

7

Coding Form Master Barang

Materi Pembahasan :

1. Koneksi Form Master Barang
2. Membuat Prosedur Bersih, Atur Tombol dan Refresh Barang
3. Coding Simpan, Edit, Hapus dan Cari

7.1 Koneksi Form Master Barang

1. Buka Form Master Barang, lalu pindah ke bagian kode program / *view code* (F7)
2. Ketik kode **using System.Data.SqlClient;** di bagian header, yang merupakan kode untuk menambahkan library koneksi ke SQL Server

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
//1.tambahkan library sqlserver
using System.Data.SqlClient;
```

3. Ketikkan kode berikut di bawah **public partial class FBarang : Form {**

```
public partial class FBarang : Form
{
    //2.import kelas koneksi.cs yang berasal dari folder Kelas menjadi objek --> konn
    Kelas.Koneksi konn = new Kelas.Koneksi();

    //3.tambahkan beberapa variabel yang digunakan untuk proses CRUD
    private SqlCommand cmd;
    private DataSet ds;
    private SqlDataAdapter da;
```

7.2 Membuat Prosedur Bersih, Atur Tombol dan Refresh Barang

Prosedur merupakan bagian kecil dari kode program yang terpisah dalam blok sendiri yang berisi perintah-perintah yang dapat digunakan/dipanggil berulang kali. Pada form master barang akan dibuat 3 prosedur yaitu prosedur bersih, atur tombol dan refresh barang.

1. Pada baris berikutnya, ketik kode berikut :

```
//4.membuat prosedur bersih
5.references
void BERSIH()
{
    TxtKodeBarang.Clear();
    TxtNamaBarang.Clear();
    TxtHargaBarang.Clear();
    TxtStokBarang.Clear();
    CmbSatuan.Text = "-Pilih Satuan-";
    TxtCariBarang.Clear();
}
```

Prosedur Bersih() berisi perintah untuk membersihkan komponen inputan (textbox dan combobox) pada form barang.

2. Pada baris berikutnya, ketik kode berikut :

```
//5.membuat prosedur atur_tombol  
5 references  
void ATUR_TOMBOL(Boolean status)  
{  
    BtnEdit.Enabled = status;  
    BtnHapus.Enabled = status;  
}
```

Prosedur Atur_Tombol() berisi perintah untuk mengatur ketersediaan (aktif / non aktif) button edit dan button hapus pada form barang.

3. Pada baris berikutnya, Ketik kode berikut :

```
//6.membuat prosedur refresh_barang  
4 references  
void refresh_barang()  
{  
    SqlConnection conn = konn.GetConn();  
    try  
    {  
        conn.Open();  
        cmd = new SqlCommand("SELECT * FROM barang", conn);  
        ds = new DataSet();  
        da = new SqlDataAdapter(cmd);  
        da.Fill(ds, "barang");  
        ListBarang.DataSource = ds;  
        ListBarang.DataMember = "barang";  
        ListBarang.Refresh();  
    }  
    catch (Exception e)  
    {  
        MessageBox.Show(e.ToString());  
    }  
    finally  
    {  
        conn.Close();  
    }  
}
```

Prosedur Refresh_Barang() berisi perintah untuk memperbaharui data pada listbarang. Data yang diperbaharui merupakan data yang ada pada tabel barang.

7.3 Coding Simpan, Edit, Hapus dan Cari

1. Kembali ke desain Form Barang, klik 2 kali **button Baru**
Ketik kode berikut :

```
1 reference  
private void BtnBaru_Click(object sender, EventArgs e)  
{  
    //7.memanggil prosedur bersih dan atur_tombol saat klik tombol baru  
    BERSIH();  
    TxtKodeBarang.Focus();  
    ATUR_TOMBOL(false); //membuat nilai status pada prosedur menjadi false  
    BtnSimpan.Enabled = true; //mengaktifkan button Simpan  
}
```

2. Kembali ke desain Form Barang, klik 2 kali **button Simpan**
 Ketik kode berikut :

```
1 reference
private void BtnSimpan_Click(object sender, EventArgs e)
{
    //8.perintah simpan data barang
    //jika ada komponen inputan yang kosong
    if (TxtKodeBarang.Text.Trim() == "" || TxtNamaBarang.Text.Trim() == ""
        || TxtHargaBarang.Text.Trim() == "" || TxtStokBarang.Text.Trim() == "" ||
        CmbSatuan.Text.Trim() == "")
    {
        //tampilkan pesan bahwa data belum lengkap
        MessageBox.Show("Data belum lengkap, lengkapi data", "WARNING", MessageBoxButtons.OK,
            MessageBoxIcon.Exclamation);
    }
}
```

```
else
{
    //mempersiapkan sqlconnection
    SqlConnection conn = konn.GetConn();
    try
    {
        //mempersiapkan sqlcommand yang akan dieksekusi
        cmd = new SqlCommand("EXECUTE simpan_barang '" + TxtKodeBarang.Text + "', '"
            + TxtNamaBarang.Text + "', '" + TxtHargaBarang.Text + "', '"
            + TxtStokBarang.Text + "', '" + CmbSatuan.Text + "'", conn);
        //membuka koneksi database
        conn.Open();
        //menjalankan perintah query
        cmd.ExecuteNonQuery();
        //menampilkan pesan berhasil disimpan
        MessageBox.Show("Data berhasil disimpan", "INFORMASI", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        //memanggil prosedur BERSIH dan refresh_barang
        BERSIH();
        refresh_barang();
    }
    catch (Exception x)
    {
        MessageBox.Show(x.ToString());
    }
}
}
```

EXECUTE simpan_barang merupakan perintah untuk mengeksekusi **Stored Procedure simpan_barang** yang ada pada database

3. Kembali ke desain form Barang, klik komponen **Datagridview ListBarang**, lalu klik 2x event **CellClick**, ketik kode berikut :

```
1 reference
private void ListBarang_CellClick(object sender, DataGridViewCellEventArgs e)
{
    //9.mengambil nilai dari record yang dipilih menuju ke textbox
    try
    {
        BtnSimpan.Enabled = false;
        ATUR_TOMBOL(true);
        DataGridViewRow row = this.ListBarang.Rows[e.RowIndex];
        TxtKodeBarang.Text = row.Cells["KodeBarang"].Value.ToString();
        TxtNamaBarang.Text = row.Cells["NamaBarang"].Value.ToString();
        TxtHargaBarang.Text = row.Cells["HargaBarang"].Value.ToString();
        TxtStokBarang.Text = row.Cells["StokBarang"].Value.ToString();
        CmbSatuan.Text = row.Cells["Satuan"].Value.ToString();
    }
    catch (Exception x)
    {
        MessageBox.Show(x.ToString());
    }
}
```

4. Kembali ke desain Form Barang, klik 2 kali **button Edit**
 Ketik kode berikut :

```
1 reference
private void BtnEdit_Click(object sender, EventArgs e)
{
    //10.perintah edit data barang
    if (TxtKodeBarang.Text.Trim() == "" || TxtNamaBarang.Text.Trim() == ""
        || TxtHargaBarang.Text.Trim() == "" || TxtStokBarang.Text.Trim() == "" ||
        CmbSatuan.Text.Trim() == "")
    {
        MessageBox.Show("Data belum lengkap, lengkapi data", "WARNING", MessageBoxButtons.OK,
            MessageBoxIcon.Exclamation);
    }
}
```

```
else
{
    SqlConnection conn = konn.GetConn();
    try
    {
        cmd = new SqlCommand("EXECUTE ubah_barang '" + TxtNamaBarang.Text + "', '"
            + TxtHargaBarang.Text + "', '" + TxtStokBarang.Text + "', '"
            + CmbSatuan.Text + "', '" + TxtKodeBarang.Text + "'", conn);
        conn.Open();
        cmd.ExecuteNonQuery();
        MessageBox.Show("Data berhasil dirubah", "INFORMASI", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        BERSIH();
        refresh_barang();
        //memanggil prosedur atur_tombol
        ATUR_TOMBOL(false);
        BtnSimpan.Enabled = true;
    }
}
```

```
        catch (Exception x)
        {
            MessageBox.Show(x.ToString());
        }
    }
}
```

5. Kembali ke desain form Barang, klik 2 kali **button Hapus**
Ketik kode berikut :

```
1 reference
private void BtnHapus_Click(object sender, EventArgs e)
{
    //11.perintah hapus data barang
    if (MessageBox.Show("Yakin ingin menghapus data barang ?",
        "Konfirmasi", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.Yes)
    {
        SqlConnection conn = konn.GetConn();
        cmd = new SqlCommand("EXECUTE hapus_barang '" + TxtKodeBarang.Text + "'", conn);
        conn.Open();
        cmd.ExecuteNonQuery();
        MessageBox.Show("Data barang berhasil dihapus", "informasi", MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        BERSIH();
        refresh_barang();
        //memanggil prosedur atur_tombol
        ATUR_TOMBOL(false);
        BtnSimpan.Enabled = true;
    }
}
```

6. Kembali ke desain form Barang, klik 2 kali **button Keluar**
Ketik kode berikut :

```
1 reference
private void BtnKeluar_Click(object sender, EventArgs e)
{
    //12.keluar form barang
    Close();
}
```

7. Kembali ke desain form Barang, klik 2 kali **txtcaribarang**, ketik kode berikut :

```
1 reference
private void TxtCariBarang_TextChanged(object sender, EventArgs e)
{
    //13.perintah cari nama barang
    SqlConnection conn = konn.GetConn();
    conn.Open();
    cmd = new SqlCommand("EXECUTE cari_barang '" + TxtCariBarang.Text + "'", conn);
    ds = new DataSet();
    da = new SqlDataAdapter(cmd);
    da.Fill(ds, "barang");
    ListBarang.DataSource = ds;
    ListBarang.DataMember = "barang";
    ListBarang.Refresh();
    conn.Close();
}
```

8. Masih di mode view code, cari kode berikut

```
1 reference
public FBarang()
{
    InitializeComponent();
}
```

9. Ketik kode berikut di bawah **InitializeComponent();**

```
1 reference
public FBarang()
{
    InitializeComponent();
    //14.panggil prosedur pada saat form barang pertama kali dijalankan
    BERSIH();
    ATUR_TOMBOL(false);
    refresh_barang();
}
```

10. **Running program**, gambar 7.1 merupakan tampilan form master barang yang sudah terkoneksi dengan (database) tabel barang

KodeBarang	NamaBarang	HargaBarang	S
B0001	Xiaomi Redmi 5A	1210000	1
B0002	Xiaomi Redmi 4X	1650000	9
B0003	HP Pavilion 14	3925000	2
B0004	Acer Aspire One	3420000	9
B0005	Canon EOS 700D	6200000	5
B0006	ADATA Premiere	651000	7
B0007	RJ45	50000	9

Gambar 7.1. Form Master Barang yang Terkoneksi Database