

Lampiran I

LISTING PROGRAM

Pembuatan Database:

1. Pembuatan tabel Barang

```
CREATE TABLE "BARANG"  
(  
  "KODEBRG"      CHAR(8) NOT NULL,  
  "NAMABRG"     CHAR(30),  
  "HRGBELI"     NUMERIC(9, 2),  
  "HRGJUAL"     NUMERIC(9, 2),  
  "STOK"        NUMERIC(9, 2),  
  "KATEGORI"    CHAR(10),  
  CONSTRAINT "PKKODEBRG" PRIMARY KEY ("KODEBRG")  
);
```

2. Pembuatan tabel Beli

```
CREATE TABLE "BELI"  
(  
  "BONBELI"     CHAR(6) NOT NULL,  
  "TGL"         DATE,  
  "KODESUPP"    CHAR(6),  
  CONSTRAINT "PKBONBELI" PRIMARY KEY ("BONBELI")  
);  
ALTER TABLE "BELI" ADD CONSTRAINT "FKKODESUPP" FOREIGN  
KEY ("KODESUPP") REFERENCES SUPPLIER ("KODESUPP");
```

3. Pembuatan tabel Beli Item

```
CREATE TABLE "BELIITEM"  
(  
  "BONBELI"     CHAR(6) NOT NULL,  
  "KODEBRG"     CHAR(8) NOT NULL,  
  "QTY"         NUMERIC(9, 2),  
  "HRGBELI"     NUMERIC(9, 2),  
  CONSTRAINT "PKBELIBRG" PRIMARY KEY ("BONBELI",  
  "KODEBRG")
```

- ```
);
```
4. Pembuatan tabel Jual
 

```
CREATE TABLE "JUAL"
(
 "BONJUAL" CHAR(6) NOT NULL,
 "TGL" DATE,
 "KODECUST" CHAR(6),
 "MEJA" CHAR(4),
 "STATUS" CHAR(1),
 CONSTRAINT "PKBONJUAL" PRIMARY KEY ("BONJUAL")
);
ALTER TABLE "JUAL" ADD CONSTRAINT "FKKODECUST"
FOREIGN KEY ("KODECUST") REFERENCES CUSTOMER
("KODECUST");
```
  5. Pembuatan tabel Jual Item
 

```
CREATE TABLE "JUALITEM"
(
 "BONJUAL" CHAR(6) NOT NULL,
 "KODEBRG" CHAR(8) NOT NULL,
 "QTY" NUMERIC(9, 2),
 "HRGJUAL" NUMERIC(9, 2),
 CONSTRAINT "PKBONBRG" PRIMARY KEY ("BONJUAL",
 "KODEBRG")
);
```
  6. Pembuatan tabel Customer
 

```
CREATE TABLE "CUSTOMER"
(
 "KODECUST" CHAR(8) NOT NULL,
 "NAMACUST" CHAR(30),
 "ALAMATCUST" CHAR(100),
 "TLP" CHAR(15),
 "HP" CHAR(15),
 CONSTRAINT "PKKODECUST" PRIMARY KEY ("KODECUST")
);
```
  7. Pembuatan tabel Supplier
 

```
CREATE TABLE "SUPPLIER"
(
```

```

"KODESUPP" CHAR(8) NOT NULL,
"NAMASUPP" CHAR(30),
"ALAMATSUPP" CHAR(100),
"TLP" CHAR(5),
"HP" CHAR(15),
CONSTRAINT "PKKODESUPP" PRIMARY KEY ("KODESUPP")
);

```

Pembuatan Program:

- a. Pembuatan program form Menu Awal  
unit UMenuAwal;

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Menus, StdCtrls, XPMAN;
```

```
type
```

```
TFrmMenuAwal = class(TForm)
 MainMenu1: TMainMenu;
 Menu1: TMenuItem;
 DaftarMenu1: TMenuItem;
 InputMakanan1: TMenuItem;
 Customer1: TMenuItem;
 Supplier1: TMenuItem;
 ransaksi1: TMenuItem;
 Pembelian1: TMenuItem;
 PenjualanResto1: TMenuItem;
 Laporan1: TMenuItem;
 PembelianBarang1: TMenuItem;
 PenjualanResto2: TMenuItem;
 Keluar1: TMenuItem;
 Label1: TLabel;
 XPManifest1: TXPManifest;
 procedure DaftarMenu1Click(Sender: TObject);
end;

```

```

 procedure InputMakanan1Click(Sender: TObject);
 procedure Keluar1Click(Sender: TObject);
 procedure Supplier1Click(Sender: TObject);
 procedure Pembelian1Click(Sender: TObject);
 procedure Customer1Click(Sender: TObject);
 procedure PembelianBarang1Click(Sender: TObject);
 procedure PenjualanResto1Click(Sender: TObject);
 procedure PenjualanResto2Click(Sender: TObject);
 private
 { Private declarations }
 public
 { Public declarations }
 end;

var
 FrmMenuAwal: TFrmMenuAwal;

implementation
Uses UModule, UDaftar, UInput, UCustomer, USupp, UBeliVarang,
ULogin1, ULogin2,
 ULogin4, ULogin3, ULogin5, ULogin6, ULogin7;
{$R *.dfm}

procedure TFrmMenuAwal.DaftarMenu1Click(Sender: TObject);
begin
 FrmDaftarMenu.Show;
end;

procedure TFrmMenuAwal.InputMakanan1Click(Sender: TObject);
begin
 Form7.Show;
end;

procedure TFrmMenuAwal.Keluar1Click(Sender: TObject);
begin
 Application.terminate;
end;

```

```
procedure TFrmMenuAwal.Supplier1Click(Sender: TObject);
begin
Form2.Show;
end;
```

```
procedure TFrmMenuAwal.Pembelian1Click(Sender: TObject);
begin
Form3.Show;
end;
```

```
procedure TFrmMenuAwal.Customer1Click(Sender: TObject);
begin
Form1.Show;
end;
```

```
procedure TFrmMenuAwal.PembelianBarang1Click(Sender: TObject);
begin
Form4.show;
end;
```

```
procedure TFrmMenuAwal.PenjualanResto1Click(Sender: TObject);
begin
Form5.show;
end;
```

```
procedure TFrmMenuAwal.PenjualanResto2Click(Sender: TObject);
begin
Form6.show;
end;
```

end.

- b. Pembuatan program form Modul  
unit UModule;

```
interface
```

```
uses
 SysUtils, Classes, DB, IBCustomDataSet, IBQuery, IBDatabase;
```

```
type
 TdmMakanan = class(TDataModule)
 TrMakanan: TIBTransaction;
 qsdata: TDataSource;
 qsProses: TDataSource;
 DbMakanan: TIBDatabase;
 QData: TIBQuery;
 QProses: TIBQuery;
 private
 { Private declarations }
 public
 { Public declarations }
 end;
```

```
var
 dmMakanan: TdmMakanan;
```

```
implementation
```

```
{ $R *.dfm }
```

```
end.
```

- c. Pembuatan program form Daftar Menu  
unit UDaftar;

```
interface
```

```
uses
 Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
 Dialogs, StdCtrls;
```

```
type
 TFrmDaftarMenu = class(TForm)
 Label1: TLabel;
```

Label2: TLabel;  
Label3: TLabel;  
Label4: TLabel;  
Label5: TLabel;  
Label6: TLabel;  
EdMeja: TEdit;  
EdTS: TEdit;  
EdSS: TEdit;  
EdTBS: TEdit;  
EdS: TEdit;  
EdNT: TEdit;  
EdNG: TEdit;  
EdMG: TEdit;  
EdAG: TEdit;  
EdAB: TEdit;  
EdNP: TEdit;  
EdCC: TEdit;  
EdIFM: TEdit;  
EdKC: TEdit;  
EdCo: TEdit;  
EdSp: TEdit;  
EdF: TEdit;  
EdJS: TEdit;  
EdJA: TEdit;  
EdJM: TEdit;  
EdJSr: TEdit;  
EdETM: TEdit;  
EdETT: TEdit;  
BtSimpan: TButton;  
BtKeluar: TButton;  
Label7: TLabel;  
Label8: TLabel;  
Label9: TLabel;  
Label10: TLabel;  
Label11: TLabel;  
Label12: TLabel;  
Label13: TLabel;

```

Label14: TLabel;
Label15: TLabel;
Label16: TLabel;
Label17: TLabel;
Label18: TLabel;
Label19: TLabel;
Label20: TLabel;
Label21: TLabel;
Label22: TLabel;
Label23: TLabel;
Label24: TLabel;
Label25: TLabel;
Label26: TLabel;
Label27: TLabel;
Label28: TLabel;
Button1: TButton;
procedure BtKeluarClick(Sender: TObject);
procedure BtSimpanClick(Sender: TObject);
procedure hitungbon();
procedure FormShow(Sender: TObject);
procedure Button1Click(Sender: TObject);

// procedure EdMejaChange(Sender: TObject);
// procedure FormShow(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmDaftarMenu: TFrmDaftarMenu;
 D, B , C : String;
implementation
Uses UModule, UJual;
{$R *.dfm}

```



```

procedure TFrmDaftarMenu.hitungbon;
var a: string;//,b
begin

//Tampilkan BonJual (No Bon Jual)
With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select BONJUAL, Meja, Status From JUAL');
Open;

//Jika Ternyata tidak ada (kosong/null) maka
If (Fields[0].IsNull) {or (Fields[2].IsNull)} Then
Begin
a := '00001';
End
Else

//Jika ada nilainya maka, tentukan nobon paling akhir
Begin
With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Max(BonJual) From JUAL');
Open;
//Ini adalah format agar diawali dengan 0000
a := FormatFloat('00000',StrToFloat(Fields[0].Value+1));
End;
End;
End;
//Simpan nobon tersebut ke dalam tabel jual ke field Bonjual
With dmMakanan.qProses Do
Begin
Close;
Sql.Clear;

```

```

 Sql.Add('Insert Into JUAL(BONJUAL,Tgl) Values(:bj, :tg)');
 Prepare;
 ParamByname('bj').AsString := a;
 ParamByName('tg').AsDate := now;
 ExecSql;
 end;
 dmMakanan.TrMakanan.Commit;

end;
procedure TFrmDaftarMenu.BtKeluarClick(Sender: TObject);
begin
Close;
end;

procedure TFrmDaftarMenu.BtSimpanClick(Sender: TObject);
var Stok : double ;
begin

 dmMakanan.DbMakanan.Close;
 dmMakanan.TrMakanan.Active;

hitungbon;

 if EdTS.Text <> '0' Then
 Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString :='ME0001';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;

```

```

dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').AsString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdTS.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].AsFloat;
dmMakanan.qProses.ExecSql;
End;
End;
if EdSS.Text <> '0' Then
Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'ME0002';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').AsString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdSS.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].AsFloat;
 dmMakanan.qProses.ExecSql;
 End;
End ;

```

```

if EdTBS.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'ME0003';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdTBS.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End ;
End ;
 if EdS.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'ME0004';
 Open;
 dmMakanan.qProses.Close;

```

```

dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').AsFloat := StrToFloat(EdS.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End ;
End ;
if EdNT.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString :='MI0001';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdNT.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End;
End;

```

```

if EdNG.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MI0002';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdNG.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End ;
End ;
if EdMG.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MI0003';
 Open;
 dmMakanan.qProses.Close;

```

```

dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdMG.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End ;
End ;
if EdAG.Text <> '0' Then
Begin
With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select KodeBrg, HrgJual from Barang');
Sql.Add('where KodeBrg=:kb');
Prepare;
ParamByName('kb').AsString := 'MI0004';
Open;
dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdAG.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End ;

```

```

End ;
if EdAB.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MI0005';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdAB.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End;
End;
if EdNP.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MI0006';
 Open;

```



```

dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').ASString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdNP.Text);
dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End ;
End ;
if EdCC.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').ASString := 'MC0001';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').ASString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdCC.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;

```

```

 End ;
End;
if EdIFM.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MC0002';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdIFM.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End;
End;
if EdKC.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'MC0003';

```

```

Open;
dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').ASString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdKC.Text);
dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End;
End ;
if EdCo.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').ASString := 'Mn0001';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').ASString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdCo.Text);
 dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;

```

```

 dmMakanan.qProses.ExecSql;
 End;
End;
if EdSp.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'Mn0002';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').AsString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdSp.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].AsFloat;
 dmMakanan.qProses.ExecSql;
 End;
End;
if EdF.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;

```

```

ParamByName('kb').AsString := 'Mn0003';
Open;
dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').AsFloat := StrToFloat(EdF.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End;
End ;
if EdJS.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'Mn0004';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdJS.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;

```

```

 dmMakanan.qProses.ExecSql;
 End;
End ;
if EdJA.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'Mn0005';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').AsString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdJA.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].AsFloat;
 dmMakanan.qProses.ExecSql;
 End;
End ;
if EdJM.Text <> '0' Then
Begin
 With dmMakanan.qData Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;

```

```

ParamByName('kb').AsString := 'Mn0006';
Open;
dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').AsString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdJM.Text);
dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].AsFloat;
dmMakanan.qProses.ExecSql;
End;
End;
if EdJSr.Text <> '0' Then
Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'Mn0007';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').AsString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].AsString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
StrToFloat(EdJSr.Text);

```

```

 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End;
End;
if EdETM.Text <> '0' Then
 Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');
 Prepare;
 ParamByName('kb').AsString := 'Mn0008';
 Open;
 dmMakanan.qProses.Close;
 dmMakanan.QProses.SQL.Clear;
 dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
 HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
 dmMakanan.qProses.Prepare;
 dmMakanan.qProses.ParamByName('bj').ASString := B;
 dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
 dmMakanan.qProses.ParamByName('qy').AsFloat :=
 StrToFloat(EdETM.Text);
 dmMakanan.qProses.ParamByName('hg').AsFloat := Fields[1].ASFloat;
 dmMakanan.qProses.ExecSql;
 End;
 End;
 End;
 if EdETT.Text <> '0' Then
 Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KodeBrg, HrgJual from Barang');
 Sql.Add('where KodeBrg=:kb');

```



```

Prepare;
ParamByName('kb').AsString := 'Mn0009';
Open;
dmMakanan.qProses.Close;
dmMakanan.QProses.SQL.Clear;
dmMakanan.QProses.Sql.Add('Insert into JualItem(BonJual, KodeBrg, Qty,
HrgJual) '+
 'Values(:bj, :kb, :qy, :hg)');
dmMakanan.qProses.Prepare;
dmMakanan.qProses.ParamByName('bj').ASString := B;
dmMakanan.qProses.ParamByName('kb').AsString := Fields[0].ASString;
dmMakanan.qProses.ParamByName('qy').ASFloat :=
StrToFloat(EdETT.Text);
dmMakanan.qProses.ParamByName('hg').ASFloat := Fields[1].ASFloat;
dmMakanan.qProses.ExecSql;
End;
End;

dmMakanan.TrMakanan.Commit;
button1.click;
FrmJual.show;

End;
procedure TFrmDaftarMenu.FormShow(Sender: TObject);
begin

with dmMakanan.QProses Do
Begin
Close;
sql.Clear;
Sql.Add('Select Max(BonJual) From Jual');
Open;
B := Fields[0].AsString;
end;

End;

```

```

procedure TFrmDaftarMenu.Button1Click(Sender: TObject);
var stok: double;
begin

 dmMakanan.DbMakanan.Close;
 dmMakanan.TrMakanan.Active;

 if EdTS.text <> '0' Then
 //Untuk mengurangi jumlah stok barang
 begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'ME0001';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByName('kb').AsString := 'ME0001';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdTs.text);
 dmMakanan.qproses.ExecSql;
 end;
 end;
end;

```

```

if EdSS.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'ME0002';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByname('kb').AsString := 'ME0002';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(Edss.text);
 dmMakanan.qproses.ExecSql;
 end;
 end;
if EdTBS.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;

```

```

Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'ME0003';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'ME0003';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(Edtbs.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdS.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'ME0004';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari

```

```

//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'ME0004';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(Eds.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdNT.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MI0001';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MI0001';

```

```

 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdNT.text);
 dmMakanan.qproses.ExecSql;
 end;
end;
if EdNG.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'MI0002';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByName('kb').AsString := 'MI0002';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdNG.text);
 dmMakanan.qproses.ExecSql;
 end;
end;
if EdMG.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

```

```

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MI0003';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MI0003';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdMG.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdAG.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

```

```

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;

```

```

Parambyname('kb').AsString := 'MI0004';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByName('kb').AsString := 'MI0004';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdAG.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdAB.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MI0005';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;

```



```

dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MI0005';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdAB.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdNP.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MI0006';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MI0006';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdNP.text);
dmMakanan.qproses.ExecSql;

```

```

end;
end;
if EdCC.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MC0001';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MC0001';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdCC.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdIFM.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin

```

```

Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MC0002';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MC0002';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdIFM.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdKC.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'MC0003';
Open;
Stok := Fields[0].AsFloat;

```

```

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'MC0003';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdKC.text);
 dmMakanan.qproses.ExecSql;
end;
end;
if EdCo.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'Mn0001';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');

```

```

 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0001';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdCo.text);
 dmMakanan.qproses.ExecSql;
 end;
end;
if EdSp.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'Mn0002';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0002';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdSp.text);
 dmMakanan.qproses.ExecSql;
 end;
end;
if EdF.text <> '0' Then

```

```

//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'Mn0003';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0003';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
 StrToFloat(EdF.text);
 dmMakanan.qproses.ExecSql;
 end;
 end;
if EdJS.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');

```

```

Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'Mn0004';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0004';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdJS.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdJA.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'Mn0005';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan

```

```

//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0005';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdJA.text);
 dmMakanan.qproses.ExecSql;
end;
end;
if EdJM.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'Mn0006';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0006';

```



```

 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdJM.text);
 dmMakanan.qproses.ExecSql;
 end;
 end;
if EdJSr.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Stok from Barang');
 Sql.Add('Where KodeBrg =:kb');
 Prepare;
 Parambyname('kb').AsString := 'Mn0007';
 Open;
 Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu kurangi dengan qty penjualan
 //stok yang di update harus sesuai dengan kode barang nya.
 dmMakanan.qproses.Close;
 dmMakanan.qproses.SQL.Clear;
 dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
 dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
 dmMakanan.qproses.Prepare;
 dmMakanan.qproses.ParamByName('kb').AsString := 'Mn0007';
 dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdJSr.text);
 dmMakanan.qproses.ExecSql;
 end;
 end;
if EdETM.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

```

```

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := 'Mn0008';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByname('kb').AsString := 'Mn0008';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdETM.text);
dmMakanan.qproses.ExecSql;
end;
end;
if EdETT.text <> '0' Then
//Untuk mengurangi jumlah stok barang
begin

```

```

With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;

```

```

Parambyname('kb').AsString := 'Mn0009';
Open;
Stok := Fields[0].AsFloat;

//update data stok barang dengan cara membaca nilai Stok dari
//query diatasnya lalu kurangi dengan qty penjualan
//stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qproses.Close;
dmMakanan.qproses.SQL.Clear;
dmMakanan.qproses.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qproses.SQL.Add('Where KodeBrg =:kb');
dmMakanan.qproses.Prepare;
dmMakanan.qproses.ParamByName('kb').AsString := 'Mn0009';
dmMakanan.qproses.ParamByName('st').AsFloat := Stok-
StrToFloat(EdETT.text);
dmMakanan.qproses.ExecSql;
end;
end;
dmMakanan.TrMakanan.Commit;
end;

End.

```

- d. Pembuatan program form Login Customer  
unit ULogin1;

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls;
```

```
type
```

```
TForm1 = class(TForm)
```

```
Label1: TLabel;
```

```
Edit1: TEdit;
```

```
Label2: TLabel;
```

```
Edit2: TEdit;
```

```

 Button1: TButton;
 Button2: TButton;
 procedure Button1Click(Sender: TObject);
 procedure Button2Click(Sender: TObject);
 procedure FormShow(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 Form1: TForm1;

implementation

uses UCustomer;

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
begin
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmCustomer.show;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
Close;
end;

procedure TForm1.FormShow(Sender: TObject);

```

```
begin
Edit1.Clear;
Edit2.Clear;
end;
```

```
end.
```

e. Pembuatan program form Login Supplier

```
unit ULogin2;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls;
```

```
type
```

```
TForm2 = class(TForm)
```

```
Label1: TLabel;
```

```
Edit1: TEdit;
```

```
Label2: TLabel;
```

```
Edit2: TEdit;
```

```
Button1: TButton;
```

```
Button2: TButton;
```

```
procedure Button1Click(Sender: TObject);
```

```
procedure Button2Click(Sender: TObject);
```

```
procedure FormShow(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form2: TForm2;
```

```
implementation
```

```
Uses USupp;
```

```
{ $R *.dfm }
```

```
procedure TForm2.Button1Click(Sender: TObject);
begin
if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
begin
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmSupplier.show;
end;
```

```
procedure TForm2.Button2Click(Sender: TObject);
begin
Close;
end;
```

```
procedure TForm2.FormShow(Sender: TObject);
begin
Edit1.Clear;
Edit2.Clear;
end;
```

```
end.
```

- f. Pembuatan program form Login Input Makanan  
unit ULogin7;

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls;
```

```
type
```

```
TForm7 = class(TForm)
Label1: TLabel;
```

```

Label2: TLabel;
Edit1: TEdit;
Edit2: TEdit;
Button1: TButton;
Button2: TButton;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure FormShow(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 Form7: TForm7;

implementation

uses UInput;

{$R *.dfm}

procedure TForm7.Button1Click(Sender: TObject);
begin
if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
begin
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmInputMakanan.show;
end;

procedure TForm7.Button2Click(Sender: TObject);
begin
Close;

```

end;

```
procedure TForm7.FormShow(Sender: TObject);
```

```
begin
```

```
Edit1.Clear;
```

```
Edit2.Clear;
```

```
end;
```

```
end.
```

- g. Pembuatan program form Login Pembelian Barang  
unit ULogin3;

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls;
```

```
type
```

```
TForm3 = class(TForm)
```

```
Label1: TLabel;
```

```
Edit1: TEdit;
```

```
Label2: TLabel;
```

```
Edit2: TEdit;
```

```
Button1: TButton;
```

```
Button2: TButton;
```

```
procedure Button1Click(Sender: TObject);
```

```
procedure Button2Click(Sender: TObject);
```

```
procedure FormShow(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```
Form3: TForm3;
```



implementation

uses UBeliBarang;

{ \$R \*.dfm }

```
procedure TForm3.Button1Click(Sender: TObject);
begin
if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
begin
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmBeliBarang.show;
end;
```

```
procedure TForm3.Button2Click(Sender: TObject);
begin
Close;
end;
```

```
procedure TForm3.FormShow(Sender: TObject);
begin
Edit1.Clear;
Edit2.Clear;
end;
```

end.

- h. Pembuatan program form Login Laporan Penjualan Resto unit ULogin5;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls;

type

```
TForm6 = class(TForm)
 Label1: TLabel;
 Label2: TLabel;
 Edit1: TEdit;
 Edit2: TEdit;
 Button1: TButton;
 Button2: TButton;
 procedure FormShow(Sender: TObject);
 procedure Button1Click(Sender: TObject);
 procedure Button2Click(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;
```

var

```
Form6: TForm6;
```

implementation

uses ULaporanJual;

```
{ $R *.dfm }
```

```
procedure TForm6.FormShow(Sender: TObject);
begin
 Edit1.Clear;
 Edit2.clear;
end;
```

```
procedure TForm6.Button1Click(Sender: TObject);
begin
 if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
```

```

begin
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmLaPORANPenjualanResto.show;
end;

```

```

procedure TForm6.Button2Click(Sender: TObject);
begin
Close;
end;

```

end.

- i. Pembuatan program form Login Laporan Pembelian Barang  
unit ULogin4;

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls;

```

```

type

```

```

TForm4 = class(TForm)

```

```

Label1: TLabel;

```

```

Label2: TLabel;

```

```

Edit1: TEdit;

```

```

Edit2: TEdit;

```

```

Button1: TButton;

```

```

Button2: TButton;

```

```

procedure Button1Click(Sender: TObject);

```

```

procedure Button2Click(Sender: TObject);

```

```

procedure FormShow(Sender: TObject);

```

```

private

```

```

{ Private declarations }

```

```

public

```

```

 { Public declarations }
end;

var
 Form4: TForm4;

implementation

uses UBeliVarang, ULaporanBeli;

{$R *.dfm}

procedure TForm4.Button1Click(Sender: TObject);
begin
 if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then
 begin
 MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
 Exit;
 end;
 Close;
 FrmLaporanBeliBarang.show;
end;

procedure TForm4.Button2Click(Sender: TObject);
begin
 Close;
end;

procedure TForm4.FormShow(Sender: TObject);
begin
 Edit1.Clear;
 Edit2.clear;
end;

end.
j. Pembuatan program form Login Penjualan Resto
 unit ULogin6;

```

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls;

type

TForm5 = class(TForm)

Label1: TLabel;

Label2: TLabel;

Edit1: TEdit;

Edit2: TEdit;

Button1: TButton;

Button2: TButton;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure FormShow(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form5: TForm5;

implementation

uses UJual;

{ \$R \*.dfm }

procedure TForm5.Button1Click(Sender: TObject);

begin

if (Edit1.Text <> 'Admin') or (Edit2.Text <> '1234') then

begin

```
MessageDlg('User Name atau Password salah', MTWarning, [MBOK], 0);
Exit;
end;
Close;
FrmJual.show;
end;
```

```
procedure TForm5.Button2Click(Sender: TObject);
begin
Close;
end;
```

```
procedure TForm5.FormShow(Sender: TObject);
begin
Edit1.Clear;
Edit2.Clear;
end;
```

end.

- k. Pembuatan program form Input Makanan  
unit UInput;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, Grids, DBGrids, RpCon, RpConDS, RpDefine, RpRave, DB,  
IBCustomDataSet, IBQuery, StdCtrls;

type

TfrmInputMakanan = class(TForm)

Label1: TLabel;  
EdKodeBrg: TEdit;  
Label2: TLabel;  
EdNamaBrg: TEdit;  
Label3: TLabel;  
EdHrgBeli: TEdit;

```

Label4: TLabel;
EdHrgJual: TEdit;
Label5: TLabel;
EdKategori: TEdit;
BtSimpan: TButton;
BtCari: TButton;
BtKeluar: TButton;
BtHapus: TButton;
QBarang: TIBQuery;
RvInput: TRvProject;
RvfInput: TRvDataSetConnection;
DBGrid1: TDBGrid;
Label6: TLabel;
EdStok: TEdit;
BtCetak: TButton;
QBarangKODEBRG: TIBStringField;
QBarangNAMABRG: TIBStringField;
QBarangHRGBELI: TIBBCDField;
QBarangHRGJUAL: TIBBCDField;
QBarangSTOK: TIBBCDField;
QBarangKATEGORI: TIBStringField;
Label7: TLabel;
procedure BtSimpanClick(Sender: TObject);
procedure BtCariClick(Sender: TObject);
procedure BtKeluarClick(Sender: TObject);
procedure BtHapusClick(Sender: TObject);
procedure BtCetakClick(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmInputMakanan: TFrmInputMakanan;

implementation

```

```

Uses UModule;
{$R *.dfm}

procedure TFrmInputMakanan.BtSimpanClick(Sender: TObject);
begin
 With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Insert Into BARANG(KODEBRG, NAMABRG, HRGBELI,
HRGJUAL, STOK, KATEGORI) '+
 'Values(:kb, :nb, :hb, :hj, :st, :kt)');
 Prepare;
 ParamByName('kb').AsString := EdKodeBrg.Text;
 ParamByName('nb').AsString := EdNamaBrg.Text;
 ParamByName('hb').AsString := EdHrgBeli.Text;
 ParamByName('hj').AsString := EdHrgJual.Text;
 ParamByName('st').AsString := EdStok.Text;
 ParamByName('kt').AsString := EdKategori.Text;
 ExecSql;
 End;

 dmMakanan.TrMakanan.Commit;

 EdKodeBrg.Clear;
 EdNamaBrg.Clear;
 EdHrgBeli.Clear;
 EdHrgJual.Clear;
 EdStok.Clear;
 EdKategori.Clear;

 With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODEBRG, NAMABRG, HRGBELI, HRGJUAL,
STOK, KATEGORI From BARANG');

```



```
 Open;
 End;
end;
```

```
procedure TFrmInputMakanan.BtCariClick(Sender: TObject);
begin
 With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODEBRG, NAMABRG, HRGBELI, HRGJUAL,
STOK, KATEGORI From BARANG');
 Sql.Add('Where KODEBRG Like :kb');
 Prepare;
 ParamByName('kb').AsString := '%'+EdKodeBrg.Text+'%';
 Open;
 End;
 end;
```

```
procedure TFrmInputMakanan.BtKeluarClick(Sender: TObject);
begin
 Close;
end;
```

```
procedure TFrmInputMakanan.BtHapusClick(Sender: TObject);
begin
 With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Delete From BARANG');
 Sql.Add('Where KODEBRG =:kb');
 Prepare;
 ParamByName('kb').AsString := Trim(DBGrid1.Fields[0].AsString);
 ExecSql;
 End;
```

```

dmMakanan.TrMakanan.Commit;

With dmMakanan.QProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select KODEBRG, NAMABRG, HRGBELI, HRGJUAL,
STOK, KATEGORI From BARANG');
Open;
End;
end;

```

```

procedure TFrmInputMakanan.BtCetakClick(Sender: TObject);
begin
with QBarang Do
Begin
Close;
Prepare;
ParamByName('kb').AsString := Trim(EdKodeBrg.Text);
Open;
End;
RvInput.Execute;
end;

```

end.

1. Pembuatan program form Customer  
unit UCustomer;

```
interface
```

```
uses
```

```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, RpRave, RpDefine, RpCon, RpConDS, DB,
IBCustomDataSet,
IBQuery, Grids, DBGrids;

```

```
type
```

```

TFrmCustomer = class(TForm)
 Label1: TLabel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 EdKodeCust: TEdit;
 EdNamaCust: TEdit;
 EdAlmCust: TEdit;
 EdTlp: TEdit;
 EdHP: TEdit;
 DBGrid1: TDBGrid;
 BtSimpan: TButton;
 BtHapus: TButton;
 BtCari: TButton;
 BtCetak: TButton;
 BtKeluar: TButton;
 QCustomer: TIBQuery;
 RvCust: TRvDataSetConnection;
 RvProject1: TRvProject;
 QCustomerALAMATCUST: TIBStringField;
 QCustomerHP: TIBStringField;
 QCustomerKODECUST: TIBStringField;
 QCustomerNAMACUST: TIBStringField;
 QCustomerTLP: TIBStringField;
 procedure BtSimpanClick(Sender: TObject);
 procedure BtCariClick(Sender: TObject);
 procedure BtKeluarClick(Sender: TObject);
 procedure BtHapusClick(Sender: TObject);
 procedure BtCetakClick(Sender: TObject);
 procedure EdNamaCustChange(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

```

```

var
 FrmCustomer: TFrmCustomer;

implementation

uses UModule;

{$R *.dfm}

procedure TFrmCustomer.BtSimpanClick(Sender: TObject);
begin
 With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Insert Into
CUSTOMER(KODECUST,NAMACUST,ALAMATCUST,TLP,HP) '+
'Values(:kc, :nc, :ac, :tp, :hp)');
 Prepare;
 ParamByName('kc').AsString := EdKodeCust.Text;
 ParamByName('nc').AsString := EdNamaCust.Text;
 ParamByName('ac').AsString := EdAlmCust.Text;
 ParamByName('tp').AsString := EdTlp.Text;
 ParamByName('hp').AsString := EdHP.Text;
 ExecSql;
 End;

 dmMakanan.TrMakanan.Commit;

 EdKodeCust.Clear;
 EdNamaCust.Clear;
 EdAlmCust.Clear;
 EdTlp.Text:='0';
 EdHP.Text:='0';

 With dmMakanan.QProses Do

```

```

 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODECUST,NAMACUST,ALAMATCUST,TLP,HP
From CUSTOMER');
 Open;
 End;
end;

```

```

procedure TFrmCustomer.BtCariClick(Sender: TObject);

```

```

begin
 With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODECUST,NAMACUST,ALAMATCUST,TLP,HP
From CUSTOMER');
 Sql.Add('Where KODECUST Like :kc');
 Prepare;
 ParamByName('kc').AsString := '%'+EdKodeCust.Text+'%';
 Open;
 End;
end;

```

```

procedure TFrmCustomer.BtKeluarClick(Sender: TObject);

```

```

begin
 Close;
end;

```

```

procedure TFrmCustomer.BtHapusClick(Sender: TObject);

```

```

begin
 With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Delete From CUSTOMER');
 Sql.Add('Where KODECUST =:kc');

```

```

Prepare;
 ParamByName('kc').AsString := Trim(DBGrid1.Fields[0].AsString);
ExecSql;
End;

dmMakanan.TrMakanan.Commit;

With dmMakanan.QProses Do
Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODECUST,NAMACUST,ALAMATCUST,TLP,HP
From CUSTOMER');
 Open;
End;

end;

procedure TFrmCustomer.BtCetakClick(Sender: TObject);
begin
rvProject1.Execute;
end;

procedure TFrmCustomer.EdNamaCustChange(Sender: TObject);
begin
With dmMakanan.qProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select * From Customer');
 Sql.Add('Where NAMACUST Like :nc');
 Prepare;
 ParamByName('nc').AsString := '%'+EdNamaCust.Text+'%';
 Open;
End;
end;

```

```

end.
m. Pembuatan program form Supplier
 unit USupp;

 interface

 uses
 Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
 Dialogs, Grids, DBGrids, StdCtrls, RpRave, RpDefine, RpCon, RpConDS,
 DB,
 IBCustomDataSet, IBQuery;

 type
 TFrmSupplier = class(TForm)
 Label1: TLabel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 EdKodeSupp: TEdit;
 EdNamaSupp: TEdit;
 EdAlmSupp: TEdit;
 EdTlp: TEdit;
 EdHP: TEdit;
 DBGrid1: TDBGrid;
 BtSimpan: TButton;
 BtCari: TButton;
 BtHapus: TButton;
 BtCetak: TButton;
 BtKeluar: TButton;
 QSupplier: TIBQuery;
 RvfSupp: TRvDataSetConnection;
 RvSupp: TRvProject;
 QSupplierKODESUPP: TIBStringField;
 QSupplierNAMASUPP: TIBStringField;
 QSupplierALAMATSUPP: TIBStringField;
 end;
 end;

```

```

QSupplierTLP: TIBStringField;
QSupplierHP: TIBStringField;
procedure BtSimpanClick(Sender: TObject);
procedure BtKeluarClick(Sender: TObject);
procedure BtCariClick(Sender: TObject);
procedure BtHapusClick(Sender: TObject);
procedure BtCetakClick(Sender: TObject);
procedure EdNamaSuppChange(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmSupplier: TFrmSupplier;

implementation
Uses UModule;
{$R *.dfm}

procedure TFrmSupplier.BtSimpanClick(Sender: TObject);
begin
 With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Insert Into
SUPPLIER(KODESUPP,NAMASUPP,ALAMATSUPP,TLP,HP) '+
'Values(:ks, :ns, :as, :tp, :hp)');
 Prepare;
 ParamByName('ks').AsString := EdKodeSupp.Text;
 ParamByName('ns').AsString := EdNamaSupp.Text;
 ParamByName('as').AsString := EdAlmSupp.Text;
 ParamByName('tp').AsString := EdTlp.Text;
 ParamByName('hp').AsString := EdHP.Text;
 ExecSql;

```



```

End;

dmMakanan.TrMakanan.Commit;

EdKodeSupp.Clear;
EdNamaSupp.Clear;
EdAlmSupp.Clear;
EdTlp.Text:='0';
EdHp.Text:='0';

With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODESUPP,NAMASUPP,ALAMATSUPP,TLP,HP
From SUPPLIER');
 Open;
 End;
end;

procedure TFrmSupplier.BtKeluarClick(Sender: TObject);
begin
Close;
end;

procedure TFrmSupplier.BtCariClick(Sender: TObject);
begin
With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODESUPP,NAMASUPP,ALAMATSUPP,TLP,HP
From SUPPLIER');
 Sql.Add('Where KODESUPP Like :kp');
 Prepare;
 ParamByName('kp').AsString := '%'+EdKodeSupp.Text+'%';
 Open;

```

```

 End;
end;

procedure TFrmSupplier.BtHapusClick(Sender: TObject);
begin
With dmMakanan.QData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Delete From SUPPLIER');
 Sql.Add('Where KODESUPP =:kp');
 Prepare;
 ParamByName('kp').AsString := Trim(DBGrid1.Fields[0].AsString);
 ExecSql;
 End;

dmMakanan.TrMakanan.Commit;

With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODESUPP,NAMASUPP,ALAMATSUPP,TLP,HP
From SUPPLIER');
 Open;
 End;
end;

procedure TFrmSupplier.BtCetakClick(Sender: TObject);
begin
RvSupp.Execute;
end;

procedure TFrmSupplier.EdNamaSuppChange(Sender: TObject);
begin
With dmMakanan.qProses Do
 Begin

```

```

 Close;
 Sql.Clear;
 Sql.Add('Select * From SUPPLIER');
 Sql.Add('Where NAMASUPP Like :ns');
 Prepare;
 ParamByName('ns').AsString := '%'+EdNamaSupp.Text+'%';
 Open;
 End;
end;

```

end.

- n. Pembuatan program form Pembelian Barang  
unit UBeliBarang;

```
interface
```

```
uses
```

```

 Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
 Dialogs, StdCtrls, Grids, DBGrids, RpRave, RpDefine, RpCon, RpConDS,
 DB,
 IBCustomDataSet, IBQuery;

```

```
type
```

```

TFrmBeliBarang = class(TForm)
 Label1: TLabel;
 Label3: TLabel;
 EdTgl: TEdit;
 EdBon: TEdit;
 Label4: TLabel;
 Label5: TLabel;
 Label6: TLabel;
 EdKodeSupp: TEdit;
 EdAlamat: TEdit;
 CmbSupp: TComboBox;
 Label2: TLabel;
 Label7: TLabel;
 EdQty: TEdit;

```

Label8: TLabel;  
EdHrgBeli: TEdit;  
CmbNamaBrg: TComboBox;  
DBGrid1: TDBGrid;  
BtSimpan: TButton;  
Label9: TLabel;  
EdKodeBrg: TEdit;  
BtTambah: TButton;  
BtKeluar: TButton;  
BtCetak: TButton;  
QFBeli: TIBQuery;  
RvfBeli: TRvDataSetConnection;  
RvfBeliItem: TRvDataSetConnection;  
RvBeli: TRvProject;  
QFBeliItem: TIBQuery;  
Label10: TLabel;  
EdJumlah: TEdit;  
QFBeliBONBELI: TIBStringField;  
QFBeliTGL: TDateField;  
QFBeliKODESUPP: TIBStringField;  
QFBeliNAMASUPP: TIBStringField;  
QFBeliALAMATSUPP: TIBStringField;  
QFBeliItemBONBELI: TIBStringField;  
QFBeliItemKODEBRG: TIBStringField;  
QFBeliItemNAMABRG: TIBStringField;  
QFBeliItemQTY: TIBBCDField;  
QFBeliItemHRGBELI: TIBBCDField;  
QFBeliItemJUMLAH: TIBBCDField;  
procedure CmbSuppChange(Sender: TObject);  
procedure CmbNamaBrgChange(Sender: TObject);  
procedure BtTambahClick(Sender: TObject);  
procedure BtKeluarClick(Sender: TObject);  
procedure BtCetakClick(Sender: TObject);  
procedure FormActivate(Sender: TObject);  
procedure TampilkanSemua;  
procedure JumlahTotal;  
procedure TampilkanItem;

```

 procedure FormShow(Sender: TObject);
 procedure BtSimpanClick(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmBeliBarang: TFrmBeliBarang;

implementation
Uses UModule;
{$R *.dfm}

procedure TFrmBeliBarang.CmbSuppChange(Sender: TObject);
begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODESUPP,ALAMATSUPP From SUPPLIER');
 Sql.Add('Where NAMASUPP =:NS');
 Prepare;
 ParamByName('NS').AsString := CmbSupp.Text;
 Open;
 EdKodeSupp.Text := Trim(Fields[0].AsString);
 EdAlamat.Text := Trim(Fields[1].AsString);
 End;

If CmbSupp.Text <> " Then
 Begin
 With dmMakanan.qProses DO
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Update BELI Set KODESUPP =:ks');

```

```

 Sql.Add('Where BONBELI =:bb');
 Prepare;
 ParamByName('bb').AsString := Trim(EdBon.Text);
 ParamByName('ks').AsString := Trim(EdKodeSupp.Text);
 ExecSql;
 End;
 dmMakanan.TrMakanan.Commit
End
Else
 Begin
 ShowMessage('Maaf Supplier Harus Di Isi ');
 CmbSupp.SetFocus;
 end;
 TampilkanSemua;
end;

procedure TFrmBeliBarang.CmbNamaBrgChange(Sender: TObject);
begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select KODEBRG, HRGBELI from BARANG');
 Sql.Add('Where NAMABRG =:nb');
 Prepare;
 ParamByName('nb').AsString := Trim(CmbNamaBrg.Text);
 Open;
 EdKodeBrg.Text := Trim(Fields[0].Value);
 EdHrgBeli.Text := Trim(FloatToStr(Fields[1].Value));
 EdQty.Text := Trim('1');
 End;
end;

procedure TFrmBeliBarang.BtTambahClick(Sender: TObject);
begin
 //Tampilkan BonJual (No Bon Jual)
 With dmMakanan.qData Do

```

```

Begin
Close;
Sql.Clear;
Sql.Add('Select BONBELI From BELI');
Open;

//Jika Ternyata tidak ada (kosong/null) maka
If Fields[0].IsNull Then
 Begin
 EdBon.Text := '00001';
 End
Else
 //Jika ada nilainya maka, tentukan nobon paling akhir
 Begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Max(BONBELI) From BELI');
 Open;
 //Ini adalah format agar diawali dengan 0000
 EdBon.Text := FormatFloat('00000',StrToFloat(Fields[0].Value+1));
 End;
 End;
 End;
End;

//Simpan nobon tersebut kedalam tabel jual ke field Bonjual
With dmMakanan.qProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Insert Into BELI(BONBELI,TGL) Values(:bb,:tg)');
 Prepare;
 ParamByName('bb').AsString := EdBon.Text;
 ParamByName('tg').AsDate := StrToDate(EdTgl.Text);
 ExecSql;
 end;

```

```

dmMakanan.TrMakanan.Commit;

CmbSupp.Text := Trim("");
EdKodeSupp.Clear;
EdAlamat.Clear;

TampilkanSemua;
end;

procedure TFrmBeliBarang.BtKeluarClick(Sender: TObject);
begin
Close;
end;

procedure TFrmBeliBarang.BtCetakClick(Sender: TObject);
begin
with QFBeli Do
Begin
Close;
Prepare;
ParamByName('bb').AsString := Trim(EdBon.Text);
Open;
End;

with QFBeliItem Do
Begin
Close;
Prepare;
ParamByName('bb').AsString := Trim(EdBon.Text);
Open;
End;

RvBeli.Execute;
end;

procedure TFrmBeliBarang.FormActivate(Sender: TObject);
begin

```



```

With dmMakanan.QData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select KODESUPP From BELI');
Sql.Add('Where BONBELI =:BB');
Prepare;
ParamByName('BB').AsString := Trim(EdBon.Text);
Open;
EdKodeSupp.Text := Fields[0].AsString;
End;

//Panggil atau jalankan procedure TampilkanItem
TampilkanSemua;
end;

procedure TFrmBeliBarang.TampilkanSemua;
begin
TampilkanItem;
JumlahTotal;
end;
procedure TFrmBeliBarang.TampilkanItem;
begin
//Menampilkan semua data penjualan
With dmMakanan.qProses Do
Begin
Close;
Sql.Clear;
sql.Add('Select
BELIITEM.KODEBRG, BARANG.NAMABRG, BELIITEM.QTY, '+
'BELIITEM.HRGBELI,
(BELIITEM.HRGBELI*BELIITEM.QTY) JUMLAH From BELIITEM');
sql.Add('Inner Join BARANG On
BARANG.KODEBRG=BELIITEM.KODEBRG');
Sql.Add('Where BELIITEM.BONBELI=:BB');
Prepare;
ParamByName('BB').AsString := Trim(EdBon.Text);

```

```

 Open;
 End;
end;

procedure TFrmBeliBarang.JumlahTotal;
begin
 //Procedure untuk menghitung jumlah total penjualan per bon
 // dan ditampilkan di edJumlahTotal
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Sum(QTY*HRGBELI) As JUMLAHTOTAL From
BELIITEM');
 Sql.Add('Where BELIITEM.BONBELI =:BB');
 Prepare;
 ParamByName('BB').AsString := Trim(EdBon.Text);
 Open;
 EdJumlah.Text := FloattoStr(Fields[0].AsFloat);
 End;
end;

procedure TFrmBeliBarang.FormShow(Sender: TObject);
begin
 //Tampilkan tanggal di edTgl
 EdTgl.Text := DateToStr(Date);

 //Tarik Nama customer ke cmbNamaCust
 CmbSupp.Clear;
 With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Distinct NAMASUPP From SUPPLIER');
 Sql.Add('Order by NAMASUPP');
 Open;
 while not Eof do

```

```

begin
 CmbSupp.Items.Add(Trim(Fields[0].AsString));
 Next;
end;
end;

//Tarik Nama Barang ke cmbNamaBrg
CmbNamaBrg.Clear;
With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Distinct NAMABRG From BARANG');
 Sql.Add('Order by NAMABRG');
 Open;
 while not Eof do
 begin
 CmbNamaBrg.Items.Add(Trim(Fields[0].AsString));
 Next;
 end;
 end;
 TampilkanSemua;
end;

procedure TFrmBeliBarang.BtSimpanClick(Sender: TObject);
var Stok : Double;
begin
 With dmMakanan.qData Do
 Begin
 Close;
 sql.Clear;
 // Sql.Add('Insert Into
JUALMOBIL(BONJUAL,NOPOL,QTY,HRGJUAL) '+
 // 'Values(:BJ, :NP, :QY, :HJ)');
 Sql.Add('Insert Into
BELIITEM(BONBELI,KODEBRG,QTY,HRGBELI)'+
 'Values(:BB, :KB, :QY, :HB)');

```

```

Prepare;
ParamByName('BB').AsString := Trim(EdBon.Text);
ParamByName('KB').AsString := Trim(EdKodeBrg.Text);
ParamByName('QY').AsFloat := StrToFloat(EdQty.Text);
ParamByName('HB').AsFloat := StrToFloat(EdHrgBeli.Text);
ExecSql;
End;
 //Untuk menambah jumlah stok barang
With dmMakanan.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Stok from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
Parambyname('kb').AsString := Trim(edKodeBrg.Text);
Open;
Stok := Fields[0].AsFloat;

 //update data stok barang dengan cara membaca nilai Stok dari
 //query diatasnya lalu tambahkan dengan qty pembelian
 //stok yang di update harus sesuai dengan kode barang nya.
dmMakanan.qData.Close;
dmMakanan.qData.SQL.Clear;
dmMakanan.qData.SQL.Add('Update Barang Set Stok=:st');
dmMakanan.qData.SQL.Add('Where KodeBrg =:kb');
Prepare;
ParamByname('kb').AsString := Trim(edKodeBrg.Text);
ParamByName('st').AsFloat := Stok+StrToFloat(EdQty.text);
ExecSql;
end;
 dmMakanan.TrMakanan.Commit;
 //TampilkanItem;
EdKodeBrg.Clear;
edHrgBeli.Text := Trim("");
edQty.Clear;
CmbNamaBrg.Text := Trim("");

```

```
 cmbNamaBrg.SetFocus;
 TampilkanSemua;
end;
```

```
end.
```

- o. Pembuatan program form Penjualan Resto  
unit UJual;

```
interface
```

```
uses
```

```
 Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
 Dialogs, Grids, DBGrids, StdCtrls, RpRave, RpDefine, RpCon, RpConDS,
 DB,
 IBCustomDataSet, IBQuery;
```

```
type
```

```
TfrmJual = class(TForm)
 Label1: TLabel;
 EdTgl: TEdit;
 Label2: TLabel;
 EdBon: TEdit;
 DBGrid1: TDBGrid;
 BtCetak: TButton;
 Btkeluar: TButton;
 QJual: TIBQuery;
 RvfJual: TRvDataSetConnection;
 RvJual: TRvProject;
 RvfJualItem: TRvDataSetConnection;
 QJualItem: TIBQuery;
 edJumlahTotal: TEdit;
 QJualBONJUAL: TIBStringField;
 QJualTGL: TDateField;
 QJualKODECUST: TIBStringField;
 QJualNAMACUST: TIBStringField;
 QJualALAMATCUST: TIBStringField;
 QJualItemBONJUAL: TIBStringField;
```

```

QJualItemKODEBRG: TIBStringField;
QJualItemNAMABRG: TIBStringField;
QJualItemQTY: TIBBCDField;
QJualItemHRGJUAL: TIBBCDField;
QJualItemJUMLAH: TIBBCDField;
EdBon2: TEdit;
procedure BtkeluarClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure TampilkanItem;
procedure TampilkanSemua;
procedure JumlahTotal;
procedure BtCetakClick(Sender: TObject);
procedure EdBon2Change(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmJual: TFrmJual;

implementation
 Uses UModule;
 {$R *.dfm}

 procedure TFrmJual.BtkeluarClick(Sender: TObject);
 begin
 Close;
 end;

 procedure TFrmJual.FormShow(Sender: TObject);
 var a : String;
 var Stok : Double;
 begin
 //Ambil nilai bon akhir lalu tampilkan di EdBon
 With dmMakanan.qData Do

```

```

Begin
Close;
Sql.Clear;
Sql.Add('Select Max(BONJUAL) From JUAL');
Open;
EdBon.Text := FormatFloat('00000',StrToFloat(Fields[0].Value));
End;
//Tampilkan tanggal di EdTgl
EdTgl.Text := DateToStr(Date);

//no bon dikurangi satu
//EdBon2.Text := '000'InttoStr(StrToInt(EdBon.Text)-1);
EdBon2.Text:= FormatFloat('00000',StrToFloat(EdBon.text)-1);
 end;
procedure TFrmJual.TampilkanSemua;
begin
 TampilkanItem;
 JumlahTotal;
end;

procedure TFrmJual.TampilkanItem;
begin
 With dmMakanan.qProses Do
 Begin
 Close;
 Sql.Clear;
 sql.Add('Select
JUALITEM.KODEBRG,BARANG.NAMABRG,JUALITEM.QTY, '+
 'JUALITEM.HRGJUAL,
(JUALITEM.HRGJUAL*JUALITEM.QTY) As Jumlah From JUALITEM');
 sql.Add('Inner Join BARANG On
BARANG.KODEBRG=JUALITEM.KODEBRG');
 Sql.Add('Where JUALITEM.BONJUAL=:BJ');
 Prepare;
 ParamByName('BJ').AsString := Trim(EdBon2.Text);
 Open;
 End;

```

```

end;

procedure TFrmJual.JumlahTotal;
begin
 With dmMakanan.qData Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Sum(QTY*HRGJUAL) As JumlahTotal From
JUALITEM');
 Sql.Add('Where JUALITEM.BONJUAL =:BJ');
 Prepare;
 ParamByName('BJ').AsString := Trim(EdBon2.Text);
 Open;
 edJumlahTotal.Text := FloattoStr(Fields[0].AsFloat);
 End;
end;

procedure TFrmJual.BtCetakClick(Sender: TObject);
begin
 with QJual Do
 Begin
 Close;
 Prepare;
 ParamByName('BJ').AsString := Trim(EdBon2.text);
 Open;
 End;
 with QJualItem Do
 Begin
 Close;
 Prepare;
 ParamByName('BJ').AsString := Trim(EdBon2.text);
 Open;
 End;
 RvJual.Execute;

end;

```



```
procedure TFrmJual.EdBon2Change(Sender: TObject);
begin
 TampilkanItem;
 JumlahTotal;
end;
```

end.

- p. Pembuatan program form Laporan Pembelian Barang  
unit ULaporanBeli;

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, ComCtrls, RpRave, RpDefine, RpCon, RpConDS, DB,
IBCustomDataSet, IBQuery;
```

```
type
```

```
TFrmLaporanBeliBarang = class(TForm)
 GroupBox1: TGroupBox;
 Label1: TLabel;
 Label2: TLabel;
 Label3: TLabel;
 Label4: TLabel;
 dtpTglA: TDateTimePicker;
 dtpTglB: TDateTimePicker;
 cmbNamaSupp: TComboBox;
 edBon: TEdit;
 btCetak: TButton;
 btKeluar: TButton;
 QLBeli: TIBQuery;
 RvfLaporanBeliBarang: TRvDataSetConnection;
 RvPembelian: TRvProject;
 QLBeliBONBELI: TIBStringField;
 QLBeliTGL: TDateField;
 QLBeliKODESUPP: TIBStringField;
```

```

QLBeliNAMASUPP: TIBStringField;
QLBeliALAMATSUPP: TIBStringField;
QLBeliKODEBRG: TIBStringField;
QLBeliNAMABRG: TIBStringField;
QLBeliQTY: TIBBCDField;
QLBeliHRGBELI: TIBBCDField;
QLBeliJUMLAH: TIBBCDField;
procedure btCetakClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure btKeluarClick(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;

var
 FrmLaporanBeliBarang: TFrmLaporanBeliBarang;

implementation
 Uses UModule;
 {$R *.dfm}

procedure TFrmLaporanBeliBarang.btCetakClick(Sender: TObject);
 var Path : String;
begin
 //Menentukan directory
 Path := 'D:\my document\TA (Tugas Akhir)_Skripsi\program
Skripsi\laporan\';
 ShowMessage('Laporan Pembelian');

 With QLBeli Do
 Begin
 Close;
 Prepare;
 ParambyName('tglA').AsDate := dtpTglA.Date;
 ParambyName('tglB').AsDate := dtpTglB.Date;

```

```

 ParamByName('ns').AsString := Trim('%'+cmbNamaSupp.Text+'%');
 ParamByName('bb').AsString := Trim('%'+edBon.Text+'%');
 Open;
 End;
 //rvLaporan.ProjectFile :=
'D:\Wong\Maranatha\Database\Toko\Laporan\LJual.rav'
 RvPembelian.ProjectFile := Path+ 'LBeliVarang.rav';
 RvPembelian.execute;
end;

```

```

procedure TFrmLaporanBeliBarang.FormShow(Sender: TObject);
begin
//Tarik Nama supplier ke cmbNamaSupp
cmbNamaSupp.Clear;
With dmMakanan.QProses Do
 Begin
 Close;
 Sql.Clear;
 Sql.Add('Select Distinct NAMASUPP From SUPPLIER');
 Sql.Add('Order by NAMASUPP');
 Open;
 while not Eof do
 begin
 cmbNamaSupp.Items.Add(Trim(Fields[0].AsString));
 Next;
 end;
 end;
end;

```

```

procedure TFrmLaporanBeliBarang.btKeluarClick(Sender: TObject);
begin
Close;
end;

```

end.

- q. Pembuatan program form Laporan Penjualan Resto  
unit ULaporanJual;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, DB, IBCustomDataSet, IBQuery, RpRave, RpDefine, RpCon,  
RpConDS,  
StdCtrls, ComCtrls, Grids, DBGrids;

type

```
TFrmLaporanPenjualanResto = class(TForm)
 dtpTglA: TDateTimePicker;
 dtpTglB: TDateTimePicker;
 BtCetak: TButton;
 BtKeluar: TButton;
 RvfPenjualanResto: TRvDataSetConnection;
 RvPenjualan: TRvProject;
 QLaporJual: TIBQuery;
 Label1: TLabel;
 EdBon: TEdit;
 QLaporJualBONJUAL: TIBStringField;
 QLaporJualTGL: TDateField;
 QLaporJualKODEBRG: TIBStringField;
 QLaporJualNAMABRG: TIBStringField;
 QLaporJualQTY: TIBBCDField;
 QLaporJualHRGJUAL: TIBBCDField;
 QLaporJualJUMLAH: TIBBCDField;
 Label2: TLabel;
 Label3: TLabel;
 procedure BtKeluarClick(Sender: TObject);
 procedure BtCetakClick(Sender: TObject);
private
 { Private declarations }
public
 { Public declarations }
end;
```

```

var
 FrmLaporanPenjualanResto: TFrmLaporanPenjualanResto;

implementation
 Uses UModule;
 {$R *.dfm}

 procedure TFrmLaporanPenjualanResto.BtKeluarClick(Sender: TObject);
 begin
 Close;
 end;

 procedure TFrmLaporanPenjualanResto.BtCetakClick(Sender: TObject);
 var Path : String;
 begin
 //Menentukan directory
 Path := 'D:\my document\TA (Tugas Akhir)_Skripsi\program
 Skripsi\laporan\';
 // ShowMessage('Laporan Penjualan');

 With QLaporJual Do
 Begin
 Close;
 Prepare;
 ParamByName('tglA').AsDate := dtpTglA.Date;
 ParamByName('tglB').AsDate := dtpTglB.Date;
 ParamByName('bj').AsString := Trim('%'+EdBon.Text+'%');
 Open;
 End;
 RvPenjualan.ProjectFile := Path+ 'LJualResto.rav';
 RvPenjualan.execute;
 end;

 end.

```