

Bab 11

DataGrid

11.1 Kelas DataGrid

Kelas `DataGrid` atau `GridView` merupakan salah satu penampil data yang sangat fleksibel, sebuah data yang terikat pada daftar kontrol, menampilkan item dari sumber data dalam tabel. Kontrol `DataGrid` memungkinkan untuk memilih, memilah, dan mengedit item ini. Posisi dari kelas `DataGrid` termasuk anggota dari `BaseDataList` yang merupakan turunan dari kelas `WebControl`, seperti yang terlihat dibawah ini:

```
System.Object
  System.Web.UI.Control
    System.Web.UI.WebControls.WebControl
      System.Web.UI.WebControls.BaseDataList
        System.Web.UI.WebControls.DataGrid
```

Secara umum kelas `DataGrid` menerima warisan dari `BaseDataList`, sehingga kode secara umum ditunjukkan di bawah ini:

```
public class DataGrid: BaseDataList {
    public virtual object DataSource { get; set; }
    public virtual bool AutoGenerateColumns { get; set; }
    public virtual DataGridColumnCollection Columns {get;}
    public virtual DataGridItemsCollection Items { get; set; }
    public virtual DataGridItem SelectedItem { get; set; }
    public virtual int SelectedIndex { get; set; }
    ...}
```

Nilai `DataGridColumn` merupakan sebuah kelas dengan beberapa properti yang berisi setting dari tipe kolom. Kelas tsb secara umum ditunjukkan dibawah ini:

```
public class DataGridColumn: ... {
    public virtual string HeaderText { get; set; }
    public virtual string FooterText { get; set; }
    public virtual TableItemStyle HeaderStyle {get;}
    public virtual TableItemStyle FooterStyle {get;}
    ...}
```

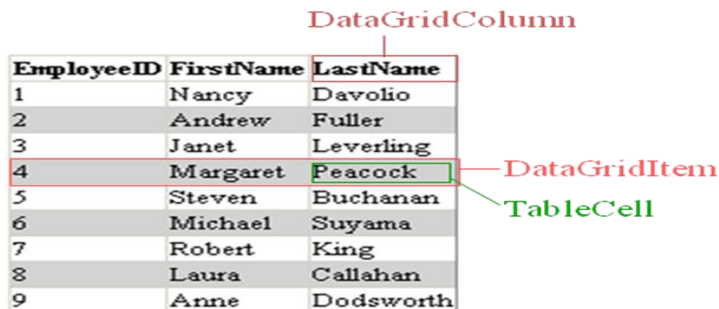
Nilai `DataGridItem` merupakan sebuah kelas yang mengacu pada data koleksi. Kelas tsb secara umum ditunjukkan dibawah ini:

```
public class DataGridItem: ... {
    public virtual TableCellCollection Cells { get; }
    ...}
```

Nilai `TableCell` diatas merupakan turunan dari `webcontrol`. Kelas tsb secara umum ditunjukkan dibawah ini:

```
public class TableCell: WebControl {
    public virtual string Text { get; set; }
    public virtual bool Wrap { get; set; }
    ...}
```

Visualisasi sebuah datagrid, terdiri dari `DataGridColumn`, `DataGridItem` dan `TableCell`, digambarkan pada gambar 11.1 dibawah ini:



Gambar 11.1 Visualisasi DataGrid

11.1.1 Konfigurasi kelas DataGrid

Sebuah DataGrid dapat diatur konfigurasi, berupa formatting, misalnya format dari cellpadding, cellspace, GridLines dsb. Kesemuanya ini, terdapat pada kelas datagrid, secara umum dapat dilihat dibawah ini

```
public class DataGrid: BaseDataList {
public virtual GridLines GridLines { get; set; }
    public virtual int CellPadding { get; set; }
    public virtual int CellSpacing { get; set; }
    public virtual bool ShowHeader { get; set; }
    public virtual bool ShowFooter { get; set; }
    public virtual TableItemStyle AlternatingItemStyle { get; }
    public virtual TableItemStyle HeaderStyle { get; }
    public virtual TableItemStyle FooterStyle { get; }
    public virtual TableItemStyle ItemStyle { get; }
    public virtual TableItemStyle SelectedItemStyle { get; }}
```

Pilihan Gridlines merupakan sebuah enumerasi terhadap nilai Both, Horizontal, None, Vertical. Nilai enumerasi ini dapat dituliskan sebagai sebuah properti kelas:

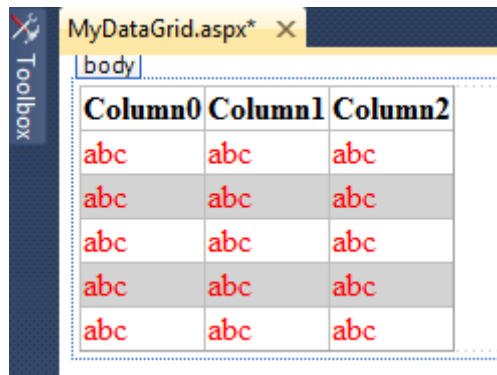
```
public enum GridLines {
    Both, Horizontal, None, Vertical}
```

Pilihan untuk menampilkan data dengan tipe tertentu yang terdapat pada datagrid; seperti jenis huruf, warna huruf, warna latarbelakang dsb, merupakan sebuah kelas TableItemStyle diambil dari turunan style seperti yang dituliskan dibawah ini:

```
public class TableItemStyle: Style {
    public FontInfo Font { get; }
    public Color ForeColor { get; set; }
    public Color BackColor { get; set; }
    public Unit Width { get; set; }
    public Unit Height { get; set; }}
```

Contoh pemakain kelas DataGrid pengaturan pengaturan tertentu, dengan langkah-langkah sebagai berikut:

1. Buka VS2010 dan buat situs baru (atau buka situs yang telah ada pada bab sebelumnya)
2. Tambahkan file program dengan menekan CTL-N atau pilihan menu File-New-File dan beri nama file kode program tsb dengan nama MyDataGrid.aspx
3. Setelah file dibuat, pada mode desain, rancang program sesuai gambar 11.1, beserta settingnya



Gambar 11.1 Pembuatan File MyDataGrid.aspx

4. Setting sesuai dengan skrip dibawah ini:

```
<asp:DataGrid ID="DataGrid1"  
    HeaderStyle-Font-Bold="true" Runat="server">  
    <ItemStyle ForeColor="Red" Font-Name="Times" />  
    <AlternatingItemStyle BackColor="LightGray" />  
</asp:DataGrid>
```

Tampilan kontrol Datagrid dapat disesuaikan dengan mengatur properti gaya untuk bagian yang berbeda dari kontrol. Tabel berikut berisi sifat gaya yang berbeda. Hal ini diperlihatkan pada table 11.1

Tabel 11.1 Setting Tampilan DataGrid

Properti	Deskripsi
AlternatingItemStyle	Menentukan gaya untuk bolak item dalam kontrol Datagrid.
EditItemStyle	Menentukan gaya untuk item yang sedang diedit dalam kontrol Datagrid.
FooterStyle FooterStyle	Menentukan gaya untuk bagian footer di kontrol Datagrid.
HeaderStyle HeaderStyle	Menentukan gaya untuk bagian header di kontrol Datagrid.
ItemStyle ItemStyle	Menentukan gaya untuk item dalam kontrol Datagrid.
PagerStyle PagerStyle	Menentukan gaya untuk bagian pemilihan halaman kontrol Datagrid.
SelectedItemStyle	Menentukan gaya untuk item yang dipilih dalam kontrol Datagrid.

Tampilan juga dapat diatur sebagai bagian yang tampil atau tersembunyi pada bagian yang berbeda dari kontrol. Tabel berikut berisi daftar properti yang mengontrol bagian mana yang akan ditampilkan atau disembunyikan. Pengaturan ini diperlihatkan pada table 11.2

Tabel 11.2 Setting Tampilan DataGrid

Properti	Deskripsi
ShowFooter	Menunjukkan atau menyembunyikan bagian footer dari kontrol Datagrid.
ShowHeader	Menunjukkan atau menyembunyikan bagian header kontrol Datagrid.

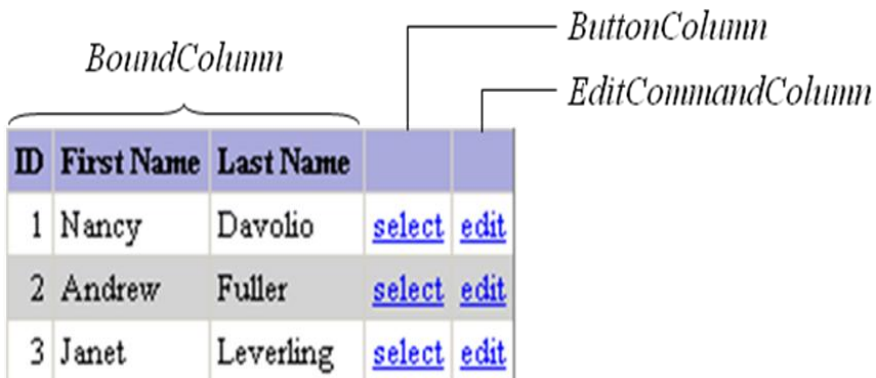
Tampilan pada kontrol Datagrid dapat pula diatur lewat pemrograman dengan menambahkan atribut ke <td> dan tag <tr> seperti pada pembuatan tabel.

11.1.2 Method dan events pada Kelas DataGrid

Sebuah DataGrid sebagaimana kelas-kelas lain dalam ASP.NET, memiliki *method* dan *events*. Daftar *method* dan *events* pada kelas DataGrid secara umum diperlihatkan dibawah ini:

```
public class DataGrid: BaseDataList {
    ...
    //--- methods
    public override void DataBind();
    ...
    //--- events
    public event DataGridCommandEventHandler ItemCommand;
    public event DataGridCommandEventHandler EditCommand;
    public event DataGridCommandEventHandler CancelCommand;
    public event DataGridCommandEventHandler UpdateCommand;
    public event DataGridCommandEventHandler DeleteCommand;
    public event EventHandler SelectedIndexChanged;
}
```

Sebuah *events* yang dibangkitkan tergantung pada jenis kolom pada DataGrid, contoh sebuah *events* digambarkan pada gb 11.2 dibawah ini:



Gambar 11.2 Jenis events pada kolom DataGrid

Jenis-jenis kolom yang disediakan pada kelas DataGrid, dimana jenis kolom yang berbeda menentukan perilaku kolom dalam kontrol. Tabel berikut berisi daftar jenis kolom yang berbeda yang dapat digunakan.

Tabel 11.2 Jenis Kolom pada Kelas DataGrid

Jenis Kolom	Deskripsi
BoundColumn	Menampilkan kolom terikat ke lapangan dalam sumber data. Hal ini menampilkan setiap item dalam bidang sebagai teks. Pada kolom jenis ini merupakan kolom default pada jenis kontrol Datagrid.
ButtonColumn	Tombol perintah ditampilkan pada setiap item dalam kolom. Hal ini memungkinkan Anda untuk membuat sebuah kolom tombol kontrol kustom, seperti tombol Tambah atau Hapus.
EditCommandColumn	Menampilkan kolom yang berisi perintah editing untuk setiap item dalam kolom.
HyperLinkColumn	Menampilkan isi dari setiap item pada kolom sebagai hyperlink. Isi kolom dapat terikat ke lapangan dalam sumber data atau teks statis.
TemplateColumn	Menampilkan setiap item dalam kolom berikut template tertentu. Hal ini memungkinkan untuk memberikan kontrol kustom dalam kolom.

Contoh pemakaian pada tabel 11.1 ditunjukkan dibawah ini:

1. `<asp:BoundColumn ...`, Secara otomatis terikat ke kolom sumber data: `DataField = "dbColumnName"`
2. `<asp:ButtonColumn ...`, Setiap baris berisi sebuah tombol yang memunculkan `ItemCommand`:

```
ButtonType = "LinkButton" | "PushButton"
Text = "buttonLabel"
CommandName = "Select" | "Delete" | "anyText"
```



3. <asp:EditCommandColumn ..., Setiap baris berisi tombol edit. Jika diklik itu diganti dengan pembaruan dan membatalkan tombol.

```
ButtonType = "LinkButton" | "PushButton"
EditText = "editButtonLabel"
UpdateText = "updateButtonLabel"
CancelText = "cancelButtonLabel"
```

Contoh visualisasi seperti dibawah ini:

1	Nancy	Davolio	edit
---	-------	---------	----------------------

Saat tombol edit di klik, baris tsb berubah seperti dibawah ini

<input type="text" value="1"/>	<input type="text" value="Nancy"/>	<input type="text" value="Davolio"/>	update cancel
--------------------------------	------------------------------------	--------------------------------------	---

Secara default, properti `AutoGenerateColumns` diatur ke `True`, yang menciptakan obyek `BoundColumn` untuk setiap *field* dalam sumber data. Setiap *field* ini kemudian diberikan sebagai kolom dalam kontrol `DataGrid` dalam urutan yang muncul di setiap *field* sumber data.

Kontrol properti dapat pula diatur secara manual dengan mengontrol kolom yang muncul dalam kontrol `DataGrid`. Kontrol properti `AutoGenerateColumns` ke `false` dan kemudian daftar kolom yang ingin disertakan ditulis diantara tag `<Columns>`. Kolom yang ditentukan ditambahkan ke Kolom koleksi dalam urutan yang tercantum.. Hal ini memungkinkan untuk pemrograman mengontrol kolom dalam kontrol `DataGrid`.

Contoh pemakaian `DataGrid` secara sederhana ditampilkan dengan langkah-langkah dibawah ini:

1. Buka VS2010 dan buat situs baru (atau buka situs yang telah ada pada bab sebelumnya)

2. Tambahkan file program dengan menekan CTL-N atau pilihan menu File-New-File dan beri nama file kode program tsb dengan nama MyDataGrid.aspx,
3. Setelah file dibuat, buka file tsb, dan program dengan kode program dibawah ini:

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="MyDataGrid.aspx.cs" Inherits="MyDataGrid" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="Form1" OnInit="Page_Load" Runat="server">
        <asp:DataGrid ID="grid" Runat="server" />
    </form>
</body>
</html>
```

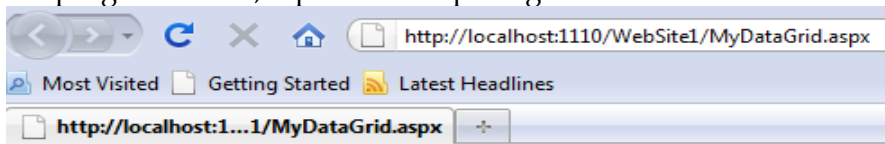
4. Penanganan aksi dilakukan pada kode sumber (*code behind*) MyDataGrid.aspx.cs yang berisi inialisasi untuk koneksi ke database dan melakukan query data. Kode program MyDataGrid.aspx.cs merupakan kode program di sisi server yang mengatur penampilan data pada datagrid, pertama kali meminta koneksi ke database, kemudian mengatur sumber dari datagrid menuju pada database yang telah terkoneksi.

```
using System;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;

public partial class MyDataGrid : System.Web.UI.Page
{
```

```
protected void Page_Load(object sender, EventArgs e)
{
    DataSet ds = new DataSet();
    SqlConnection con = new SqlConnection(
        "data source=DWI-PC\\SQLEXPRESS;initial
        catalog=samples;user id=sa;
        password=;Trusted_Connection=true");
    string sqlString = "SELECT EmployeeID, FirstName,
        LastName FROM Employees";
    SqlDataAdapter adapter =
        new SqlDataAdapter(sqlString, con);
    adapter.Fill(ds, "Employees");
    if (ds.HasErrors) ds.RejectChanges();
    else ds.AcceptChanges();
    grid.DataSource =
        ds.Tables["Employees"].DefaultView;
    grid.DataBind();
    grid.HeaderStyle.Font.Bold = true;
    grid.AlternatingItemStyle.BackColor =
        System.Drawing.Color.LightGray;
}
}
```

5. Jalankan program dengan menekan F5, hasil dari contoh program diatas, diperlihatkan pada gambar 11.3



EmployeeID	FirstName	LastName
1	Nancy	Davolio
2	Andrew	Fuller
3	Janet	Leverling
4	Margaret	Peacock
5	Steven	Buchanan
6	Michael	Suyama
7	Robert	King
8	Laura	Callahan
9	Anne	Dodsworth

Gambar 11.3 Hasil Visualisasi datagrid sederhana

Pemakaian dataGrid dapat pula ditambahkan berbagai macam pengaturan. Pemakaian datagrid dengan pengaturan-pengaturan tampilan diperlihatkan. Langkah-langkah untuk membuat contoh tsb adalah:

1. Buka VS2010 dan buat situs baru (atau buka situs yang telah ada pada bab sebelumnya)
2. Tambahkan file program dengan menekan CTL-N atau pilihan menu File-New-File dan beri nama file kode program tsb dengan nama MyDataGrid2.aspx,
3. Setelah file dibuat, buka file tsb, dan program dengan kode program dibawah ini:

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="MyDataGrid2.aspx.cs" Inherits="MyDataGrid2" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body><div>
<form id="Form1" OnLoad="Page_Load" Runat="server">
<asp:DataGrid ID="grid" Runat="server"
AutoGenerateColumns="false" CellPadding="3"
HeaderStyle-BackColor="#aaaadd" AlternatingItemStyle-
BackColor="LightGray" OnDeleteCommand="DeleteRow"
OnSelectedIndexChanged="SelectRow" >
<Columns>
  <asp:BoundColumn HeaderText="ID" DataField="EmployeeID">
    <ItemStyle HorizontalAlign="Right" />
  </asp:BoundColumn>
  <asp:BoundColumn HeaderText="First Name"
    DataField="FirstName" />
  <asp:BoundColumn HeaderText="Last Name"
    DataField="LastName" />
  <asp:ButtonColumn ButtonType="LinkButton" Text="delete"
    CommandName="Delete" />
  <asp:ButtonColumn ButtonType="LinkButton" Text="select"
```

```
                CommandName="Select" />
            </Columns>
        </asp:DataGrid><br>
        <asp:Label ID="label" Runat="server" />
    </div>
</form>
</body>
</html>
```

4. Penanganan aksi dilakukan pada kode sumber (*code behind*) MyDataGrid2.aspx.cs yang berisi inisialisasi untuk koneksi ke database dan melakukan query data, penanganan pemilihan baris serta penghapusan record. Kode program MyDataGrid2.aspx.cs merupakan kode program di sisi server yang mengatur penampilan data pada datagrid, pertama kali meminta koneksi ke database, kemudian mengatur sumber dari datagrid menuju pada database yang telah terkoneksi dan menyajikan pilihan bagi pengguna untuk memilih baris dan menghapus record.

```
using System;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;

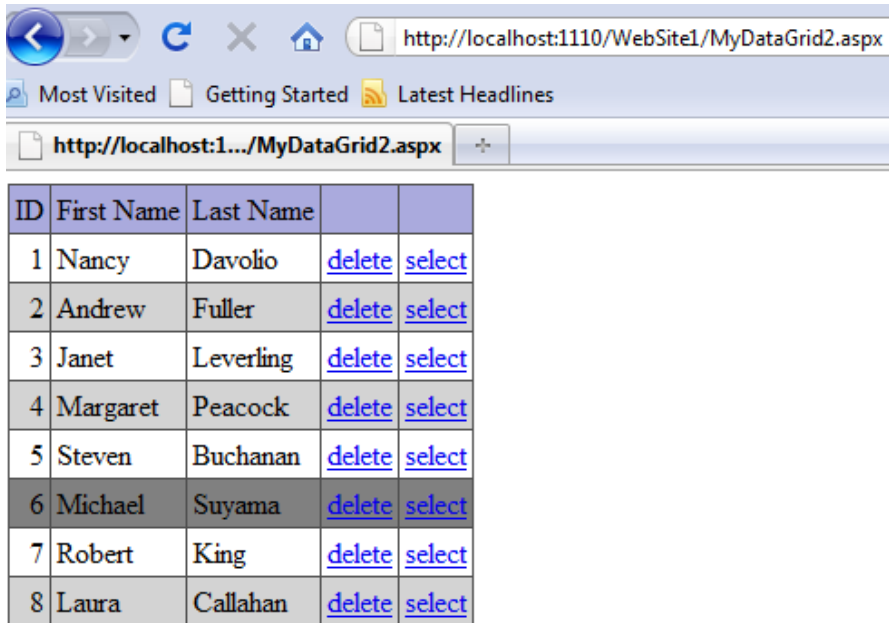
public partial class MyDataGrid2 : System.Web.UI.Page
{
    DataView dataView;
    protected void Page_Load(object sender, EventArgs e)
    {
        DataSet ds;
        if (!IsPostBack)
        {
            ds = new DataSet();
            SqlConnection con = new SqlConnection("
                data source=DWI-PC\\SQLEXPRESS;initial
                catalog=samples;user id=sa;
                password=;Trusted_Connection=true");
            string sqlString = "SELECT EmployeeID, FirstName,
```

```
                LastName FROM Employees";
        SqlDataAdapter adapter =
            new SqlDataAdapter(sqlString, con);
        adapter.Fill(ds, "Employees");
        if (ds.HasErrors) ds.RejectChanges();
        else ds.AcceptChanges();
        Session["Data"] = ds;
    }
    else ds = (DataSet)Session["Data"];
    dataView = ds.Tables["Employees"].DefaultView;
    grid.DataSource = dataView;
    grid.DataBind();
}

public void DeleteRow (object sender,
                        DataGridCommandEventArgs e)
{
    dataView.Delete(e.Item.DataSetIndex);
    grid.DataSource = dataView;
    grid.DataBind();
}

public void SelectRow (object sender, EventArgs e)
{
    grid.SelectedItemStyle.BackColor =
        System.Drawing.Color.Gray;
    label.Text = grid.SelectedItem.Cells[1].Text + " " +
        grid.SelectedItem.Cells[2].Text;
}
}
```

5. Jalankan program dengan menekan F5, hasil dari contoh program diatas, diperlihatkan pada gambar 11.4



Michael Suyama

Gambar 11.4 Hasil Visualisasi datagrid sederhana

Penanganan *events* pada datagrid untuk pemilihan baris dan penghapusan baris sesuai kode pada pengaturan tampilan datagrid seperti kode dibawah ini

```
OnDeleteCommand="DeleteRow"  
OnSelectedIndexChanged="SelectRow" >
```