

## 04

## Form Transaksi

Materi Pembahasan

1. Desain Tabel Transaksi.
2. Desain Form.
3. Form Dialog Barang.
4. Form Transaksi.

**1. Desain Tabel Transaksi**

Tabel transaksi berisi beberapa foreign key dari beberapa tabel lain yang saling berelasi. Berikut merupakan desain tabel transaksi dan tabel lain yang saling berelasi.

**2. Desain Form**

Setidaknya dibutuhkan dua buah form yaitu form **transaksi** dan **dialog barang**. Form transaksi digunakan untuk menginputkan data – data transaksi sedangkan form dialog barang digunakan untuk daftar barang yang akan dimasukan dalam tabel **transaksi\_detail**.

## 2.1 Form Transaksi

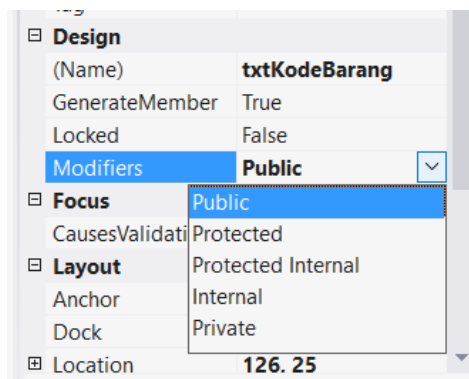
- a. Buat form **Transaksi** dengan desain form sebagai berikut :

- b. Rincian komponen pada form Transaksi adalah sebagai berikut :

No	Toolbox	Name	Text
1	Label	lblTanggal	Tanggal :
2	TextBox	txtPencarian	
3	DataGridView	gridTransaksi	
4	Label	lblIdTransaksi	ID Transaksi
5	Label	lblNamaPembeli	Nama Pembeli :
6	Label	lblTotal	Total :
7	Label	lblKodeBarang	Kode Barang :
8	Label	lblNamaBarang	Nama Barang :
9	Label	lblHargaBarang	Harga :
10	Label	lblJumlahBarang	Jumlah :
11	TextBox	txtIdTransaksi	
12	TextBox	txtNamaPembeli	
13	TextBox	txtTotal	
14	TextBox	txtKodeBarang	
15	TextBox	txtNamaBarang	

16	TextBox	txtHargaBarang	
17	TextBox	txtJumlahBarang	
18	Button	btnCari	Cari
19	Button	btnTambahkan	Tambahkan
20	Button	btnHapus	Hapus
21	Button	btnSimpanTransaksi	Simpan Transaksi
22	Button	btnHapusTransaksi	Hapus Transaksi
23	DataGridView	gridDetail	

- c. Ke bagian **properties** dan ubah akses modifier komponen no **14, 15, dan 16** menjadi **public**.



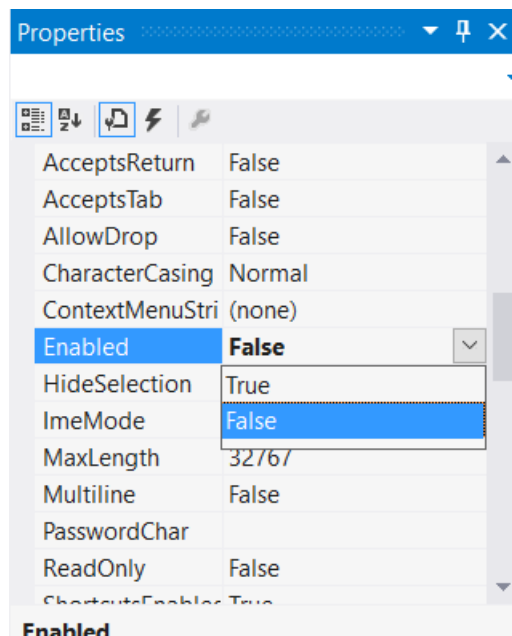
## 2.2 Form DialogBarang

- a. Buat form DialogBarang dengan desain sebagai berikut :

b. Rincian komponen pada form **DialogBarang** adalah sebagai berikut :

No	Toolbox	Name	Text
1	Label	lblId	ID :
2	Label	lblNama	Nama :
3	Label	lblHarga	Harga :
4	TextBox	txtId	
5	TextBox	txtNama	
6	TextBox	txtHarga	
7	Button	btnPilih	Pilih
8	TextBox	txtPencarian	
9	DataGridView	gridBarang	

c. Set **Enabled** pada komponen **4, 5, dan 6** menjadi **False**.



### 3. Form DialogBarang

Sama seperti **MasterBarang**, DialogBarang akan menampilkan daftar barang yang ada dalam database. Perbedaannya adalah pada form **MasterBarang** terjadi operasi CRUD sedangkan pada **DialogBarang** hanya menampilkan daftar barang kemudian dimasukan dalam form **Transaksi**.

- a. Sesuaikan program anda dengan Gambar berikut (perhatikan kotak merah) :

```

AplikasiPenjualanBarang.Gui.DialogBarang
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10 using MySql.Data.MySqlClient;
11
12 namespace AplikasiPenjualanBarang.Gui
13 {
14     public partial class DialogBarang : Form
15     {
16         public Transaksi transaksi = null;
17         Kelas.Koneksi koneksi = new Kelas.Koneksi();
18
19         public DialogBarang()
20         {
21             InitializeComponent();
22             // TEMPAT KODE 2
23         }
24
25         // TEMPAT KODE 1
26
27

```

- b. Tambahkan script berikut pada lokasi yang telah ditentukan :

```

public DataSet getData()
{
    DataSet dts = new DataSet();
    try
    {
        string sql = "SELECT * FROM barang";
        MySqlCommand command = new MySqlCommand();
        command.Connection = koneksi.getConn();
        command.CommandType = CommandType.Text;
        command.CommandText = sql;
        MySqlDataAdapter data = new
        MySqlDataAdapter(command);
        data.Fill(dts, "barang");
    }
    catch (Exception e)
    {
        MessageBox.Show("get\n" + e.ToString());
    }
}

```

1

```

    }
    return dts;
}

void header()
{
    gridBarang.Columns[0].HeaderText = "Kode Barang";
    gridBarang.Columns[1].HeaderText = "Nama Barang";
    gridBarang.Columns[2].HeaderText = "Harga Barang";
}

public void loadDaftar()
{
    DataSet data = getData();
    gridBarang.DataSource = data;
    gridBarang.DataMember = "barang";
    header();
}

```

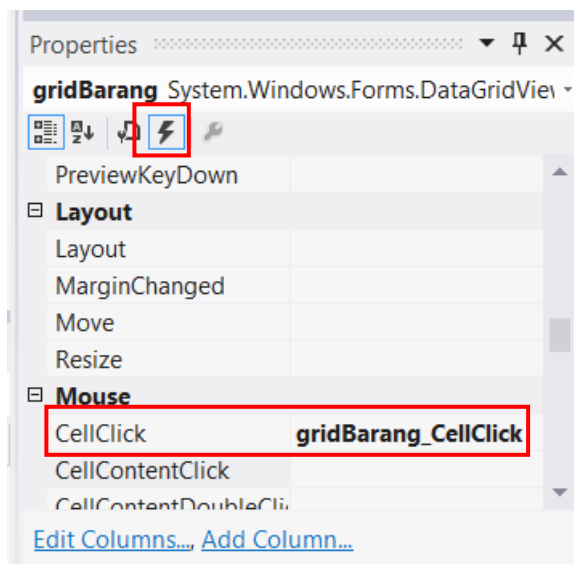
```

loadDaftar();
txtId.Enabled = false;
txtNama.Enabled = false;
txtHarga.Enabled = false;

```

2

- c. Klik **DataGridView** pada form (gridBarang). Pada bagian properties pilih **Events** dan double klik pada **CellClick**.



d. Masukkan script berikut:

```
txtId.Text =
gridBarang.Rows[e.RowIndex].Cells[0].Value.ToString();
txtNama.Text =
gridBarang.Rows[e.RowIndex].Cells[1].Value.ToString();
txtHarga.Text =
gridBarang.Rows[e.RowIndex].Cells[2].Value.ToString();
```

e. Double klik pada button Pilih (btnPilih) kemudian masukan script berikut:

```
transaksi.txtKodeBarang.Text = txtId.Text;
transaksi.txtNamaBarang.Text = txtNama.Text;
transaksi.txtHargaBarang.Text = txtHarga.Text;
this.Dispose();
```

#### 4. Form Transaksi

Form Transaksi akan merekam seluruh data transaksi. Sebagian data seperti nama pembeli, tanggal transaksi akan disimpan pada tabel **transaksi**, sedangkan daftar barang yang dibeli akan disimpan dalam tabel **transaksi\_detail**. Berikut adalah langkah untuk membuat Form Transaksi.

a. Sesuaikan program anda dengan gambar berikut:

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10 using MySql.Data.MySqlClient;
11
12 namespace AplikasiPenjualanBarang.Gui
13 {
14     public partial class Transaksi : Form
15     {
16         Kelas.Koneksi koneksi = new Kelas.Koneksi();
17
18         public Transaksi()
19         {
20             InitializeComponent();
21             lblTanggal.Text = "Tanggal : " + DateTime.UtcNow.Date.ToString("dd-M-yyyy");
22             // TEMPAT KODE 2
23         }
24
25         // TEMPAT KODE 1
26     }
```

- b. Masukan script berikut sesuai tempat yang telah ditentukan.

1

```

public DataSet getData()
{
    DataSet dts = new DataSet();
    try
    {
        string sql = "SELECT * FROM transaksi";
        MySqlCommand command = new MySqlCommand();
        command.Connection = koneksi.getConn();
        command.CommandType = CommandType.Text;
        command.CommandText = sql;
        MySqlDataAdapter data = new
MySqlDataAdapter(command);
        data.Fill(dts, "transaksi");
    }
    catch (Exception e)
    {
        MessageBox.Show("get\n" + e.ToString());
    }
    return dts;
}

void headerTransaksi()
{
    gridTransaksi.Columns[0].HeaderText = "ID";
    gridTransaksi.Columns[1].HeaderText = "Nama Pembeli";
    gridTransaksi.Columns[2].HeaderText = "Total";
    gridTransaksi.Columns[2].HeaderText = "Tanggal";
}

public void loadDaftarTransaksi()
{
    DataSet data = getData();
    gridTransaksi.DataSource = data;
    gridTransaksi.DataMember = "transaksi";
    headerTransaksi();
}

void headerDetail()
{
    gridDetail.Columns.Add("Kode", "Kode");
    gridDetail.Columns.Add("Nama", "Nama");
    gridDetail.Columns.Add("Harga", "Harga");
    gridDetail.Columns.Add("Jumlah", "Jumlah");
    gridDetail.Columns.Add("Total", "Total");

    gridDetail.Columns[0].HeaderText = "Kode";
    gridDetail.Columns[1].HeaderText = "Nama";
    gridDetail.Columns[2].HeaderText = "Harga";
    gridDetail.Columns[3].HeaderText = "Jumlah";
    gridDetail.Columns[4].HeaderText = "Total";
}

```



```

void hitungTotal()
{
    int totalBelanja = 0;
    int index = 0;
    foreach (DataGridViewRow row in gridDetail.Rows)
    {
        if
        (!String.IsNullOrEmpty(gridDetail.Rows[index].Cells[0].Value
        as String))
        {
            totalBelanja +=
            Int32.Parse(row.Cells[4].Value.ToString());
        }
        index++;
    }
    txtTotal.Text = totalBelanja.ToString();
}

void clearTransaksi()
{
    txtIdTransaksi.Clear();
    txtKodeBarang.Clear();
    txtNamaPembeli.Clear();
    txtTotal.Clear();
}

void clearDetail()
{
    txtKodeBarang.Clear();
    txtNamaBarang.Clear();
    txtHargaBarang.Clear();
    txtJumlahBarang.Text = "1";
}

void clearGridDetail()
{
    try
    {
        foreach (DataGridViewRow row in gridDetail.Rows)
        {
            gridDetail.Rows.RemoveAt(0);
        }
    }
    catch
    {
    }
}

```

2

```
headerDetail();
loadDaftarTransaksi();
```

- c. Double klik pada button Cari (btnCari) dan masukan script berikut:

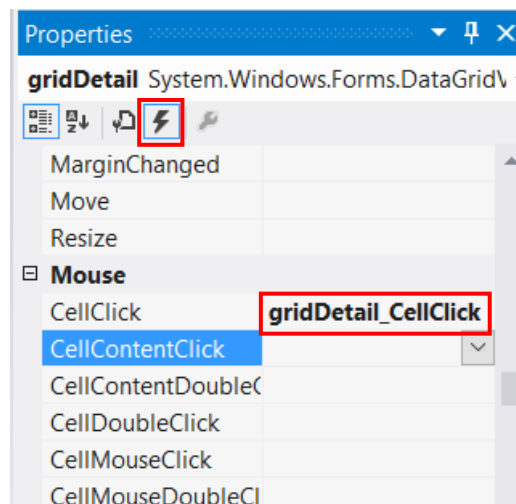
```
DialogBarang x = new DialogBarang();
x.transaksi = this;
x.Show();
```

- d. Double klik pada button Tambahkan (btnTambahkan) dan masukan script berikut:

```
string kodeBarang = txtKodeBarang.Text;
string namaBarang = txtNamaBarang.Text;
int hargaBarang = Int32.Parse(txtHargaBarang.Text);
int jumlahBarang = Int32.Parse(txtJumlahBarang.Text);
int total = hargaBarang * jumlahBarang;

gridDetail.Rows.Add(kodeBarang, namaBarang, hargaBarang,
jumlahBarang, total);
clearDetail();
hitungTotal();
```

- e. Klik **DataGridView** pada form (**gridDetail**). Pada bagian properties pilih **Events** dan double klik pada **CellClick**.



- f. Masukan script berikut:

```
try
{
    txtKodeBarang.Text =
gridDetail.Rows[e.RowIndex].Cells[0].Value.ToString();
```

```

        txtNamaBarang.Text =
gridDetail.Rows[e.RowIndex].Cells[1].Value.ToString();
        txtHargaBarang.Text =
gridDetail.Rows[e.RowIndex].Cells[2].Value.ToString();
        txtJumlahBarang.Text =
gridDetail.Rows[e.RowIndex].Cells[3].Value.ToString();
    }
    catch
    {
        MessageBox.Show("Anda memilih cell yang kosong.");
    }

```

g. Double klik pada button hapus berikut:

The screenshot shows a transaction form. It has input fields for 'ID Transaksi', 'Nama Pembeli', and 'Total'. Below these is a section titled 'Barang' containing fields for 'Kode Barang', 'Nama Barang', 'Harga', and 'Jumlah' (which has the value '1'). At the bottom of the form are three buttons: 'Cari', 'Tambahkan', and 'Hapus'. The 'Hapus' button is highlighted with a red rectangular border.

h. Masukan script berikut:

```

try
{
    int index = 0;
    foreach (DataGridViewRow row in gridDetail.Rows)
    {
        string kodeBarang = row.Cells[0].Value.ToString();
        if (kodeBarang == txtKodeBarang.Text)
        {
            gridDetail.Rows.RemoveAt(index);
            break;
        }
        index++;
    }
    hitungTotal();
}
catch
{
    MessageBox.Show("Terjadi Kesalahan");
}

```

- i. Double klik pada button **Simpan Transaksi (btnSimpanTransaksi)** dan masukan script berikut:

```
try
{
    string sql = "insert into transaksi(nama_pembeli, total)
values(@nama_pembeli, @total)";
    MySqlCommand command = new MySqlCommand();
    command.Connection = koneksi.getConn();
    command.Connection.Open();
    command.CommandType = CommandType.Text;
    command.CommandText = sql;
    command.Parameters.AddWithValue("@nama_pembeli",
txtNamaPembeli.Text);
    command.Parameters.AddWithValue("@total",
Int32.Parse(txtTotal.Text));
    command.ExecuteNonQuery();
    command.Connection.Close();

    command.Connection.Open();
    command.CommandText = "SELECT * FROM transaksi ORDER BY
trx_id DESC LIMIT 1";
    MySqlDataReader reader = command.ExecuteReader();
    int trx_id = 0;
    while (reader.Read())
    {
        trx_id = Convert.ToInt16(reader.GetString(0));
    }
    command.Connection.Close();

    if (trx_id != 0)
    {
        command.Connection.Open();
        int index = 0;
        foreach (DataGridViewRow row in gridDetail.Rows)
        {
            if
(!String.IsNullOrEmpty(gridDetail.Rows[index].Cells[0].Value
as String))
            {
                command.Parameters.Clear();
                command.CommandText = "INSERT INTO
transaksi_detail(trx_id, barang_id, jumlah) VALUES(@trx_id,
@barang_id, @jumlah)";
                command.Parameters.AddWithValue("@trx_id",
trx_id);
                command.Parameters.AddWithValue("@barang_id",
row.Cells[0].Value.ToString());
                command.Parameters.AddWithValue("@jumlah",
row.Cells[3].Value.ToString());
                command.ExecuteNonQuery();
            }
            index++;
        }
        command.Connection.Close();
    }
}
```

```

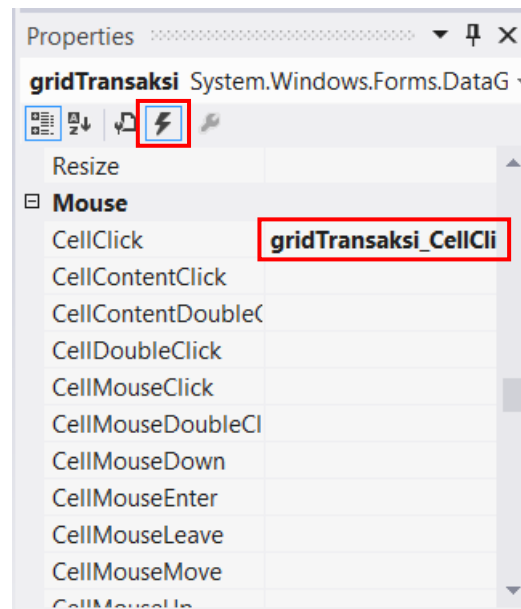
    }

    loadDaftarTransaksi();
    clearTransaksi();
    clearDetail();
    clearGridDetail();

    MessageBox.Show("Data berhasil disimpan", "info",
    MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (Exception ex)
{
    MessageBox.Show("Data gagal disimpan\n" + ex, "info",
    MessageBoxButtons.OK, MessageBoxIcon.Information);
}

```

- j. Klik **DataGridView** pada form (**gridTransaksi**). Pada bagian properties pilih **Events** dan double klik pada **CellClick**.



- k. Masukkan script berikut:

```

try
{
    clearGridDetail();
    int trx_id =
    Convert.ToInt16(gridTransaksi.Rows[e.RowIndex].Cells[0].Value
    .ToString());
    txtIdTransaksi.Text = trx_id.ToString();
    txtNamaPembeli.Text =
    gridTransaksi.Rows[e.RowIndex].Cells[1].Value.ToString();
    txtTotal.Text =
    gridTransaksi.Rows[e.RowIndex].Cells[2].Value.ToString();
}

```

```

string sql = "SELECT "
            + "b.kode, "
            + "b.nama, "
            + "b.harga, "
            + "td.jumlah, "
            + "(td.jumlah * b.harga) AS total "
            + "FROM transaksi_detail AS td "
            + "INNER JOIN barang as b "
            + "ON td.barang_id = b.kode "
            + "WHERE td.trx_id=@trx_id;";

MySQLCommand command = new MySQLCommand();
command.Connection = koneksi.getConn();
command.Connection.Open();
command.CommandType = CommandType.Text;
command.CommandText = sql;
command.Parameters.AddWithValue("@trx_id", trx_id);
MySQLDataReader reader = command.ExecuteReader();
while (reader.Read())
{
    gridDetail.Rows.Add(
        reader.GetString(0),
        reader.GetString(1),
        reader.GetString(2),
        reader.GetString(3),
        reader.GetString(4)
    );
}
command.Connection.Close();
}
catch (Exception ex)
{
    MessageBox.Show("get\n" + ex.ToString());
}

```

- I. Double klik pada button Hapus Transaksi (btnHapusTransaksi) dan masukan script berikut:

```

DialogResult akses = MessageBox.Show("Apakah anda yakin?",
    "Informasi", MessageBoxButtons.YesNo,
    MessageBoxIcon.Question);
if (akses == DialogResult.Yes)
{
    try
    {
        MySQLCommand command = new MySQLCommand();
        command.Connection = koneksi.getConn();
        command.Connection.Open();
        command.CommandType = CommandType.Text;
        command.Parameters.Clear();
        command.CommandText = "DELETE FROM transaksi where
trx_id=@trx_id";
        command.Parameters.AddWithValue("@trx_id",
txtIdTransaksi.Text);
    }
}

```

```

        command.ExecuteNonQuery();
        command.Connection.Close();

        command.Connection.Open();
        command.Parameters.Clear();
        command.CommandText = "DELETE FROM transaksi_detail
where trx_id=@trx_id";
        command.Parameters.AddWithValue("@trx_id",
txtIdTransaksi.Text);
        command.ExecuteNonQuery();
        command.Connection.Close();

        MessageBox.Show("Data berhasil dihapus", "info",
MessageBoxButtons.OK, MessageBoxIcon.Information);
        loadDaftarTransaksi();
        clearGridDetail();
        clearTransaksi();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Data gagal disimpan\n" + ex, "info",
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

- m. Selanjutnya silahkan tambahkan fitur pencarian pada tabel transaksi. Pencarian dapan menggunakan text ataupun rentang tanggal.