

*Pertemuan ke 11*  
*Contoh Kasus*



***Pemrograman 2***

Dosen : Eko Budi Setiawan, S.Kom

Universitas Komputer Indonesia

***Sistem informasi penjualan barang***

## Spesifikasi Aplikasi

- Dapat mengelola data Barang, Pelanggan, Supplier, dan Transaksi Penjualan. Untuk setiap pengolahan data, harus tersedia fasilitas Penambahan, Perubahan, Penghapusan dan Pencarian Data.
- Dapat melakukan operasi penjualan Barang
- Dapat mencetak laporan Barang, Pelanggan, Supplier dan Penjualan baik keseluruhan atau berdasarkan kriteria tertentu

## Struktur Database

- Data Barang
- Data Pelanggan
- Data Supplier
- Data Penjualan



## Spesifikasi Data Barang

- Kode Barang harus dapat dimasukan dengan menggunakan pengkodean Barcode
- Data barang harus menampung gambar produk
- Harga dibedakan menjadi HargaBeli dan HargaJual
- Data barang harus dapat digunakan untuk membuat rencana penambahan barang, ketika stok telah menipis. Hal ini bisa dilakukan dengan membuat field Stok dan Stok Minimal
- Dapat melakukan pencarian pada Field Kode, Nama, Harga Jual dan Stok



## Struktur Data Barang

Field Name	Type	Size	Key
Kode	A	13	*
Nama	A	50	
HargaBeli	\$		
HargaJual	\$		
Stok	I		
StokMinimal	I		
Gambar	G		

## Secondary Index Data Barang

Nama Index	Index Field	Maintained
IdxNama	Nama	√
IdxHargaJual	HargaJual	√
IdxStok	Stok	√



## Struktur Data Supplier

Field Name	Type	Size	Key
Kode	+		*
Nama	A	50	
Alamat	M	240	
Kota	A	20	
NoTelepon	A	20	
Fax	A	20	
Email	A	50	

## Secondary Index Data Supplier

Nama Index	Index Field	Maintained
IdxNama	Nama	√
IdxKota	Kota	√
IdxEmail	Email	√



## Struktur Data Pelanggan

Field Name	Type	Size	Key
Kode	+		*
Nama	A	50	
Alamat	M	240	
Kota	A	20	
NoTelepon	A	20	
Fax	A	20	
Email	A	50	

## Secondary Index Data Pelanggan

Nama Index	Index Field	Maintained
IdxNama	Nama	√
IdxKota	Kota	√
IdxEmail	Email	√



## Spesifikasi Data Penjualan

- ❑ Sebuah faktur memiliki nomor faktur berupa angka yang terus bertambah (Autoincrement)
- ❑ Sebuah faktur boleh terdiri dari lebih dari 1 item penjualan. Ini menandakan bahwa kita memiliki 2 tabel yaitu tabel Penjualan (Faktur) dan tabel isi Faktur (Detail Penjualan) yang mempunyai relasi One – To – Many (Satu ke Banyak)
- ❑ Data yang disimpan dalam faktur terdiri dari pelanggan, tanggal dan waktu transaksi, total pembayaran
- ❑ Data yang disimpan dalam item faktur adalah data barang, harga, qty





## Struktur Data Penjualan

Field Name	Type	Size	Key
Nomor	+		*
KodePelanggan	I		
WaktuTransaksi	@		
Total	\$		

## Struktur Data Detail Penjualan

Field Name	Type	Size	Key
NomorPenjualan	I		*
KodeBarang	A	13	*
Qty	I		
Harga	\$		

Silahkan Anda buat alias semua tabel dengan nama Alias = penjualan





## Langkah 1 : Form Utama

- Buka Delphi
- Aktifkan Form1, ganti properti berikut :
  - Name : Futama (Form Utama)
  - Caption : Program Penjualan PT.Banjar Idaman
- Simpan form utama dengan nama U\_Futama.pas
- Simpan project dengan nama Penjualan

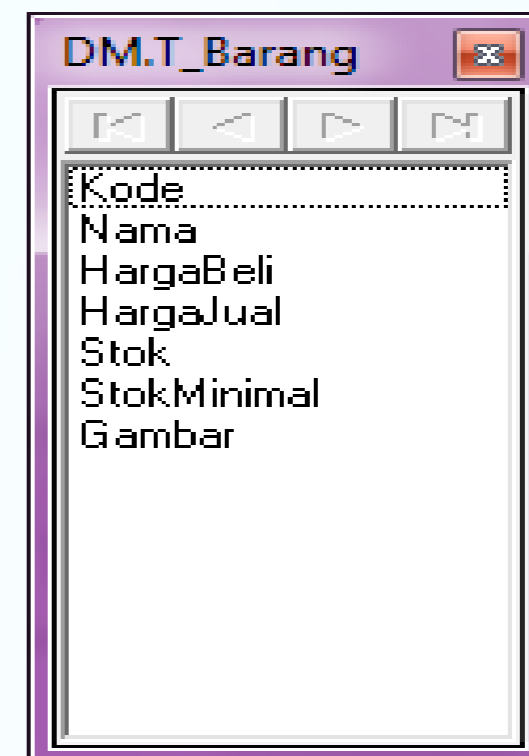
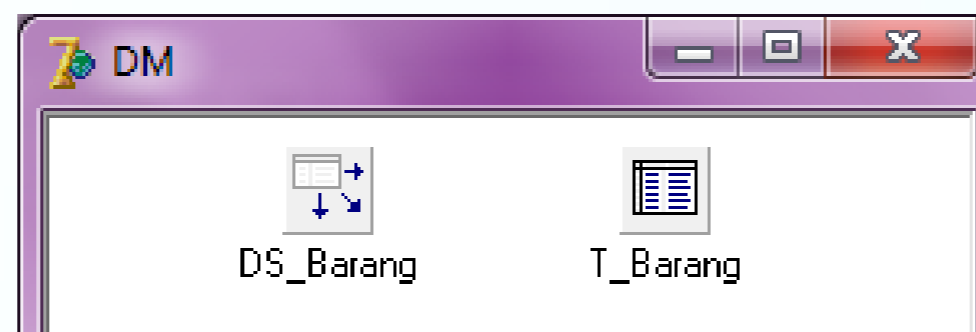
## Langkah 2 : Membuat Data Modul

- Data Module digunakan untuk memusatkan komponen-komponen database, seperti datasource, table, dan lain-lain
- Langkah membuat Data Module adalah
  - Klik File → New → Data Module
  - Klik Data Module, ganti properti Name menjadi DM
  - Simpan Data Module dengan mengklik File → Save  
Simpan dengan nama file U\_DM.pas



## Data Module (lanjutan)

1. Tempatkan sebuah komponen TTable ke Data Module DM. Isinya
  - DatabaseName : penjualan (sesuai dengan nama Alias)
  - TableName : barang.DB
  - Name : T\_barang
  - Active : True
2. Tempatkan sebuah TDataSource ke Data Module DM. Isinya
  - Dataset : T\_Barang
  - Name : DS\_Barang
3. Untuk mempermudah dalam mengakses field, buatlah komponen-komponen fieldnya dengan cara :
  - Double klik di tabel T\_Barang
  - Klik kanan di Field Editor, klik Add All Fields



## Form Data Barang (Fbarang)

1. Buatlah sebuah Form untuk mengolah data Barang, dengan cara Klik File → New → Form
2. Ganti Properti dari Form tersebut menjadi :
  - Caption : Pengolahan Data Barang
  - Name : Fbarang
3. Simpan form tersebut dengan nama file U\_Fbarang.pas
4. Karena Form Fbarang ini akan menggunakan data dari T\_barang yang ada di Data Module DM, maka form ini harus dikenalkan dengan data module DM, dengan cara File → Use Unit. Pilih Unit UDM, kemudian klik tombol OK
5. Aturlah Fbarang seperti halaman berikutnya



# Form Data Barang (Fbarang)

**PENGOLAHAN DATA BARANG**

Kode	Nama	H.
------	------	----

DBGrid1

**Data Diurutkan Berdasarkan**

- Kode
- Nama Barang
- Harga Jual
- Stok

**Pencarian Data**

E\_Cari [ ] Cari

DBNavigator1 [ < > << >> ]

B\_Tambah [ Tambah ] [ Edit ]

B\_Hapus [ Hapus ] [ Laporan ]

[ Tutup ]

RB\_Kode

RB\_Nama

RB\_HargaJual

RB\_Stok

B\_Cari

B\_Edit

B\_Laporan

B\_Tutup



# Properties Form Data Barang (Fbarang)

1. DBGrid, dengan property  
Name : DBGrid1  
DataSource : DM.DS\_Barang
2. DBNavigator  
Name : DBNavigator1  
DataSource : DM.DS\_Barang  
VisibleButton : nbFirst, nbPrior, nbNext, nbLast
3. Button  
Name : B\_Cari  
Caption : Cari
4. Button  
Name : B\_Tambah  
Caption : Cari
5. Button  
Name : B\_Hapus  
Caption : Hapus
6. Button  
Name : B\_Edit  
Caption : Edit
7. Button  
Name : B\_Laporan  
Caption : Laporan
8. Button  
Name : B\_Tutup  
Caption : Tutup
9. RadioButton  
Name : Rb\_Kode  
Caption : Kode
10. RadioButton  
Name : Rb>Nama  
Caption : Nama Barang
11. RadioButton  
Name : Rb\_HargaJual  
Caption : HargaJual
12. RadioButton  
Name : Rb\_Stok  
Caption : Stok
13. Button  
Name : E\_Cari



# Event Form Data Barang (Fbarang)

## 1. Event OnClick tombol Tutup

```
procedure TFBarang.B_TutupClick(Sender: TObject);  
begin  
    Close;  
end;
```

## 2. Event OnClick tombol Rb\_Kode

```
procedure TFBarang.RB_KodeClick(Sender: TObject);  
begin  
    DM.T_Barang.IndexName:='';  
end;
```

## 3. Event OnClick tombol Rb\_Nama

```
procedure TFBarang.RB_NamaClick(Sender: TObject);  
begin  
    DM.T_Barang.IndexName:='idxNama';  
end;
```

## 4. Event OnClick tombol Rb\_Harga Jual

```
procedure TFBarang.RB_HargaJualClick(Sender: TObject);  
begin  
    DM.T_Barang.IndexName:='idxHargaJual';  
end;
```





# Event Form Data Barang (Fbarang)

## 5. Event OnClick tombol RB\_Stok

```
procedure TFBarang.RB_StokClick(Sender: TObject);  
begin  
    DM.T_Barang.IndexName:='idxStok';  
end;
```

## 6. Event OnClick tombol B\_Cari

```
procedure TFBarang.B_CariClick(Sender: TObject);  
var  
    ketemu:boolean;  
begin  
    //Cari berdasarkan key sesuaikan dengan isi Text yang ada di E_Cari  
    ketemu:=DM.T_Barang.FindKey([E_Cari.Text]);  
    // ganti FindKey dengan FindNearest jika ingin pencarian  
    // menggunakan metode pencarian data terdekat (boleh sebagian)  
    if ketemu=false then  
        showmessage('Data Tidak Ditemukan !');  
end;
```





# Event Form Data Barang (Fbarang)

## 7. Event OnClick tombol B\_Hapus

```
procedure TFBarang.B_HapusClick(Sender: TObject);
```

```
var
```

```
    Pesan:String;
```

```
begin
```

```
    Pesan:='Anda yakin data berikut : '+#13+#13+ // #13 = Enter (pindah baris)
```

```
    'Kode Barang : '+DM.T_BarangKode.value+#13+
```

```
    'Nama Barang : '+DM.T_BarangNama.value+#13+#13+
```

```
    'akan dihapus?';
```

```
    if MessageDlg(Pesan,mtConfirmation,[mbYes,mbNo],0)=mrYes then
```

```
    begin
```

```
        DM.T_Barang.Delete;
```

```
    end;
```

```
End;
```

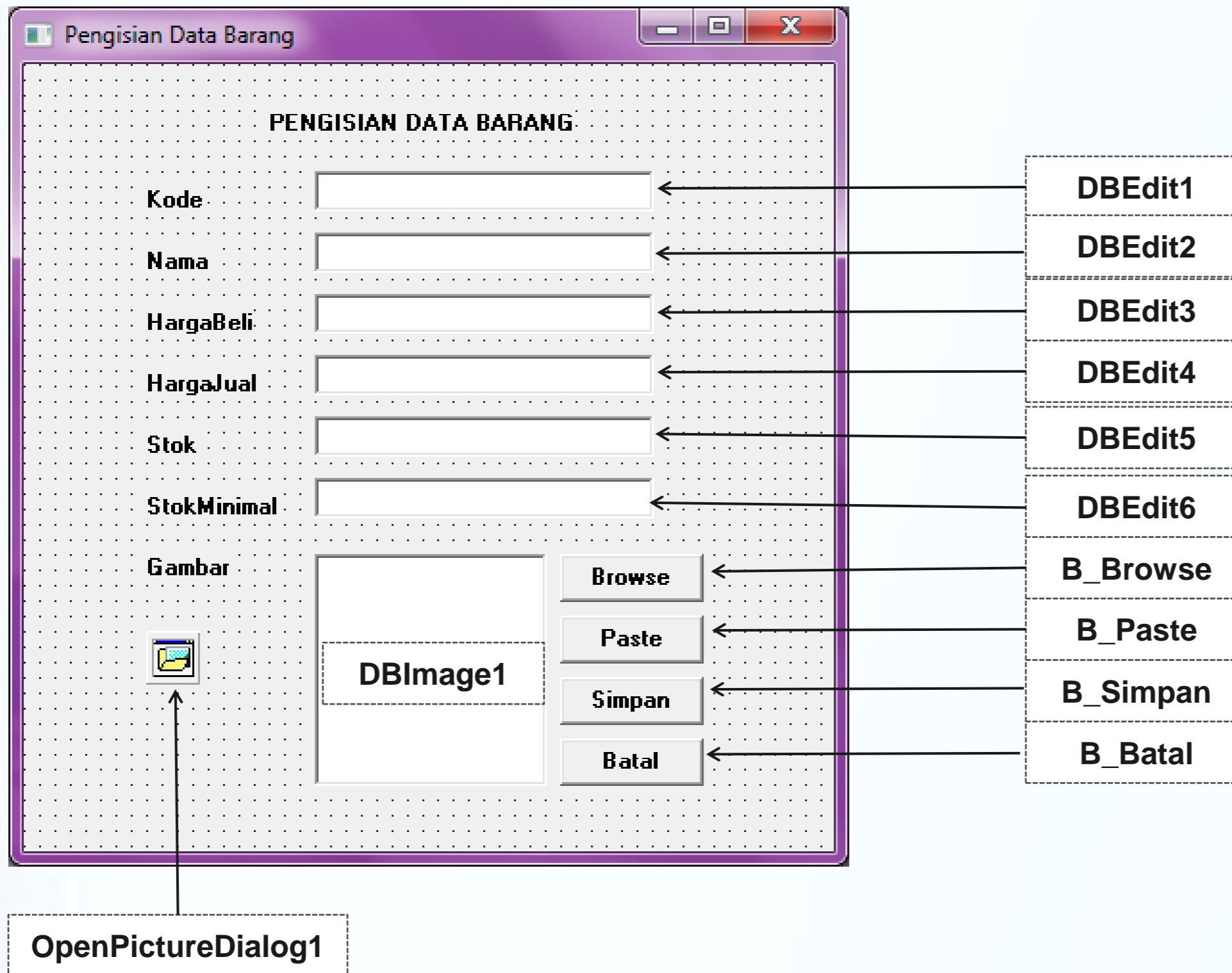


## Form Isi Data Barang (FIsiBarang)

1. Buatlah sebuah Form untuk melakukan pengisian data barang, baik untuk penambahan data atau pengeditan data. Klik File → New → Form
2. Ganti Properti dari Form tersebut menjadi :
  - Caption : Pengisian Data Barang
  - Name : FIsiBarang
3. Simpan form tersebut dengan nama file U\_Flsibarang.pas
4. Karena Form Fbarang ini akan menggunakan data dari T\_barang yang ada di Data Module DM, maka form ini harus dikenalkan dengan data module DM, dengan cara File → Use Unit. Pilih Unit U\_DM, kemudian klik tombol OK
5. Aturlah Fbarang seperti halaman berikutnya



# Form Isi Data Barang (FIsiBarang)



# Properties Form Isi Data Barang (FIsibarang)

1. DBEdit1, dengan property  
DataSource : DM.DS\_Barang  
DataField : Kode
2. DBEdit2  
DataSource : DM.DS\_Barang  
DataField : Nama
3. DBEdit3  
DataSource : DM.DS\_Barang  
DataField : HargaBeli
4. DBEdit4  
DataSource : DM.DS\_Barang  
DataField : HargaJual
5. DBEdit5  
DataSource : DM.DS\_Barang  
DataField : Stok
6. DBEdit6  
DataSource : DM.DS\_Barang  
DataField : HargaBeli
7. DBImage1, dengan property  
DataSource : DM.DS\_Barang  
DataField : Gambar
8. OpenPictureDialog1  
terdapat di Tab Komponen Dialog
9. B\_Simpan  
Caption : Simpan
10. B\_Batal  
Caption : Batal
11. B\_Browse  
Caption : Browse
12. B\_Paste  
Caption : Paste

**Jangan lupa untuk menggunakan  
File → Use Unit**



# Event Form Isi Data Barang (FIsiBarang)

## 1. Event OnClick tombol Browse

```
procedure TFIsiBarang.B_BrowseClick(Sender: TObject);
begin
  if OpenPictureDialog1.Execute then
  begin
    DM.T_BarangGambar.LoadFromFile(OpenPictureDialog1.FileName);
  end;
```

## 2. Event OnClick tombol Simpan

```
procedure TFIsiBarang.B_SimpanClick(Sender: TObject);
begin
  DM.T_Barang.Post; // Simpan data
  Close;// Tutup form FIsiBarang (Kembali ke FBarang)
end;
```

## 3. Event OnClick tombol Batal

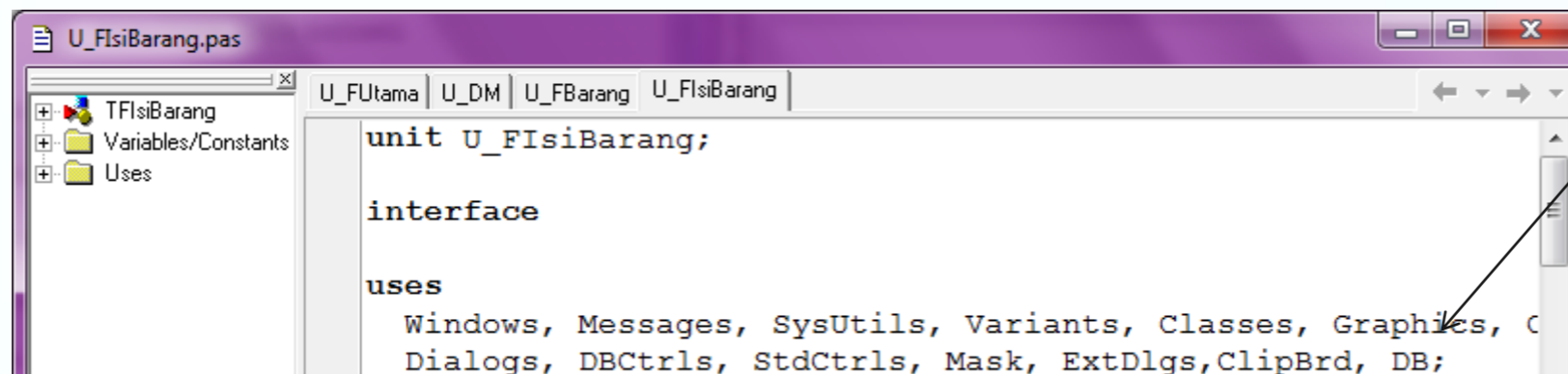
```
procedure TFIsiBarang.B_BatalClick(Sender: TObject);
begin
  if MessageDlg('Pengisian data dibatalkan ?',mtConfirmation,
    [mbYes,mbNo],0)=mrYes then
  begin
    DM.T_Barang.Cancel;
    Close;// Tutup form FIsiBarang (Kembali ke FBarang)
  end;
```



## Event Form Isi Data Barang (FIsiBarang)

Pada FisiBarang kita menginginkan bahwa user hanya akan keluar kalau menekan tombol Simpan atau Batal. FisiBarang yang tadi dibuat masih mempunyai kelemahan yaitu form dapat ditutup dengan mengklik Close (X). Cara menanganinya adalah, isi event OnCloseQuery di FIsiBarang, dengan :

```
procedure TFIsiBarang.FormCloseQuery(Sender: TObject;  
    var CanClose: Boolean);  
begin  
    if DM.T_Barang.State in [dsInsert, dsEdit] then  
        begin  
            CanClose:=False; // Layar tidak boleh ditutup  
            ShowMessage('Gunakan tombol Simpan atau Batal untuk menutup window ini');  
        end;  
end;  
end;
```



Tambahkan DB di  
uses U\_IsiBarang





## Event Form Isi Data Barang (FIsibarang)

4. **Kembali ke Fbarang (gunakan F12) untuk berganti Form. Isi event OnClick B\_Tambah dengan :**

```
procedure TFBarang.B_TambahClick(Sender: TObject);  
begin  
    DM.T_Barang.Append;  
    FIsiBarang.ShowModal;  
end;
```

5. **Masih di Fbarang (gunakan F12). Isi event OnClick B\_Edit dengan :**

```
procedure TFBarang.B_EditClick(Sender: TObject);  
begin  
    DM.T_Barang.Edit;  
    FIsiBarang.ShowModal;  
end;
```

6. **Kembali ke Futama (gunakan F12) untuk berganti Form. Tambahkan sebuah Tombol "DATA BARANG" Isi event tombol tersebut dengan :**

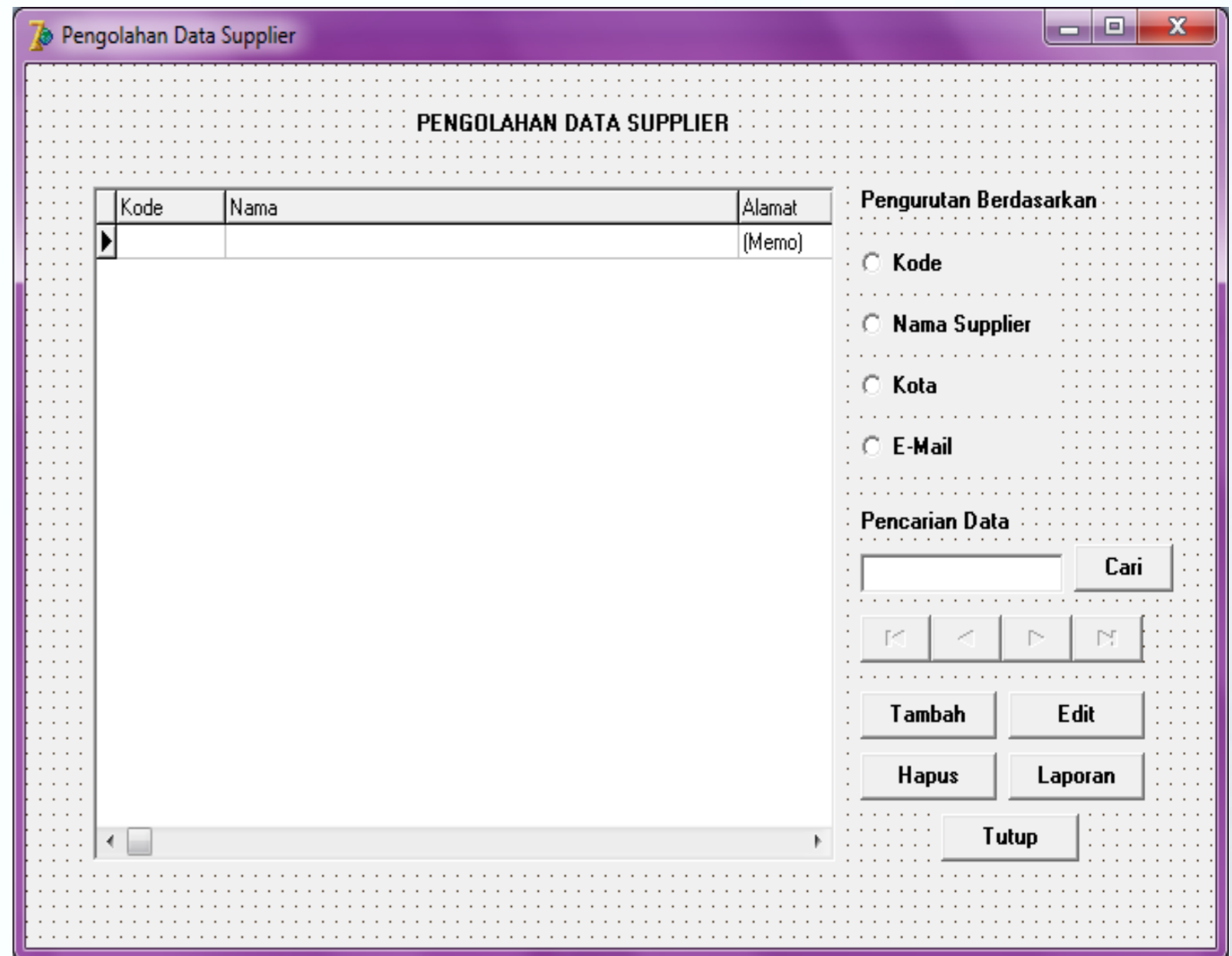
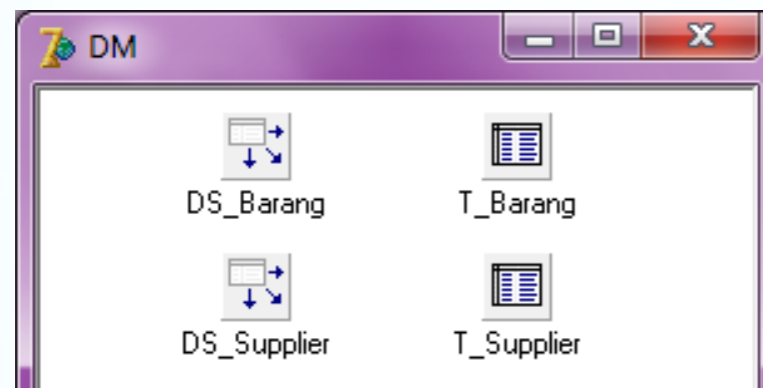
```
procedure TFUtama.Button1Click(Sender: TObject);  
begin  
    FBarang.ShowModal;  
end;
```





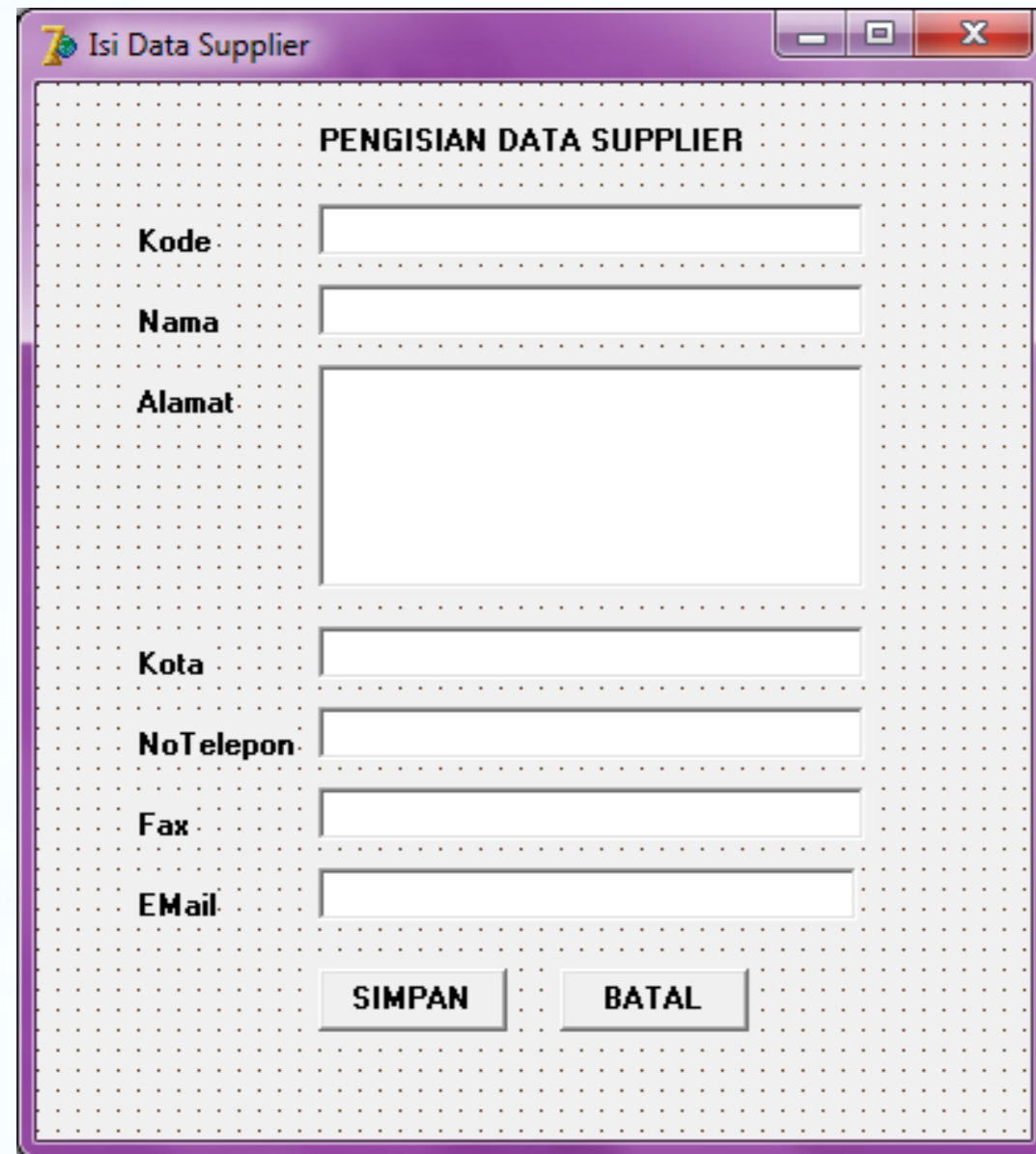
# Pengolahan Data Supplier (Fsupplier)

1. Langkah-langkah untuk pengolahan Data Supplier mirip dengan pengolahan data Barang. Hanya saja semua yang mengakses T\_Barang diganti dengan T\_Supplier. Simpan dengan nama **FSupplier**
2. Tempatkan Tabel ke DataModule DM untuk mengakses data Supplier



# Form Isi Data Supplier (FIsiSupplier)

1. Buatlah Form untuk pengolahan data Supplier, Name : FIsiSupplier



The screenshot shows a Windows-style window titled "Isi Data Supplier". Inside the window, the text "PENGISIAN DATA SUPPLIER" is centered at the top. Below this, there are seven input fields, each with a label to its left: "Kode", "Nama", "Alamat", "Kota", "NoTelepon", "Fax", and "EMail". The "Alamat" field is significantly larger than the others. At the bottom of the form, there are two buttons: "SIMPAN" and "BATALL". The window has a standard Windows XP-style title bar with minimize, maximize, and close buttons.

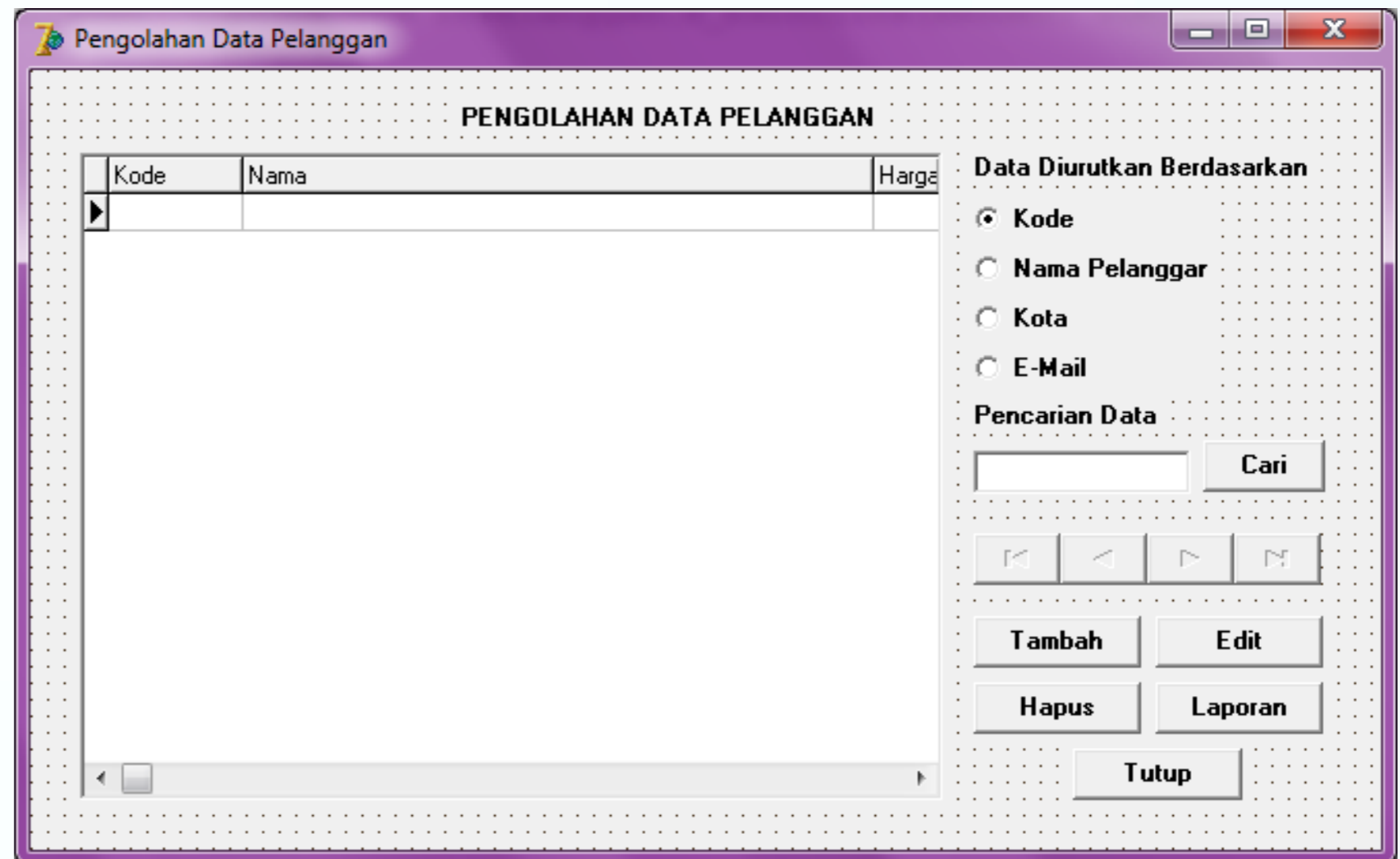
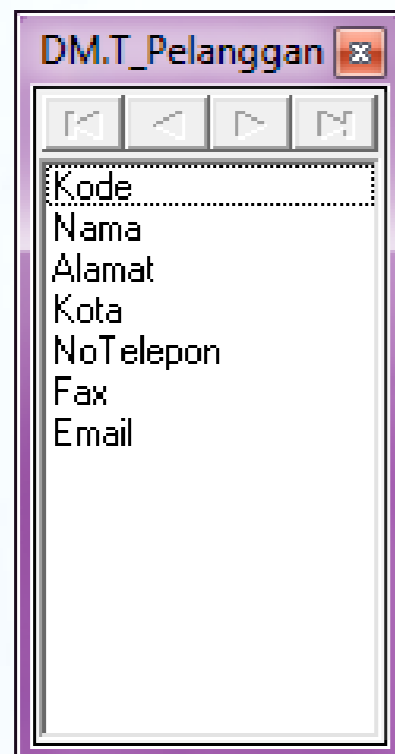
2. Kembali ke Futama, kemudian berikan Tombol “DATA SUPPLIER” dan isikan perintah berikut :

**FSupplier.ShowModal;**



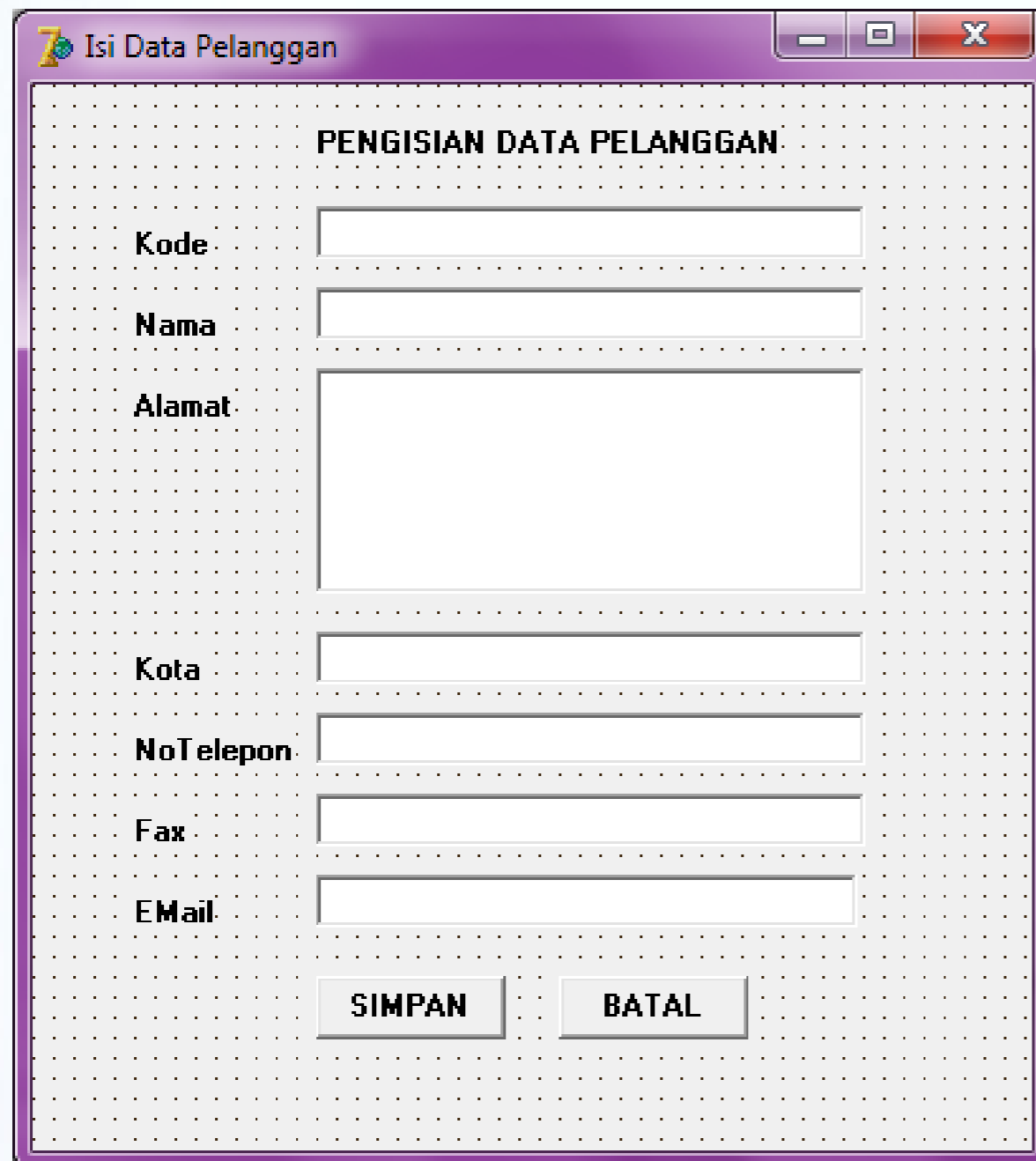
# Pengolahan Data Pelanggan (Fpelanggan)

1. Langkah-langkah untuk pengolahan Data Pelanggan mirip dengan pengolahan data Supplier. Hanya saja semua yang mengakses T\_Supplier diganti dengan T\_Pelanggan. Simpan dengan nama **Fpelanggan**



# Form Isi Data Pelanggan (FIsiPelanggan)

1. Buatlah Form untuk pengolahan data Pelanggan, Name: FIsiPelangan



The screenshot shows a Windows application window titled "Isi Data Pelanggan". The window contains a form with the following fields and controls:

- PENGISIAN DATA PELANGGAN:** (Section header)
- Kode:** Text input field
- Nama:** Text input field
- Alamat:** Text area
- Kota:** Text input field
- NoTelepon:** Text input field
- Fax:** Text input field
- EMail:** Text input field
- SIMPAN:** Button
- BATAL:** Button

2. Kembali ke Futama, kemudian berikan Tombol “DATA PELANGGAN” dan isikan perintah berikut :

**FPelanggan.ShowModal;**



# Pengolahan Data Penjualan

## 1. Tempatkan 2 buah Table dan 2 buah Datasource ke DM

Tabel Pertama

Name : T\_Penjualan

TableName : Penjualan.DB

Active : True

Tabel Kedua

Name : T\_DetailPenjualan

TableName : Penjualan.DB

Active : True

Datasoure Pertama

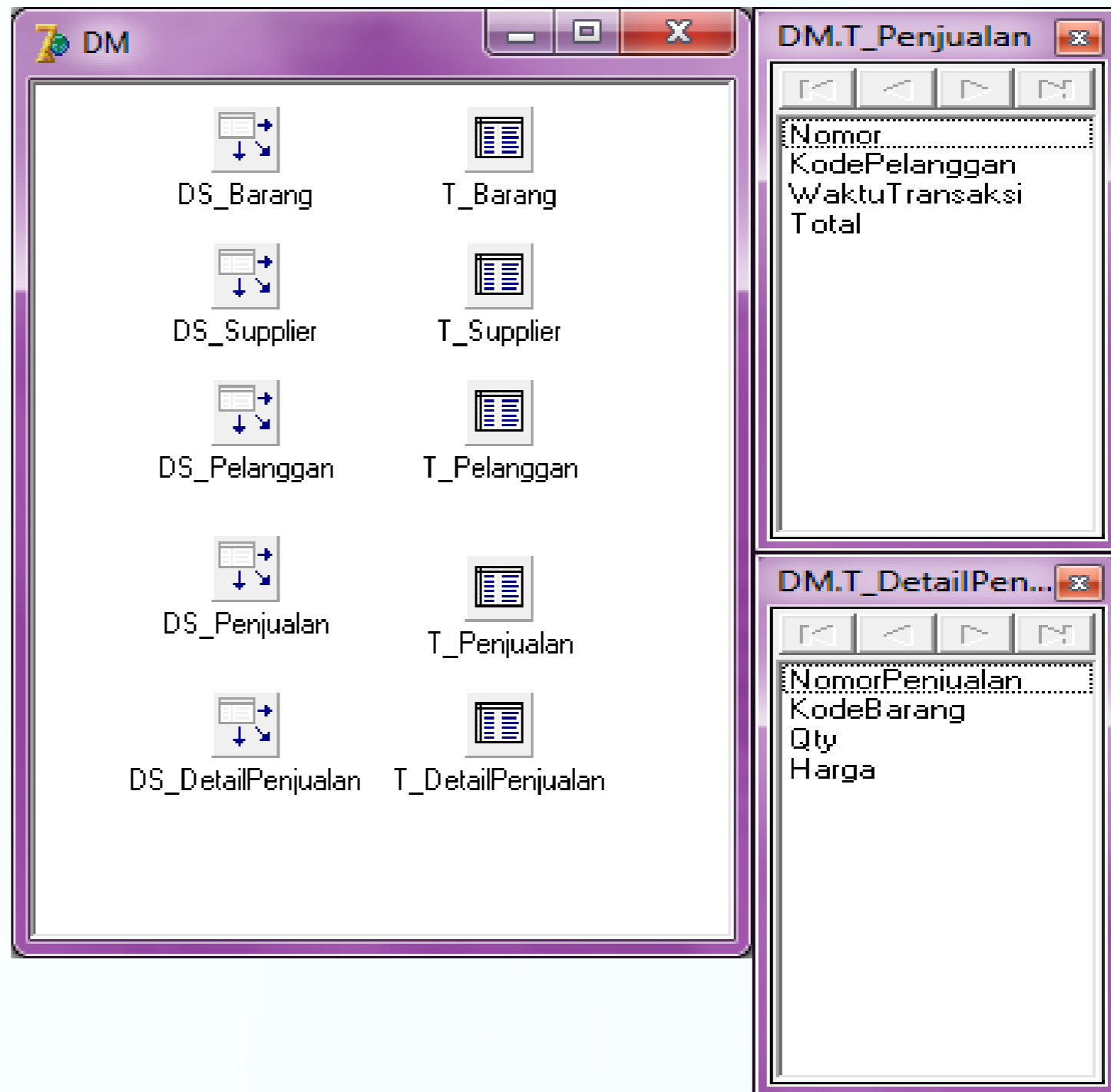
Name : DS\_Penjualan

DataSet : T\_Penjualan

Datasoure Kedua

Name : DS\_DetailPenjualan

DataSet : T\_DetailPenjualan



# Pengolahan Data Penjualan (Fpenjualan)

1. Buatlah Form baru untuk pengolahan data penjualan
  - File → New → Form
  - Ganti PropertiName : FPenjualan
  - Caption : Pengolahan Data Penjualan
  - File → Save dengan nama U\_FPenjualan

**PENGOLAHAN DATA PENJUALAN**

Nomor	KodePelanggan	WaktuTransaksi	Total
▶			

**PERINCIAN FAKTUR PENJUALAN**

NomorPenjualan	KodeBarang	Qty	Harga
▶			

◀ ◁ ▷ ▶ Penjualan Baru Lihat Faktur Tutup





# Properties Form Data Penjualan (Fpenjualan)

1. DBGrid1 dengan property  
DataSource : DM.DS\_Penjualan  
Name : DBGrid1
2. DBGrid2 dengan property  
DataSource : DM.DS\_DetailPenjualan  
Name : DBGrid2
3. DBNavigator  
DataSource : DM.DS\_Penjualan  
Name : DBNavigator1  
VisibleButron : nbFirst, nbPrior, nbNext,nbLast
5. Button  
Caption : Penjualan Baru  
Name : B\_PenjualanBaru
6. Button  
Caption : Lihat Faktur  
Name : B\_LihatFaktur
7. Button  
Caption : Tutup  
Name : B\_Tutup

**Jangan lupa untuk menggunakan  
File → Use Unit**

