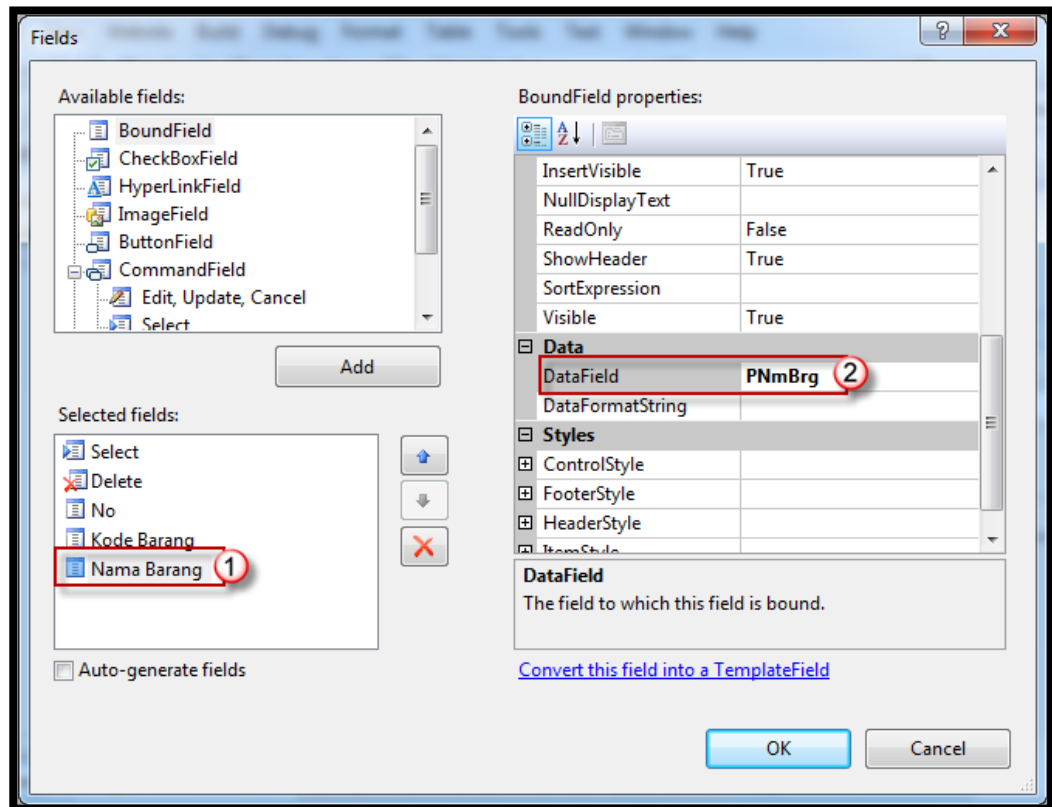
Ubah properties DataField juga menjadi PKdBrg



Pilih BoundField kemudian klik tombol Add, ubah properties HeaderText nya menjadi Nama Barang



Ubah properties DataField juga menjadi PNmBrG.



Ulangi untuk kolom Harga Barang, Jumlah dan Sub Total dengan ketentuan sebagai berikut:

Field	HeaderText	DataField
BoundField	Harga Barang	PHrgBrG
BoundField	Jumlah	PQty
BoundField	Sub Total	PSubtotal

Hasil akhir dari GridView setelah pembuatan kolom selesai semua.

		No	Kode Barang	Nama Barang	Harga Barang	Jumlah	Sub Total
Select	Delete		Databound	Databound	Databound	Databound	Databound
Select	Delete		Databound	Databound	Databound	Databound	Databound
Select	Delete		Databound	Databound	Databound	Databound	Databound
Select	Delete		Databound	Databound	Databound	Databound	Databound
Select	Delete		Databound	Databound	Databound	Databound	Databound

VI. Daftar Pustaka

1. Darie, Cristian, Wyatt Barnett, Build Your Own ASP.NET 3.5 Web Site Using C# & VB, 3rd Edition, SitePoint Pty. Ltd, 2008

2. MacDonald, Matthew, Mario Szpuszta, Vidya Vrat Argawal, Pro ASP.NET 3.5 in VB 2008: Includes Silverlight 2, Apress, 2009
3. Cox, Ken, ASP.NET 3.5 For Dummies, Wiley Publishing, Inc, 2008
4. Yuswanto & Subari, Pemrograman Database Visual Basic.NET, 2007