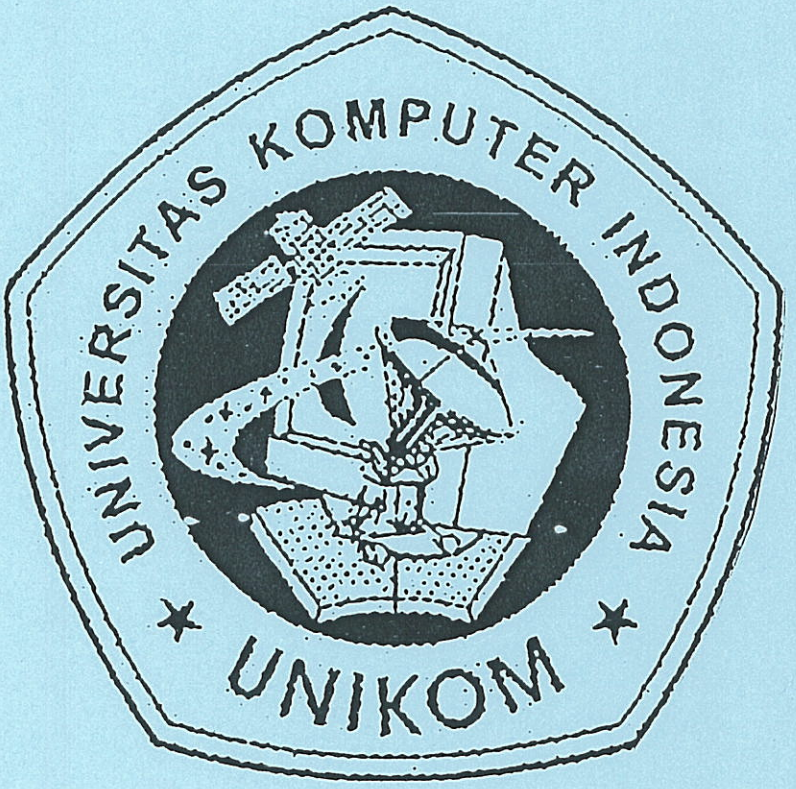
# *Daftar Pustaka*

Andrew S. Tanebaum 1990 *Jaringan Komputer*. Prenhallindo

Antony Pranata 2001 *Pemrograman Borland Delphi Edisi3*, ANDI Yogyakarta

Yogiyanto HM. 1993 *Analisis & Design Sistem Informasi Pendekatan Terstruktur*, ANDI Yogyakarta


UNIVERSITAS KOMPUTER INDONESIA
UNIKOM

# LAMPIRAN I

```
unit PSW;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, Buttons, ExtCtrls, DB, DBTables;

type
  TFormPsw = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    ScrollBox1: TScrollBox;
    Label1: TLabel;
    Label2: TLabel;
    Nama: TEdit;
    Password: TEdit;
    PInfo: TPanel;
    QDel: TQuery;
    btnBatal: TSpeedButton;
    Button1: TSpeedButton;
    sbHelp: TSpeedButton;
    QHapus: TQuery;
    Timer1: TTimer;
    procedure Button1Click(Sender: TObject);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure btnBatalClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure FormCreate(Sender: TObject);
    procedure sbHelpClick(Sender: TObject);
    procedure PasswordExit(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
  private
    idx  : integer;
    OK   : Boolean;
    Fasilitas,Jenis : string;
    function CekPassWord:Boolean;
    procedure Aktif_Fasilitas;
  public
     Pemakai : string;
  end;
var
  FormPsw: TFormPsw;

implementation

uses Main, UDM, Lap;

{$R *.DFM}

procedure TFormPsw.Aktif_Fasilitas;
begin
  with fmMain do
  begin
    Menu1.Visible:=(Pos(Chr(65),Fasilitas)<>0);
    Menu11.Visible:=(Pos(Chr(66),Fasilitas)<>0);
    Menu12.Visible:=(Pos(Chr(67),Fasilitas)<>0);
    Menu121.Visible:=(Pos(Chr(68),Fasilitas)<>0);
    Menu122.Visible:=(Pos(Chr(69),Fasilitas)<>0);
    Menu2.Visible:=(Pos(Chr(70),Fasilitas)<>0);
    Menu21.Visible:=(Pos(Chr(71),Fasilitas)<>0);
```

```
    Menu22.Visible:=(Pos(Chr(72),Fasilitas)<>0);
    Menu23.Visible:=(Pos(Chr(73),Fasilitas)<>0);
    Menu3.Visible:=(Pos(Chr(74),Fasilitas)<>0);
    Menu31.Visible:=(Pos(Chr(75),Fasilitas)<>0);
    Menu32.Visible:=(Pos(Chr(76),Fasilitas)<>0);
    Menu33.Visible:=(Pos(Chr(77),Fasilitas)<>0);
    Menu34.Visible:=(Pos(Chr(78),Fasilitas)<>0);
    Menu4.Visible:=(Pos(Chr(79),Fasilitas)<>0);
    Menu5.Visible:=(Pos(Chr(88),Fasilitas)<>0);
  end;
end;

function TFormPsw.CekPassWord:Boolean;
var
  i,t1  : byte;
  s,s1  : string;
  t2    :real;
begin
  with DM.TPassword do
  begin
    if FindKey([Nama.Text]) then
    begin
      s1:='';
      s:=FieldByName('Pass').AsString;
      for i:=1 to length(s) do
      begin
        t1:=ord(s[i]);
        t2:=t1-254;
        s1:=s1+chr(round(t2));
      end;
      if (Nama.Text<>'')and (PassWord.Text=s1) then
      begin
        Fasilitas:=FieldByName('Menu').AsString;
        Result:=True;
      end else
        Result:=False;
    end else
      Result:=False;
  end;
end;

procedure TFormPsw.Button1Click(Sender: TObject);
begin
 if DM.TPassword.RecordCount>0 then
 begin
   OK:=CekPassWord;
   Aktif_Fasilitas;
 end
 else
   begin
    if ((Nama.Text=User) and (Password.Text=PswX)) and
      ((Nama.Text<>'')and(Password.Text<>'')) then
    begin
      OK:=True;
      Jenis:='S';
    end
    else
      OK:=False;
   end;

 if OK then
 begin
   FormPsw.Hide;
   Pemakai:=Nama.Text;
   fmMain.ShowModal;
 end else
   MessageDlg('Identifikasi tidak dikenal ...',mtinformation,[mbOK],0);
end;

procedure TFormPsw.FormKeyPress(Sender: TObject; var Key: Char);
```

```
begin
  if Key=#13 then
  begin
    Key:=#0;
    SelectNext(ActiveControl, True, True);
  end else
  if Key=#5 then
    QHapus.ExecSQL;
end;

procedure TFormPsw.btnBatalClick(Sender: TObject);
begin
  Close;
end;

procedure TFormPsw.FormActivate(Sender: TObject);

 Begin
      Timer1.Enabled:=False;
      button1.Enabled:=True;
      PInfo.Visible:=False;
    end;
  end;}
  Nama.Clear;
  Password.Clear;
  DM.TpassWord.Open;
  DM.TpassWord.Refresh;
  ActiveControl:=Nama;
  end;
end;

procedure TFormPsw.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DM.TPassWord.Close;
end;

procedure TFormPsw.FormCreate(Sender: TObject);
begin
  idx:=1;
  CurrencyString:='';
  CurrencyFormat:=0;
  NegCurrFormat:=0;
  CurrencyDecimals:=2;
  DateSeparator:='/';
  DecimalSeparator:='.';
  ThousandSeparator:=',';
  ShortDateFormat:='dd/mm/yyyy';
end;

procedure TFormPsw.sbHelpClick(Sender: TObject);
begin
  Application.HelpFile:='CASHREG.HLP';
  Application.HelpContext(2);
end;


procedure TFormPsw.PasswordExit(Sender: TObject);
begin
  Button1Click(Sender);
end;

procedure TFormPsw.Timer1Timer(Sender: TObject);
begin
  if (idx mod 2)=0 then
    PInfo.Caption:='System Error !!!'
  else
    PInfo.Caption:='';
  inc(idx);
end;
```

```
end.

unit UDM;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Db, DBTables;

type
  TDM = class(TDataModule)
    DB: TDatabase;
    TBarang: TTable;
    TempBeli: TTable;
    TBeli: TTable;
    QDafBeli: TQuery;
    TJual: TTable;
    TempJual: TTable;
    QDafJual: TQuery;
    TPerusahaan: TTable;
    TAnalisa: TTable;
    TKassa: TTable;
    QTemp: TQuery;
    TPassword: TTable;
    TAnalisaBulan: TIntegerField;
    TAnalisaJual: TCurrencyField;
    TAnalisaBeli: TCurrencyField;
    TAnalisaLaba: TCurrencyField;
    TempJualKode: TStringField;
    TempJualNama: TStringField;
    TempJualBanyak: TFloatField;
    TempJualHarga: TCurrencyField;
    TempJualDiskon: TFloatField;
    TempBeliKode: TStringField;
    TempBeliNama: TStringField;
    TempBeliBanyak: TFloatField;
    TempBeliHarga: TCurrencyField;
    TempBeliDiskon: TFloatField;
    TempBeliJumlah: TCurrencyField;
    TempJualJumlah: TCurrencyField;
    TVendor: TTable;
    TReturBeli: TTable;
    TReturJual: TTable;
    procedure TempBeliCalcFields(DataSet: TDataSet);
    procedure TempBeliNewRecord(DataSet: TDataSet);
    procedure TempBeliAfterPost(DataSet: TDataSet);
    procedure TempJualNewRecord(DataSet: TDataSet);
    procedure TempJualAfterPost(DataSet: TDataSet);
    procedure TempJualCalcFields(DataSet: TDataSet);
    procedure TempJualAfterCancel(DataSet: TDataSet);
    procedure TAnalisaCalcFields(DataSet: TDataSet);
    procedure TempBeliAfterCancel(DataSet: TDataSet);
  private
  public
    Total_Pembelian,
    Total_Penjualan : Currency;
    function TotalBeli:Currency;
    function TotalJual:Currency;
    procedure ID_Usaha(var Nama,Alamat: string);
    procedure ID_Usaha1(var Nama,Alamat,Telpon,Fak: string);
    function Saldo_Stock(Kode:string;var SMin,SMax:real):real;
  end;

var
  DM: TDM;

implementation

{$R *.DFM}

function TDM.Saldo_Stock(Kode:string;var SMin,SMAx:real):real;
```

```
begin
  with QTemp do
  begin
    Close;
    Params[0].AsString:=Kode;
    Open;
    Result:=Fields[0].AsFloat;//sisa
    SMin:=Fields[1].AsFloat;
    SMAx:=Fields[2].AsFloat;
    Close;
  end;
end;

function TDM.TotalBeli:Currency;
var
   Tot : Currency;
begin
   try
    TempBeli.DisableControls;
    Tot:=0;
    TempBeli.First;
    while not TempBeli.EOF do
    begin
     Tot:=Tot+TempBeliJumlah.value;
     TempBeli.Next;
    end;
    Result:=Tot;
   finally
    TempBeli.EnableControls;
   end;
end;

procedure TDM.TempBeliNewRecord(DataSet: TDataSet);
begin
  TempBeliBanyak.Value:=1;
  TempBeliHarga.Value:=0;
  TempBeliDiskon.Value:=0;
end;

procedure TDM.TempBeliAfterPost(DataSet: TDataSet);
begin
  Total_Pembelian:=TotalBeli;
end;

procedure TDM.TempJualNewRecord(DataSet: TDataSet);
begin
  TempJualBanyak.Value:=1;
end;

procedure TDM.TempJualAfterPost(DataSet: TDataSet);
begin
  Total_Penjualan:=TotalJual;
end;

procedure TDM.TempJualAfterCancel(DataSet: TDataSet);
begin
  Total_Penjualan:=TotalJual;
end;

procedure TDM.TAnalisaCalcFields(DataSet: TDataSet);
begin
  TAnalisaLaba.Value:=TAnalisaJual.Value-TAnalisaBeli.Value;
end;

procedure TDM.ID_Usaha(var Nama,Alamat: string);
begin
  with TPerusahaan do
  begin
    Open;
    Refresh;
```

```
      Telp:='';
    if Length(FieldByName('Fax').AsString)>0 then
      Fax:=' Fax: '+FieldByName('Fax').AsString
    else
      Fax:='';

    Alamat:=FieldByName('Alamat').Asstring+' '+FieldByName('Kota').AsString+Telp+Fax;
  end;
end;

procedure TDM.ID_Usaha1(var Nama,Alamat,Telpon,Fak: string);
begin
  with TPerusahaan do
  begin
    Open;
    Refresh;
    Nama:=FieldByName('Nama').AsString;
    Alamat:=FieldByName('Alamat').Asstring+' '+FieldByName('Kota').AsString;
    if Length(FieldByName('Telepon').AsString)>0 then
      Telpon:=' Telp: '+FieldByName('Telepon').AsString
    else
      Fak:='';
  end;
end;

procedure TDM.TempBeliAfterCancel(DataSet: TDataSet);
begin
  Total_Pembelian:=TotalBeli;
end;
end.

unit Main;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, ToolWin, ExtCtrls, StdCtrls, Menus, Buttons;

type
  TfmMain = class(TForm)
    StatusBar: TStatusBar;
    MainMenu1: TMainMenu;
    Menu2: TMenuItem;
    Menu21: TMenuItem;
    Menu22: TMenuItem;
    Menu3: TMenuItem;
    Menu4: TMenuItem;
    Tentang1: TMenuItem;
    Menu31: TMenuItem;
    Menu32: TMenuItem;
    DaftarIsi1: TMenuItem;
    N1: TMenuItem;
    Tentang2: TMenuItem;
    Menu1: TMenuItem;
    Menu11: TMenuItem;
    N2: TMenuItem;
    Printer1: TMenuItem;
    Menu12: TMenuItem;
    Menu121: TMenuItem;
    Menu122: TMenuItem;
    Menu5: TMenuItem;
    Menu13: TMenuItem;
    PrinterSetup: TPrinterSetupDialog;
    Keluar1: TMenuItem;
    Panel1: TPanel;
    Timer1: TTimer;
    Panel3: TPanel;
    PInfo: TPanel;
    Timer2: TTimer;
    Panel2: TPanel;
    ScrollBox1: TScrollBox;
```

```
    ScrollBox2: TScrollBox;
    sbBarang: TSpeedButton;
    SpeedButton4: TSpeedButton;
    sbpenjualan: TSpeedButton;
    sbLaporan: TSpeedButton;
    sbUtilitas: TSpeedButton;
    SpeedButton8: TSpeedButton;
    sb3: TScrollBox;
    Label2: TLabel;
    Label4: TLabel;
    Timer3: TTimer;
    Image1: TImage;
    Label5: TLabel;
    Image2: TImage;
    Image3: TImage;
    Label7: TLabel;
    Label8: TLabel;
    N3: TMenuItem;
    Menu23: TMenuItem;
    N4: TMenuItem;
    Menu33: TMenuItem;
    Menu34: TMenuItem;
    Label1: TLabel;
    N5: TMenuItem;
    ReturBeli1: TMenuItem;
    ReturJual1: TMenuItem;
    procedure Menu21Click(Sender: TObject);
    procedure Menu31Click(Sender: TObject);
    procedure Menu32Click(Sender: TObject);
    procedure Menu4Click(Sender: TObject);
    procedure Tentang2Click(Sender: TObject);
    procedure Menu11Click(Sender: TObject);
    procedure Menu121Click(Sender: TObject);
    procedure Menu122Click(Sender: TObject);
    procedure Menu5Click(Sender: TObject);
    procedure Menu13Click(Sender: TObject);
    procedure Menu22Click(Sender: TObject);
    procedure Printer1Click(Sender: TObject);
    procedure Keluar1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormPaint(Sender: TObject);
    procedure FormDestroy(Sender: TObject);
    procedure DaftarIsi1Click(Sender: TObject);
    procedure Timer3Timer(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Menu23Click(Sender: TObject);
    procedure Menu33Click(Sender: TObject);
    procedure Menu34Click(Sender: TObject);
    procedure ReturBeli1Click(Sender: TObject);
    procedure ReturJual1Click(Sender: TObject);
  private
    //Bitmap: TBitmap;
    idx:Longint;
    procedure DisplayHint(Sender: TObject);
    function Hari:string;
  public
    { Public declarations }
  end;

var
  fmMain: TfmMain;

implementation

uses Barang, Beli,  Lap, About, IsiPer, Password, EditPass, Util,
  Tanggal, Kassa, IntJual, UDM, TJual, DafVend, Jual, TBeli, RetBeli,
  RetJual;
```

```
{$R *.DFM}

procedure TfmMain.DisplayHint(Sender: TObject);
begin
  Statusbar.Panels[1].Text:=Application.Hint;
end;

procedure TfmMain.Menu21Click(Sender: TObject);
begin
  Hide;
  fmBarang.ShowModal;
  Show;
end;

procedure TfmMain.Menu31Click(Sender: TObject);
begin
  Hide;
  fmBeli.ShowModal;
  Show;
end;

procedure TfmMain.Menu32Click(Sender: TObject);
begin
  fmIntJual.ShowModal;
end;

procedure TfmMain.Menu4Click(Sender: TObject);
begin
  fmLaporan.ShowModal;
end;

procedure TfmMain.Tentang2Click(Sender: TObject);
begin
  fmAbout.ShowModal;
end;

procedure TfmMain.Menu11Click(Sender: TObject);
begin
  formIsiPerusahaan.ShowModal;
end;

procedure TfmMain.Menu121Click(Sender: TObject);
begin
  Formpassword.ShowModal;
end;

procedure TfmMain.Menu122Click(Sender: TObject);
begin
  FormEditpass.ShowModal;
end;

procedure TfmMain.Menu5Click(Sender: TObject);
begin
  fmUtilitas.ShowModal;
end;

procedure TfmMain.Menu13Click(Sender: TObject);
begin
  fmTanggal.ShowModal;
end;

procedure TfmMain.Menu22Click(Sender: TObject);
begin
  fmKassa.ShowModal;
end;

procedure TfmMain.Printer1Click(Sender: TObject);
begin
  PrinterSetup.Execute;
```

```
end;

procedure TfmMain.Keluar1Click(Sender: TObject);
begin
  if MessageDlg('Keluar dari aplikasi ini ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
  begin
    Close;
    Application.Terminate;
    Close;
  end;
end;

procedure TfmMain.FormCreate(Sender: TObject);
begin
  Application.OnHint:=DisplayHint;
  idx:=1;
end;

procedure TfmMain.Timer1Timer(Sender: TObject);
begin
  StatusBar.Panels[0].Text:=Hari+' ,'+FormatDateTime('dd-mm-yyyy',Now)+' '+
  TimeToStr(Time);
end;

procedure TfmMain.Timer2Timer(Sender: TObject);
begin
  pInfo.Left:=PInfo.Left+1;
  if PInfo.Left=614 then
    PInfo.Left:=-586;
end;

procedure TfmMain.FormDestroy(Sender: TObject);
begin
  {Bitmap.Free;}
end;

procedure TfmMain.DaftarIsi1Click(Sender: TObject);
begin
  Application.HelpFile:='CASHREG.HLP';
  Application.HelpCommand(HELP_CONTENTS,0);
end;

procedure TfmMain.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  Keluar1Click(Sender);
end;

procedure TfmMain.Menu23Click(Sender: TObject);
begin
  fmDafVendor.ShowModal;
end;

procedure TfmMain.Menu33Click(Sender: TObject);
begin
  Hide;
  fmTBeli.ShowModal;
  Show;
end;

procedure TfmMain.Menu34Click(Sender: TObject);
begin
  Hide;
  fmTJual.ShowModal;
  Show;
end;

procedure TfmMain.ReturBeli1Click(Sender: TObject);
begin
  fmRetBeli.ShowModal;
end;
```

```
procedure TfmMain.ReturJual1Click(Sender: TObject);
begin
  fmRetJual.SHowModal;
end;

end.

unit About;
interface
uses Windows, SysUtils, Classes, Graphics, Forms, Controls, StdCtrls,
 Buttons, ExtCtrls;

type
 TfmAbout = class(TForm)
   Panel1: TPanel;
   ProductName: TLabel;
   Version: TLabel;
   Comments: TLabel;
   Label1: TLabel;
   Label2: TLabel;
   SpeedButton1: TSpeedButton;
   Label3: TLabel;
   Memo1: TMemo;
   procedure SpeedButton1Click(Sender: TObject);
 private
   { Private declarations }
 public
   { Public declarations }
 end;
var
 fmAbout: TfmAbout;

implementation

{$R *.DFM}
procedure TfmAbout.SpeedButton1Click(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

end.

unit Analisa;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, Buttons, Mask, DBCtrls, Db, DBTables, Math;

type
 TfmAnalisa = class(TForm)
   Panel1: TPanel;
   Panel2: TPanel;
   Panel3: TPanel;
   Panel4: TPanel;
   Label1: TLabel;
   EKode: TEdit;
   EKeterangan: TEdit;
   Label2: TLabel;
   sbCari: TSpeedButton;
   Panel5: TPanel;
   Label3: TLabel;
   Label4: TLabel;
   Label5: TLabel;
   Label6: TLabel;
   Label7: TLabel;
   Label8: TLabel;
   Label9: TLabel;
   Label10: TLabel;
   Label11: TLabel;
```

```
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label16: TLabel;
    JumlahLaba: TEdit;
    JumlahBeli: TEdit;
    JumlahJual: TEdit;
    Label17: TLabel;
    Label18: TLabel;
    QBeli: TQuery;
    Beli1: TEdit;
    Beli2: TEdit;
    Beli3: TEdit;
    Beli4: TEdit;
    Beli5: TEdit;
    Beli6: TEdit;
    Beli7: TEdit;
    Beli8: TEdit;
    Beli9: TEdit;
    Beli10: TEdit;
    Beli11: TEdit;
    Beli12: TEdit;
    QJual: TQuery;
    QAnalisa: TQuery;
    Laba1: TEdit;
    Laba2: TEdit;
    Laba3: TEdit;
    Laba4: TEdit;
    Laba5: TEdit;
    Laba6: TEdit;
    Laba7: TEdit;
    Laba8: TEdit;
    Laba9: TEdit;
    Laba10: TEdit;
    Laba11: TEdit;
    Laba12: TEdit;
    QDelete: TQuery;
    Jual1: TEdit;
    Jual2: TEdit;
    Jual3: TEdit;
    Jual4: TEdit;
    Jual5: TEdit;
    Jual6: TEdit;
    Jual7: TEdit;
    Jual8: TEdit;
    Jual9: TEdit;
    Jual10: TEdit;
    Jual11: TEdit;
    Jual12: TEdit;
    QUpdate: TQuery;
    SpeedButton1: TSpeedButton;
    btnTutup: TSpeedButton;
    btnGrafik: TSpeedButton;
    QInsert: TQuery;
    procedure sbCariClick(Sender: TObject);
    procedure EKodeExit(Sender: TObject);
    procedure EKodeKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure SpeedButton1Click(Sender: TObject);
    procedure btnGrafikClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
  private
    procedure Tampil_Analisa;
  public
    procedure CekTabelAnalisa;
    procedure Analisa;
    procedure Hitung_Analisa(Kode: string);
```

```
 end;

var
 fmAnalisa: TfmAnalisa;

implementation

uses LuBarang, Graph, UDM;

{$R *.DFM}

procedure TfmAnalisa.CekTabelAnalisa;
var
  i : integer;
begin
  DM.TAnalisa.Open;
  if DM.TAnalisa.RecordCount<1 then
  begin
    with QInsert do
    begin
      try
        DisableControls;
        for i:=1 to 12 do
        begin
          ParamByName('Bulan').AsInteger:=i;
          ExecSQL;
        end;
      finally
        EnableControls;
        Close;
      end;
    end;
  end;
  DM.TAnalisa.Close;
end;
Procedure TfmAnalisa.Hitung_Analisa(Kode: string);
var
  i : byte;
begin
  //analisa penjualan
  with QBeli do
  begin
    for i:=1 to 12 do
    begin
      Close;
      ParamByName('P1').AsInteger:=i;
      ParamByName('P2').AsString:=Kode;
      Open;
      with QUpdate do
      begin
        Close;
        SQL.Clear;
        SQL.Add('Update Analisa Set Beli=:p1 Where Bulan=:P2');
        Params[0].AsCurrency:=QBeli.Fields[0].AsCurrency;
        Params[1].AsInteger:=i;
        ExecSQL;
      end;
    end;
  end;
      with QUpdate do
      begin
        Close;
        SQL.Clear;
        SQL.Add('Update Analisa Set Jual=:p1 Where Bulan=:P2');
        ExecSQL;
      end;
    end;
  end;
  with QUpdate do
  begin
```

```
    for i:=1 to 12 do
    begin
     Close;
     SQL.Clear;
     SQL.Add('Update Analisa Set Laba=Jual-Beli');
     ExecSQL;
    end;
   end;
end;

procedure TfmAnalisa.Tampil_Analisa;
var
  TJual, TBeli, TLaba: array[1..12] of Double;
  i: byte;
  SumJual, SumBeli, SumLaba : Currency;
begin
  with QAnalisa do
  begin
    for i:=1 to 12 do
    begin
      Close;
      Params[0].AsInteger:=i;
      Open;
      TJual[i]:=Fields[0].AsCurrency;
      TBeli[i]:=Fields[1].AsCurrency;
      TLaba[i]:=TJual[i]-TBeli[i];
     end;
   end;

    Jual1.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[1]);
    Jual2.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[2]);
    Jual3.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[3]);
    Jual4.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[4]);
    Jual5.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[5]);
    Jual6.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[6]);
    Jual7.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[7]);
    Jual8.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[8]);
    Jual9.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[9]);
    Jual10.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[10]);
    Jual11.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[11]);
    Jual12.Text:=FormatCurr('#,#0.00;(#,#0.00)',TJual[12]);

    Beli1.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[1]);
    Beli2.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[2]);
    Beli3.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[3]);
    Beli4.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[4]);
    Beli5.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[5]);
    Beli6.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[6]);
    Beli7.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[7]);
    Beli8.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[8]);
    Beli9.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[9]);
    Beli10.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[10]);
    Beli11.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[11]),
    Beli12.Text:=FormatCurr('#,#0.00;(#,#0.00)',TBeli[12]);


    SumJual:=Sum(TJual);
    SumBeli:=Sum(TBeli);
    SumLaba:=Sum(TLaba);

    JumlahJual.Text:=FormatCurr('#,#0.00;(#,#0.00)',SumJual);
    JumlahBeli.Text:=FormatCurr('#,#0.00;(#,#0.00)',SumBeli);
    JumlahLaba.Text:=FormatCurr('#,#0.00;(#,#0.00)',SumLaba);
  end;

  procedure TfmAnalisa.sbCariClick(Sender: TObject);
  begin
    with fmLuBarang do
    begin
      ShowModal;
```

```
    if ModalResult=mrOK then
    begin
      EKode.Text:=KodeCari;
      EKeterangan.Text:=NamaCari;
      EKodeExit(Sender);
    end;
  end;
end;

procedure TfmAnalisa.EKodeExit(Sender: TObject);
begin
  if DM.TBarang.FindKey([EKode.Text]) then
  begin
    EKeterangan.Text:=DM.TBarang.Fields[1].AsString;
    Analisa;
  end else
  begin
    MessageDlg('Daftar barang belum ada !',mtInformation,[mbOK],0);
    ActiveControl:=EKode;
  end;
end;

procedure TfmAnalisa.EKodeKeyDown(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  if Key=VK_DOWN then
    sbCariClick(Sender);
end;

procedure TfmAnalisa.SpeedButton1Click(Sender: TObject);
begin
  ModalResult:=MrOK;
end;

procedure TfmAnalisa.btnGrafikClick(Sender: TObject);
begin
  fmPilihanGrafik.ShowModal;
end;

procedure TfmAnalisa.FormActivate(Sender: TObject);
begin
  DM.TBarang.Open;
  DM.TAnalisa.Open;
  EKode.Clear;
  EKeterangan.Clear;
  ActiveControl:=EKode;
end;

procedure TfmAnalisa.FormKeyPress(Sender: TObject; var Key: Char);
begin
  if Key=#13 then
  begin
    SelectNext(ActiveControl, True, True);
    Key:=#0;
  end;
end;
procedure TfmAnalisa.FormClose(Sender: TObject; var Action: TCloseAction);

end.

unit Barang;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, Db, Grids, DBGrids, ComCtrls, Buttons, DBTables, StdCtrls;

type
  TfmBarang = class(TForm)
```

```
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    TabelBarang: TDBGrid;
    DsBarang: TDataSource;
    sbRecord: TSpeedButton;
    SpeedButton2: TSpeedButton;
    QKriteria: TQuery;
    btnTutup: TSpeedButton;
    SpeedButton1: TSpeedButton;
    btnAnalisa: TSpeedButton;
    btnBantuan: TSpeedButton;
    Panel5: TPanel;
    Panel8: TPanel;
    Panel9: TPanel;
    SpeedButton3: TSpeedButton;
    SpeedButton4: TSpeedButton;
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure sbRecordClick(Sender: TObject);
    procedure TabelBarangKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure SpeedButton2Click(Sender: TObject);
    procedure btnTutupClick(Sender: TObject);
    procedure btnAnalisaClick(Sender: TObject);
    procedure btnBantuanClick(Sender: TObject);
    procedure SpeedButton3Click(Sender: TObject);
    procedure SpeedButton4Click(Sender: TObject);
  private
    procedure TampilKan;
  public
  end;

var
  fmBarang: TfmBarang;

implementation

uses detil, Kriteria, UDM, Analisa;

{$R *.DFM}

procedure TfmBarang.FormActivate(Sender: TObject);
begin
  with DSBarang.DataSet do
  begin
    Close;
    Open;
    Refresh;
  end;
end;

procedure TfmBarang.FormClose(Sender: TObject; var Action: TCloseAction);
begin

  DSBarang.DataSet.Close;
end;

procedure TfmBarang.TabelBarangKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if Key=13 then
    sbRecordClick(Sender);
end;

procedure TfmBarang.TampilKan;
var
  WhereUsed: boolean;
  HargaJual: Currency;
```

```
  Sisa    : real;

  procedure TfmBarang.SpeedButton2Click(Sender: TObject);
begin
  Tampilkan;
end;

procedure TfmBarang.btnTutupClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmBarang.btnAnalisaClick(Sender: TObject);
begin
  if TabelBarang.Fields[0].AsString<>'' then
  begin
    Hide;
    fmAnalisa.ShowModal;
    Show;
    FormActivate(Sender);
  end;
end;


procedure TfmBarang.btnBantuanClick(Sender: TObject);
begin
   Application.HelpFile:='CASHREG.HLP';
   Application.HelpContext(14);
end;

procedure TfmBarang.SpeedButton3Click(Sender: TObject);
begin
   DM.TBarang.IndexName:='';
end;

procedure TfmBarang.SpeedButton4Click(Sender: TObject);
begin
  DM.TBarang.IndexName:='BRG_NAMA';
end;

end.

unit Beli;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, ComCtrls, Grids, DBGrids, Db, DBTables, Buttons;

type
  TfmBeli = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel5: TPanel;
    TabelBeli: TDBGrid;
    DsTempBeli: TDataSource;
    Label5: TLabel;
    Label6: TLabel;
    btnHapus: TSpeedButton;
    btnSImpan: TSpeedButton;
    btnBatal: TSpeedButton;
    btnBantuan: TSpeedButton;
    btnTutup: TSpeedButton;
    QInsert: TQuery;
    QDel: TQuery;
    QUpdate: TQuery;
    SpeedButton2: TSpeedButton;
    PTotal: TPanel;
```

```
    LbTotal: TLabel;
    Panel6: TPanel;
    Bevel1: TBevel;
    Bevel2: TBevel;
    LbKode: TLabel;
    sbCari: TSpeedButton;
    Bevel3: TBevel;
    Bevel4: TBevel;
    Bevel5: TBevel;
    Bevel6: TBevel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Panel4: TPanel;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    ENoFak: TEdit;
    DTanggal: TDateTimePicker;
    EKasir: TLabel;
    Timer1: TTimer;
    QPeriksaFaktur: TQuery;
    Label4: TLabel;
    QKassa: TQuery;
    EPemasok: TEdit;
    SpeedButton3: TSpeedButton;
    QDelete: TQuery;
    procedure FormCreate(Sender: TObject);
    procedure TabelBeliKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure DsTempBeliDataChange(Sender: TObject; Field: TField);
    procedure FormActivate(Sender: TObject);
    procedure btnSImpanClick(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure btnTutupClick(Sender: TObject);
    procedure btnHapusClick(Sender: TObject);
    procedure sbCariClick(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure TabelBeliExit(Sender: TObject);
    procedure btnBantuanClick(Sender: TObject);
    procedure TabelBeliColExit(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure ENoFakKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure ENoFakExit(Sender: TObject);
    procedure DsTempBeliUpdateData(Sender: TObject);
    procedure EPemasokExit(Sender: TObject);
    procedure SpeedButton3Click(Sender: TObject);
  private
    Idx : Longint;
    Jumbeli: string;
    HargaPerUnit,TotalBeli: Currency;
    JumSaldo:real;
    procedure Simpan(Strook: string;
                     Kode_Barang:string;
                     Tanggal:TDateTime;
                     Jumlah: real;
                     Harga: Currency;
                     ID,
                     Pemasok: string;
                     Diskon: real);

    procedure Isi_Pembelian;
    procedure Update_Persediaan(Kode: string;Jumlah: real);
    procedure EditPembelian(Strook:string);
  public
    StateEdit: boolean;
```

```
  procedure Init;
end;

var
 fmBeli: TfmBeli;

implementation

uses LuBarang, UDM, TBeli, PSW, LuVendor, Vendor;

{$R *.DFM}

procedure TfmBeli.EditPembelian(Strook:string);
begin
  with QDelete do
  begin
    SQL.Text:='Select * From Beli Where Strook='''+Strook+'''';
    Open;First;
    while not EOF do
    begin
      Update_Persediaan(FieldByName('Kode_barang').AsString,
                FieldByName('Jumlah').AsFloat*-1);
      Next;
    end;
  end;
  with QDelete do
  begin
    SQL.Text:='Delete From Beli Where Strook='''+Strook+'''';
    ExecSQL;
  end;
end;

procedure TfmBeli.FormCreate(Sender: TObject);
begin
  DTanggal.Date:=Now;
  SysUtils.CurrencyString:='';
end;

procedure TfmBeli.TabelBeliKeyDown(Sender: TObject; var Key: Word;
 Shift: TShiftState);
var
  s: string;
begin
  if Key=VK_F3 then
  begin
    if TabelBeli.SelectedField=DM.TempBeliKode then
    begin
      s:=InputBox('Barang ','Kode :','');
      with DM,TBarang do
      begin
        if (FindKey([s])) and (s<>'') then
        begin
          if not (DSTempBeli.DataSet.State in[dsInsert]) then
          TempBeliNama.value:=FieldByName('Nama_Barang').AsString;
        end;
      end;
    end;
  end else
  if Key=13 then
  begin
    if TabelBeli.SelectedField=DM.TempBeliDiskon then
    begin
      with DSTempBeli.DataSet do
      begin
       if State in[dsEdit,dsInsert] then
         Post
       else
         begin
           Edit;
           Post;
```

```
        end;
      Last;
      Append;
      TabelBeli.SelectedField:=DM.TempBeliKode;
     end;
    end;
  end;
end;

procedure TfmBeli.Update_Persediaan(Kode: string;Jumlah: real);
begin
  with QUpdate do
  begin
    Close;
    ParamByName('P1').AsFloat:=Jumlah;
    ParamByName('P2').AsString:=Kode;
    ExecSQL;
  end;
end;

procedure TfmBeli.DsTempBeliDataChange(Sender: TObject; Field: TField);
begin
  TotalBeli:=DM.Total_Pembelian;
  LBTotal.Caption:=FormatFloat('#,#0.00',TotalBeli);
  JumBeli:=FormatFloat('#,#0.00',TotalBeli);
end;

procedure TfmBeli.Simpan(Strook: string;
                  Kode_Barang:string;
                  Tanggal:TDateTime;
                  Jumlah: real;
                  Harga: Currency;
                  ID,Pemasok: string;
                  Diskon: real);
begin
  with QInsert do
  begin
    Close;
    ParamByName('Strook').AsString:=Strook;
    ParamByName('Kode_Barang').AsString:=Kode_Barang;
    ParamByName('Harga').AsCurrency:=Harga;
    ParamByName('ID').AsString:=ID;
    ParamByName('Pemasok').AsString:=Pemasok;
    ParamByName('Diskon').AsFloat:=Diskon;
    ExecSQL;
    Close;
  end;
end;

procedure TfmBeli.Init;
begin
  QDel.ExecSQL;
  with DSTempBeli.DataSet do
  begin
    Append;
    Edit;
  end;
end;

procedure TfmBeli.FormActivate(Sender: TObject);
var
  Fld: TField;
begin
  DSTempBeli.DataSet.Open;
  DM.TBarang.Open;
  DM.TBeli.Open;
  if not StateEdit then
  begin
    Init;
    EKasir.Caption:=FormPsw.Pemakai;
```

```
    ENofak.Clear;
  end;
  ActiveCOntrol:=ENoFak;
  DSTempBeliDataChange(Sender, Fld);
end;

procedure TfmBeli.btnSImpanClick(Sender: TObject);
begin
  if MessageDlg('Benar data ini akan disimpan ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
  begin
    if (TotalBeli<>0)and(ENoFak.Text<>'')and(EPemasok.Text<>'')then
    begin
      Isi_Pembelian;
      FormActivate(Sender);
    end else
    begin
      ActiveControl:=TabelBeli;
      MessageDlg('Belum ada transaksi atau lihat kode faktur atau Pemasok!',mtInformation,[mbOK],0);
    end;
  end;
end;

procedure TfmBeli.btnBatalClick(Sender: TObject);
var
  Fld: TField;
begin
  Init;
  DsTempBeliDataChange(Sender,Fld);
  FormActivate(Sender);
end;

procedure TfmBeli.btnTutupClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmBeli.btnHapusClick(Sender: TObject);
begin
  DSTempBeli.DataSet.Delete;
end;

procedure TfmBeli.sbCariClick(Sender: TObject);
begin
  with fmLuBarang do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      if not (DSTempBeli.DataSet.State in[dsInsert]) then
        DSTempBeli.DataSet.Append;
      DSTempBeli.DataSet.Edit;
      DM.TempBeliNama.value:=NamaCari;
    end;
  end;
end;

procedure TfmBeli.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DM.TBarang.Close;
  DM.TBeli.Close;
  DSTempBeli.DataSet.Close;
end;

procedure TfmBeli.TabelBeliExit(Sender: TObject);
var
  Fld: TField;
begin
  with DSTempBeli.DataSet do
  begin
   if (TotalBeli<>0) and (recordcount>0) then
```

```
  begin
   Edit;
   Post;
   DSTempBeliDataChange(Sender,Fld);
  end else
  begin
   Last;
   Edit;
   Append;
  end;
 end;
end;


procedure TfmBeli.btnBantuanClick(Sender: TObject);
begin
  Application.HelpFile:='CASHREG.HLP';
  Application.HelpContext(10);
end;

procedure TfmBeli.TabelBeliColExit(Sender: TObject);
begin
 if TabelBeli.SelectedField=DM.TempBeliDiskon then
 begin
   DsTempBeli.DataSet.Refresh;
   DsTempBeli.DataSet.Append;
 end
 else
   DsTempBeli.DataSet.Refresh;
 if (TabelBeli.SelectedField=DM.TempBeliKode)and(DM.TempBeliKode.Value<>'')then
 begin
   if DM.TBarang.FindKey([DM.TempBeliKode.Value]) then
   begin
     MessageDlg('Kode barang belum ada !',mtInformation,[mbOK],0);
     DM.TempBeli.Delete;
   end;
 end;
end;


procedure TfmBeli.Timer1Timer(Sender: TObject);
begin
  if (Idx mod 2=0) then
    LbTotal.Caption:=JumBeli
  else
    LbTotal.Caption:='';
  inc(Idx);
end;

procedure TfmBeli.ENoFakKeyDown(Sender: TObject; var Key: Word;
    Shift: TShiftState);
begin
  if Key=13 then
    ActiveControl:=TabelBeli;
end;

procedure TfmBeli.ENoFakExit(Sender: TObject);
begin
  with QPeriksaFaktur do
  begin
    Close;
    Params[0].AsString:=ENoFak.Text;
    Open;
    if Fields[0].AsInteger>0 then
    begin
      MessageDlg('No.faktur sudah digunakan, ganti dengan yang lain !',mtInformation,[mbOK],0);
      ActiveControl:=ENoFak;
    end else
      ActiveControl:=TabelBeli;
  end;
```

```
end;

procedure TfmBeli.DsTempBeliUpdateData(Sender: TObject);
var
 Saldo,SaldoMinimum,SMAX: real;
begin
 if not StateEdit then
  begin
    MessageDlg('Stock barang melebihi Batas Maksimum !',mtInformation,[mbOK],0);
    DSTempBeli.Dataset.Cancel;
  end;
 end;
end;

procedure TfmBeli.EPemasokExit(Sender: TObject);
begin
 if Length(EPemasok.Text)>0 then
 begin
  DM.TVendor.Filtered:=False;
  if not DM.TVendor.FindKey([EPemasok.Text]) then
  begin
   MessageDlg('Pemasok belum ada !',mtInformation,[mbOK],0);
   ActiveControl:=EPemasok;
  end;
 end;
end;

procedure TfmBeli.SpeedButton3Click(Sender: TObject);
begin
 with fmLuVendor do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      EPemasok.Text:=KodeVendorCari;
      EPemasokExit(Sender);
    end;
  end;
end;

End.

unit DafVend;

interface

uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 Db, Grids, DBGrids, ExtCtrls, StdCtrls, Buttons, DBTables;

type
 TfmDafVendor = class(TForm)
   DsVendor: TDataSource;
   Bevel5: TBevel;
   Label4: TLabel;
   Panel3: TPanel;
   Panel4: TPanel;
   Panel1: TPanel;
   Panel5: TPanel;
   Panel6: TPanel;
   ScrollBox1: TScrollBox;
   TabelVendor: TDBGrid;
   SpeedButton1: TSpeedButton;
   Label1: TLabel;
   btnRecord: TBitBtn;
   BitBtn2: TBitBtn;
   Panel7: TPanel;
   Panel8: TPanel;
   Panel9: TPanel;
   Panel10: TPanel;
```

```
    Panel2: TPanel;
    procedure btnRecordClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure SpeedButton1Click(Sender: TObject);
    procedure TabelVendorKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure Panel7Click(Sender: TObject);
    procedure Panel2Click(Sender: TObject);
  private
  public
    KodePilih:string;
  end;

var
  fmDafVendor: TfmDafVendor;

implementation

uses Vendor, Kriteria, UDM;

{$R *.DFM}

procedure TfmDafVendor.btnRecordClick(Sender: TObject);
begin
  Hide;
  if DSVendor.DataSet.RecordCount>0 then
  begin
   with fmVendor do
   begin
     EKode.Text:=TabelVendor.Fields[0].AsString;
     FormActivate(Sender);
     ShowModal;
   end;
  end else
  begin
    with fmVendor do
    begin
      StateEdit:=False;
      ShowModal;
    end;
  end;
  Show;
end;

procedure TfmDafVendor.FormActivate(Sender: TObject);
begin
  with DSVendor.DataSet do
  begin
    Close;
    Open;
    Refresh;
  end;
end;

procedure TfmDafVendor.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
  DSVendor.DataSet.Close;
end;

procedure TfmDafVendor.SpeedButton1Click(Sender: TObject);
begin
  ModalREsult:=mrOK;
end;

procedure TfmDafVendor.TabelVendorKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if Key=13 then
```

```
    btnRecordClick(Sender);
end;

procedure TfmDafVendor.Panel7Click(Sender: TObject);
begin
  DM.TVendor.IndexName:='Vendor_Nama';
end;

procedure TfmDafVendor.Panel2Click(Sender: TObject);
begin
  DM.TVendor.IndexName:='';
end;

end.

unit detil;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, Db, Grids, DBGrids, ComCtrls, Buttons, StdCtrls, DBTables;

const
  Defaultvalue='0';
type
  TfmDetil = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    GroupBox1: TGroupBox;
    EKode: TEdit;
    EKeterangan: TEdit;
    EHargaJual: TEdit;
    EUnit: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    sbKode: TSpeedButton;
    EDiskon: TEdit;
    Label5: TLabel;
    Query1: TQuery;
    SQLBarang: TUpdateSQL;
    btnEdit: TSpeedButton;
    btnTambah: TSpeedButton;
    btnHapus: TSpeedButton;
    btnBatal: TSpeedButton;
    btnSimpan: TSpeedButton;
    btnBantuan: TSpeedButton;
    btnTutup: TSpeedButton;
    SpeedButton1: TSpeedButton;
    QLoad: TQuery;
    Status: TStaticText;
    Timer1: TTimer;
    QUpdateTransaksi: TQuery;
    EJumlah: TEdit;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    EStockMinimum: TEdit;
    EStockMaksimum: TEdit;
    Label9: TLabel;
    cbKassa: TComboBox;
    QKassa: TQuery;
    procedure btnTambahClick(Sender: TObject);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure EKodeExit(Sender: TObject);
    procedure btnSimpanClick(Sender: TObject);
```

```
  procedure FormActivate(Sender: TObject);
  procedure FormShow(Sender: TObject);
  procedure btnEditClick(Sender: TObject);
  procedure btnHapusClick(Sender: TObject);
  procedure btnBatalClick(Sender: TObject);
  procedure FormClose(Sender: TObject; var Action: TCloseAction);
  procedure btnTutupClick(Sender: TObject);
  procedure sbKodeClick(Sender: TObject);
  procedure EKodeKeyUp(Sender: TObject; var Key: Word;
    Shift: TShiftState);
  procedure Timer1Timer(Sender: TObject);
  procedure btnBantuanClick(Sender: TObject);
  procedure EHargaJualKeyPress(Sender: TObject; var Key: Char);
  procedure EDiskonKeyPress(Sender: TObject; var Key: Char);
  procedure EJumlahKeyPress(Sender: TObject; var Key: Char);
  procedure EStockMaksimumExit(Sender: TObject);
 private
  i : Longint;
  Pesan : string;
  procedure Clear;
  procedure Toggle(State:boolean);
  procedure UpdateTransaksi(Kode: string);
  procedure StateOff;
  procedure CekStatus;
  procedure LoadKassa;
 public
  StateEdit,StateAdd, StateDel : boolean;
 end;

var
 fmDetil: TfmDetil;

implementation

uses UDM, LuBarang;

{$R *.DFM}

procedure TfmDetil.LoadKassa;
begin
  cbKassa.Clear;
  with QKassa do
  begin
    Close;Open;First;
    while not EOF do
    begin
      cbKassa.Items.Add(Fields[0].AsString+'    '+Fields[1].AsString);
      Next;
    end;
  end;
end;

procedure TfmDetil.CekStatus;
begin
  if (StateAdd)or(StateEdit)or(StateDel) then
  begin
    btnEdit.Enabled:=False;
    btnTambah.Enabled:=False;
    btnHapus.Enabled:=False;
  end else
  begin
    btnEdit.Enabled:=True;
    btnTambah.Enabled:=True;
    btnHapus.Enabled:=True;
  end;
end;

procedure TfmDetil.StateOff;
begin
 StateEdit:=False;
```

```
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Buttons, StdCtrls, Db, DBTables, ExtCtrls;

type
  TfmIntJual = class(TForm)
    QKassa: TQuery;
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    GroupBox2: TGroupBox;
    Label3: TLabel;
    Label4: TLabel;
    EKasir: TEdit;
    cbKassa: TComboBox;
    btnTransaksi: TBitBtn;
    btnBatal: TBitBtn;
    procedure FormActivate(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure btnTransaksiClick(Sender: TObject);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
  private
     procedure LoadKassa;
  public
    { Public declarations }
  end;

var
  fmIntJual: TfmIntJual;

implementation

uses PSW, Jual, Main;

{$R *.DFM}

procedure TfmIntJual.LoadKassa;
begin
   cbKassa.Clear;
   with QKassa do
   begin
      Close;
      Open;
      First;
end;

procedure TfmIntJual.FormActivate(Sender: TObject);
begin
   EKasir.Text:=FormPSW.Pemakai;
   LoadKassa;
end;

procedure TfmIntJual.btnBatalClick(Sender: TObject);
begin
   Close;
end;

procedure TfmIntJual.btnTransaksiClick(Sender: TObject);
begin
   if cbKassa.ItemIndex>-1 then
   begin
      Close;
      ModalResult:=mrOK;
      with fmJual do
      begin
        Close;
        Kasir:=EKasir.Text;
        Kassa:=cbKassa.Items[cbKassa.itemIndex];
        EKassa.Caption:=Kassa;
```

```
      EOperator.Caption:=Kasir;
      fmMain.Hide;
      ShowModal;
      fmMain.Show;
    end;
  end else
  begin
    MessageDlg('Kassa harus diisi dulu !',mtInformation,[mbOK],0);
    ActiveControl:=cbKassa;
  end;
end;
procedure TfmIntJual.FormKeyPress(Sender: TObject; var Key: Char);
begin
  if Key=#13 then
  begin
    SelectNext(ActiveControl, True, True);
    Key:=#0;
  end;
end;

end.

unit Jual;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, ComCtrls, Grids, DBGrids, Db, DBTables, Buttons;

type
 TfmJual = class(TForm)
   Panel1: TPanel;
   Panel2: TPanel;
   Panel4: TPanel;
   Label1: TLabel;
   Label2: TLabel;
   DTanggal: TDateTimePicker;
   Panel5: TPanel;
   TabelJual: TDBGrid;
   DsTempJual: TDataSource;
   Label5: TLabel;
   Label6: TLabel;
   QInsert: TQuery;
   QDel: TQuery;
   QUpdate: TQuery;
   SpeedButton2: TSpeedButton;
   PTotal: TPanel;
   LbTotal: TLabel;
   Timer1: TTimer;
   QStrook: TQuery;
   EBayar: TEdit;
   EKembali: TEdit;
   ETotal: TEdit;
   Label3: TLabel;
   Label7: TLabel;
   SpeedButton3: TSpeedButton;
   btnSimpan: TBitBtn;
   btnBatal: TBitBtn;
   SpeedButton5: TSpeedButton;
   Panel3: TPanel;
   Bevel1: TBevel;
   Bevel2: TBevel;
   LbKode: TLabel;
   sbCari: TSpeedButton;
   Bevel3: TBevel;
   Bevel4: TBevel;
   Bevel5: TBevel;
   Bevel6: TBevel;
   Label9: TLabel;
   Label10: TLabel;
   Label11: TLabel;
```

```
    Label12: TLabel;
    Label13: TLabel;
    QKassa: TQuery;
    Label14: TLabel;
    ENoFak: TLabel;
    Label4: TLabel;
    GroupBox1: TGroupBox;
    EJenisBayar: TEdit;
    ENomorBayar: TEdit;
    Label8: TLabel;
    Label15: TLabel;
    QTemp: TQuery;
    EKassa: TStaticText;
    Label16: TLabel;
    EOperator: TStaticText;
    QDelete: TQuery;
    procedure FormCreate(Sender: TObject);
    procedure TabelJualKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure DsTempJualDataChange(Sender: TObject; Field: TField);
    procedure FormActivate(Sender: TObject);
    procedure btnSImpanClick(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure btnTutupClick(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure EBayarKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure TabelJualExit(Sender: TObject);
    procedure sbCariClick(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure EBayarKeyPress(Sender: TObject; var Key: Char);
    procedure DsTempJualUpdateData(Sender: TObject);
    procedure SpeedButton5Click(Sender: TObject);
    procedure TabelJualColExit(Sender: TObject);
    procedure TabelJualKeyPress(Sender: TObject; var Key: Char);
    procedure EBayarKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure EBayarExit(Sender: TObject);
  private
    StateCancel: boolean;
    idx: Longint;
    HargaPerUnit, TotalJual: Currency;
    JumSaldo: real;
    Jumlah: string;
    procedure Simpan(Strook: Longint;
                     Kode_Barang:string;
                     Tanggal:TDateTime;
                     Jumlah: real;
                     Diskon:real;
                     Kode_Kasa: string;
                     ID: string;
                     Jenis_Kode,
                     Jenis_Nama:string;
                     Harga_Jual,HPP:Currency),

    procedure Isi_Penjualan;
    procedure Update_Persediaan(Kode: string;Jumlah: real);
    procedure Clear;
    procedure CetakFaktur;
    function HitungHPP(Kode:string):currency;
    procedure EditPenjualan(Strook:integer);
  public
    Kassa, Kasir : string;
    StateEdit: boolean;
    procedure Init;
  end;

var
  fmJual: TfmJual;
```

```
implementation

uses LuBarang, UDM, TBeli, TJual, PSW, FakJual;

{$R *.DFM}

procedure TfmJual.EditPenjualan(Strook:integer);
begin
  with QDelete do
  begin
    SQL.Text:='Select * From Jual Where Strook='''+IntToStr(Strook)+'''';
    Open;First;
    while not EOF do
    begin
      Update_Persediaan(FieldByName('Kode_barang').AsString,
                FieldByName('Jumlah').AsFloat*-1);
      Next;
    end;
  end;
  with QDelete do
  begin
    SQL.Text:='Delete From Jual Where Strook='''+IntToStr(Strook)+'''';
    ExecSQL;
  end;
end;

procedure TfmJual.Clear;
begin
  EBayar.Clear;
  Ekembali.Clear;
  EJenisBayar.Clear;
  ENomorBayar.Clear;
end;

function TfmJual.HitungHPP(Kode:string):currency;
begin
  with QTemp do
  begin
    Close;
    Params[0].AsString:=Kode;
    Open;
    Result:=Fields[0].AsCurrency;
    Close;
  end;
end;

procedure TfmJual.FormCreate(Sender: TObject);
begin
   DTanggal.Date:=Now;
   SysUtils.CurrencyString:='';
   idx:=1;
end;

procedure TfmJual.TabelJualKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
var
  s:string;
begin
  if Key=VK_F3 then
    sbCariClick(Sender);
  end else
  if Key=13 then
  begin
    if TabelJual.SelectedField=DM.TempJualDiskon then
    begin
      with DSTempJual.DataSet do
      begin
       if State in[dsEdit,dsInsert] then
         Post
```

```
      else
        begin
          Edit;
          Post;
        end;
       Last;
       Append;
       TabelJual.SelectedField:=DM.TempJualKode;
      end;
    end;
  end;
end;

procedure TfmJual.Update_Persediaan(Kode: string;Jumlah: real);
begin
  with QUpdate do
  begin
    Close;
    ParamByName('P1').AsFloat:=Jumlah;
    ParamByName('P2').AsString:=Kode;
    ExecSQL;
  end;
end;

procedure TfmJual.DsTempJualDataChange(Sender: TObject; Field: TField);
begin
  Jumlah:=FormatCurr('#,#0.00',DM.Total_Penjualan);
  LbTotal.Caption:=FormatCurr('#,#0.00',DM.Total_Penjualan);
  ETotal.Text:=FormatCurr('#,#0.00',DM.Total_Penjualan);
  TotalJual:=DM.Total_Penjualan;
end;

procedure TfmJual.Simpan(Strook: Longint;
                  Kode_Barang:string;
                  Tanggal:TDateTime;
                  Jumlah: real;
                  Diskon:real;
                  Kode_Kasa: string;
                  ID: string;
                  Jenis_Kode,
                  Jenis_Nama:string;
                  Harga_Jual,HPP:Currency);
begin
  with QInsert do
  begin
    Close;
    ParamByName('Strook').AsInteger:=Strook;
    ParamByName('Kode_Barang').AsString:=Kode_Barang;
    ParamByName('Tanggal').AsDate:=Tanggal;
    ParamByName('Jumlah').AsFloat:=Jumlah;
    ParamByName('Diskon').AsFloat:=Diskon;
    ParamByName('Kode_Kasa').AsString:=Kode_Kasa;
    ParamByName('ID').AsString:=ID;
    ParamByName('Jenis_Kode').AsString:=Jenis_Kode;
    ParamByName('Jenis_Nama').AsString:=Jenis_Nama;
    ParamByName('Harga_Jual').AsCurrency:=Harga_Jual;
    ParamByName('HPP').AsCurrency:=HPP;
    ExecSQL;
    Close;
  end;
end;

procedure TfmJual.CetakFaktur;
var
  Nama, Alamat,Telp, Fax : string;
begin
  DM.ID_Usaha1(Nama, Alamat,Telp,Fax);
  with fmFakturJual do
  begin
    Judul1.Caption:=Nama;
```

```
    Judul2.Caption:=Alamat;
    QRJudul3.Caption:=Telp+' '+Fax;
    QFaktur.Caption:=ENoFak.Caption;
    QKassa.Caption:=Kassa;
    QKasir.Caption:=Kasir;
    QJenis.Caption:=EJenisBayar.Text;
    QKode.Caption:=ENomorBayar.Text;
    QFakJual.Close;QFakJual.Open;
    QTotJual.Expression:=CurrToStr(DM.Total_Penjualan);
    QRFakJual.Print;
  end;
end;

procedure TfmJual.Isi_Penjualan;
var
  Strook: Longint;
  ID: string;
  KodeKassa: string;
  Jenis_Kode,
  Jenis_Nama : string;
  HPP: currency;
  TotalUnit: real;
begin
  Strook:=StrToInt(ENoFak.Caption);
  if StateEdit then
    EditPenjualan(Strook);

  KodeKassa:=Kassa;
  ID:=Kasir;
  if EJenisBayar.Text<>'' then
    Jenis_Nama:=EJenisBayar.Text
  else
    Jenis_Nama:='Tunai';
  Jenis_Kode:=ENomorBayar.Text;
  with DM,TempJual do
  begin
    Open;
    Refresh;
    First;
    While not EOF do
    begin
      HPP:=HitungHPP(TempJualKode.Value);
      Simpan(Strook,
          TempJualKode.Value,
          DTanggal.Date,
          TempJualBanyak.Value,
          TempJualDiskon.Value,
          KodeKassa,
          ID,
          Jenis_Kode,
          Jenis_Nama,
          TempJualHarga.Value,
          HPP);
      Update_Persediaan(TempJualKode.Value,TempJualBanyak.Value);
      Next;
    end;
  end;
end;

procedure TfmJual.Init;
begin
  QDel.ExecSQL;
  with DSTempJual.DataSet do
  begin
    Append;
    Edit;
  end;
  Clear;
  btnSimpan.Enabled:=False;
end;
```

```
procedure TfmJual.FormActivate(Sender: TObject);
var
  Fld: TField;
  Nama, Alamat: string;
begin
  ActiveControl:=TabelJual;
  StateCancel:=False;
  with DM.TBarang do
  DM.ID_Usaha(Nama, Alamat);
  DM.TJual.Open;
  DSTempJual.DataSet.Open;
  if not StateEdit then
  begin
   if DM.TJual.RecordCount<1 then
    ENofak.Caption:='1'
   else
   begin
    with QStrook do
    begin
     Close;
     Open;
     ENoFak.Caption:=IntToStr(Fields[0].AsInteger);
     Close;
    end;
   end;
  end;
  DsTempJualDataChange(Sender,Fld);
  if not StateEdit then
    Init;
end;

procedure TfmJual.btnSImpanClick(Sender: TObject);
begin
  EBayarExit(Sender);
  if MessageDlg('Proses penjualan ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
  begin
   Isi_Penjualan;
   if not StateEdit then
     CetakFaktur;
   Init;
   FormActivate(Sender);
  end;
  ActiveControl:=TabelJual;
end;

procedure TfmJual.btnBatalClick(Sender: TObject);
var
  Fld: TField;
begin
  StateEdit:=False;
  StateCancel:=True;
  Ebayar.Text:=ETotal.Text;
  Init;
  DsTempJualDataChange(Sender,Fld);
  ActiveControl:=TabelJual;
  FormActivate(Sender);
end;

procedure TfmJual.btnTutupClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmJual.Timer1Timer(Sender: TObject);
begin
  if (Idx mod 2=0) then
    LbTotal.Caption:=Jumlah
  else
    LbTotal.Caption:='';
```

```
  inc(Idx);
end;

procedure TfmJual.EBayarKeyUp(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  btnSimpan.Enabled:=EBayar.Text<>'';
end;

procedure TfmJual.FormKeyPress(Sender: TObject; var Key: Char);
begin
  {if (Key=#13)and(DM.TempJualKode.value='')and
    (TabelJual.SelectedField=DM.TempJualKode) then
  begin
    SelectNext(ActiveControl, True, True);
    Key:=#0;
  end;}
end;

procedure TfmJual.TabelJualExit(Sender: TObject);
var
  Fld: TField;
begin
  if (TotalJual<>0) and (DSTempJual.dataset.recordcount>0)then
  begin
   DsTempJual.DataSet.Edit;
   DsTempJual.DataSet.Post;
   DsTempJualDataChange(Sender,Fld);
  end else
  begin
   DsTempJual.DataSet.Last;
   DsTempJual.DataSet.Edit;
   DsTempJual.DataSet.Append;
  end;
end;

procedure TfmJual.sbCariClick(Sender: TObject);
begin
  with fmLuBarang do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      if not (DSTempJual.DataSet.State in [dsInsert]) then
      DSTempJual.DataSet.Append;
      DSTempJual.Edit;
      DM.TempJualKode.value:=KodeCari;
      DM.TempJualNama.value:=NamaCari;
      DM.TempJualHarga.Value:=Harga;
      DM.TempJualDiskon.Value:=Diskon;
    end;
  end;
end;

procedure TfmJual.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DM.TBarang.Close;
  DSTempJual.DataSet.Close;
  DM.TJual.Close;
end;

procedure TfmJual.EBayarKeyPress(Sender: TObject; var Key: Char);
begin
  if not (Key in['0'..'9',',','.']) then
    Key:=#0;
end;

procedure TfmJual.DsTempJualUpdateData(Sender: TObject);
var
  Saldo,SaldoMinimum,SM: real;
```

```
 begin
   Saldo:=DM.Saldo_Stock(TabelJual.Fields[0].AsString,saldoMinimum,SM);
   if not StateEdit then
   begin
    if (Saldo<TabelJual.Fields[2].AsInteger)and(TabelJual.Fields[0].AsString<>'')then
    begin
      MessageDlg('Stock barang kurang !',mtInformation,[mbOK],0);
      DSTempJual.Dataset.Cancel;
    end;
   end;
 end;

 procedure TfmJual.SpeedButton5Click(Sender: TObject);
 begin
   Application.HelpFile:='CASHREG.HLP';
   Application.HelpContext(11);
 end;

 procedure TfmJual.TabelJualColExit(Sender: TObject);
 begin
   if (TabelJual.SelectedField=DM.TempJualDiskon)then
   begin
     DSTempJual.DataSet.Refresh;
     DSTempJual.DataSet.Append;
     TabelJual.SelectedField:=DM.TempJualKode;
   end else
   if (TabelJual.SelectedField=DM.TempJualKode)and(DM.TempJualKode.Value<>'') then
   begin
    DM.TBarang.Open;
    if DM.TBarang.FindKey([DM.TempJualKode.Value]) then
    begin
      DM.TempJual.Edit;
      DM.TempJualNama.value:=DM.TBarang.FieldByName('Nama_Barang').AsString;
      DM.TempJualHarga.Value:=DM.TBarang.FieldByName('Harga_Jual').AsCurrency;
      DM.TempJualDiskon.Value:=DM.TBarang.FieldByName('Diskon').AsFloat;
      HargaPerUnit:=DM.TBarang.FieldByName('Harga_Jual').AsCurrency;
      JumSaldo:=DM.TBarang.FieldByName('Sisa').AsFloat;
    end else
    begin
     MessageDlg('Kode barang belum ada !',mtInformation,[mbOK],0);
     DM.TempJual.Delete;
    end;
  end;
 end;

 procedure TfmJual.EBayarKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
 var
   Kembali: Currency;
 begin
   if Key=13 then
   begin
     if (EBayar.Text<>'')and(not StateCancel) then
     begin
      Kembali:=StrToCurr(EBayar.Text)-TotalJual;
      if StrToCurr(EBayar.Text)>=TotalJual then
      begin
       EKembali.Text:=FormatCurr('#,#0.00',Kembali);
       Jumlah:=EKembali.Text;
       LbTotal.Repaint;
       LbTotal.Font.Color:=clRed;
       LbTotal.Caption:=EKembali.Text;
       btnSimpanClick(Sender);
      end
     else
      begin
       MessageDlg('Jumlah pembayaran kurang !',mtInformation,[mbOK],0);
       ActiveControl:=EBayar;
      end;
 end else
```

```
        begin
          StateCancel:=True;
          ActiveControl:=TabelJual;
        end;
        end;
      end;

      procedure TfmJual.EBayarExit(Sender: TObject);
      var
        Kembali: Currency;
      begin
        if (EBayar.Text<>'')and(not StateCancel) then
        begin
          Kembali:=StrToCurr(EBayar.Text)-TotalJual;
          if StrToCurr(EBayar.Text)>=TotalJual then
          begin
            EKembali.Text:=FormatCurr('#,#0.00',Kembali);
            Jumlah:=EKembali.Text;
            LbTotal.Repaint;
            LbTotal.Font.Color:=clRed;
            LbTotal.Caption:=EKembali.Text;
          end
          else
          begin
            MessageDlg('Jumlah pembayaran kurang !',mtInformation,[mbOK],0);
            ActiveControl:=EBayar;
          end;
        end;
      end;
      end.

      unit Kassa;

      interface

      uses
        Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
        ExtCtrls, Db, Grids, DBGrids, Buttons, DBCtrls;

      type
        TfmKassa = class(TForm)
          Panel1: TPanel;
          Panel2: TPanel;
          Panel3: TPanel;
          Panel4: TPanel;
          DBGrid1: TDBGrid;
          DSKassa: TDataSource;
          DBNavigator1: TDBNavigator;
          SpeedButton1: TSpeedButton;
          procedure SpeedButton1Click(Sender: TObject);
          procedure FormActivate(Sender: TObject);
          procedure FormClose(Sender: TObject; var Action: TCloseAction);
        private
          { Private declarations }
        public
          { Public declarations }
        end;

      var
        fmKassa: TfmKassa;

      implementation

      uses UDM;

      {$R *.DFM}

      procedure TfmKassa.SpeedButton1Click(Sender: TObject);
      begin
        ModalResult:=mrOK;
```

```
end;

procedure TfmKassa.FormActivate(Sender: TObject);
begin
  DSKassa.DataSet.Open;
end;

procedure TfmKassa.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DSKassa.DataSet.Open;
end;

end.

unit Kriteria;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, Buttons, Db, DBTables;

const
  Operator_Kriteria:array[1..4]of string=('=','<','>','<>');

type
  TfmKriteria = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Panel5: TPanel;
    Kolom1: TCheckBox;
    Kolom2: TCheckBox;
    Kolom3: TCheckBox;
    sbK1: TSpeedButton;
    sbK2: TSpeedButton;
    ENilai1: TEdit;
    ENilai2: TEdit;
    ENilai3: TEdit;
    btnBatal: TSpeedButton;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    sbHapus: TSpeedButton;
    sbK3: TSpeedButton;
    Kolom4: TCheckBox;
    SpeedButton3: TSpeedButton;
    cbKassa: TComboBox;
    QKassa: TQuery;
    procedure sbHapusClick(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure sbK2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure sbK3Click(Sender: TObject);
    procedure Kolom2Click(Sender: TObject);
    procedure Kolom3Click(Sender: TObject);
    procedure sbK1Click(Sender: TObject);
    procedure SpeedButton1Click(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormShow(Sender: TObject);
  private
  public
    i,j : byte;
    Relasi1,Relasi2,Relasi3 : string;
    Op1,Op2,Op3 : string[2];
    StateKriteriaOn.
```

```
    StateKriteriaOff : boolean;
  end;

var
  fmKriteria: TfmKriteria;

implementation

{$R *.DFM}

procedure TfmKriteria.sbHapusClick(Sender: TObject);
begin
  ENilai1.Clear;
  ENilai2.Text:='0';
  ENilai3.Text:='0';
  SBK1.Caption:='=';
  SBK2.Caption:='=';
  SBK3.Caption:='=';
  Kolom1.Checked:=False;
  Kolom2.Checked:=False;
  Kolom3.Checked:=False;
end;

procedure TfmKriteria.btnBatalClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmKriteria.sbK2Click(Sender: TObject);
begin
  if i<4 then
    inc(i);
  sbK2.Caption:=Operator_Kriteria[i];
  if i=4 then
  i:=0;
end;

procedure TfmKriteria.FormCreate(Sender: TObject);
begin
  i:=1;
  j:=1;
end;

procedure TfmKriteria.sbK3Click(Sender: TObject);
begin
  if j<4 then
    inc(j);
  sbK3.Caption:=Operator_Kriteria[j];
  if j=4 then
  j:=0;
  end;

procedure TfmKriteria.Kolom2Click(Sender: TObject);
begin
  if Kolom2.Checked then
    Relasi2:='And'
  else
    Relasi2:='or';
end;

procedure TfmKriteria.Kolom3Click(Sender: TObject);
begin
  if Kolom3.Checked then
    Relasi3:='And'
  else
    Relasi3:='or';
end;

procedure TfmKriteria.sbK1Click(Sender: TObject);
begin
```

```
    Op1:=sbK1.Caption;
  end;

  procedure TfmKriteria.SpeedButton1Click(Sender: TObject);
  begin
    StateKriteriaOn:=True;
    StateKriteriaOff:=False;
    ModalResult:=mrOK;
  end;

  procedure TfmKriteria.SpeedButton2Click(Sender: TObject);
  begin
    StateKriteriaOn:=False;
    StateKriteriaOff:=True;
    ModalResult:=mrOK;
  end;

  procedure TfmKriteria.FormActivate(Sender: TObject);
  begin
   sbHapusClick(Sender);
  end;

  procedure TfmKriteria.FormShow(Sender: TObject);
  begin
    cbKassa.Clear;
    cbKassa.Items.Add('-----semua-----');
    with QKassa do
    begin
      Close;Open;First;
      while not EOF do
      begin
        cbKassa.Items.Add(Fields[0].AsString+'  '+Fields[1].AsString);
        Next;
      end;
    end;
  end;

  end.

  unit KriteJB;

  interface

  uses
   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
   ExtCtrls, StdCtrls, Buttons, Db, DBTables, ComCtrls;

  const
   Operator_Kriteria:array[1..3]of string=('=','<','>');

  type
   TfmKriteriaJB = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Panel5: TPanel;
    Kolom1: TCheckBox;
    Kolom3: TCheckBox;
    sbK1: TSpeedButton;
    btnBatal: TSpeedButton;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    SpeedButton3: TSpeedButton;
    sbK3: TSpeedButton;
    ENilai3: TDateTimePicker;
    ENilai1: TComboBox;
```

```
    Kolom2: TCheckBox;
    sbK2: TSpeedButton;
    ENilai2: TComboBox;
    QLoad: TQuery;
    sbK4: TSpeedButton;
    Kolom4: TCheckBox;
    ENilai4: TEdit;
    procedure SpeedButton3Click(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure sbK3Click(Sender: TObject);
    procedure SpeedButton1Click(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
  private
    procedure Load;
  public
    i,j : byte;
    Relasi1,Relasi2,Relasi3 : string;
    Op1,Op2,Op3 : string[2];
    StateKriteriaOn,
    StateKriteriaOff : boolean;
  end;

var
  fmKriteriaJB: TfmKriteriaJB;

implementation

{$R *.DFM}

procedure TfmKriteriaJB.SpeedButton3Click(Sender: TObject);
begin
  ENilai1.Clear;
  ENilai4.Clear;
  //ENilai3.Clear;
  SBK1.Caption:='=';
  SBK2.Caption:='=';
  SBK3.Caption:='=';
end;

procedure TfmKriteriaJB.Load;
begin
  ENilai1.Clear;
  ENilai2.Clear;
  ENilai4.Clear;
  with QLoad do
  begin
    Close;
    SQL.Clear;
    SQL.Add('Select Kode_Barang From Barang');
    Open;
    First;
    while not EOF do
    begin
      ENilai1.Items.Add(Fields[0].AsString);
      Next;
    end;
    Close;
    SQL.Clear;
    SQL.Add('Select * From Kassa');
    Open;
    First;
    While not EOF do
    begin
      ENilai2.Items.Add(Fields[0].AsString);
      Next;
    end;
    Close;
end;
```

```
  end;

  procedure TfmKriteriaJB.btnBatalClick(Sender: TObject);
  begin
    ModalResult:=mrOK;
  end;

  procedure TfmKriteriaJB.FormCreate(Sender: TObject);
  begin
    i:=1;
    j:=1;
    ENilai3.Date:=Now;
  end;

  procedure TfmKriteriaJB.sbK3Click(Sender: TObject);
  begin
    if j<3 then
      inc(j);
    sbK3.Caption:=Operator_Kriteria[j];
    if j=3 then
    j:=0;
    end;
  procedure TfmKriteriaJB.SpeedButton1Click(Sender: TObject);
  begin
    StateKriteriaOn:=True;
    StateKriteriaOff:=False;
    ModalResult:=mrOK;
  end;

  procedure TfmKriteriaJB.SpeedButton2Click(Sender: TObject);
  begin
    StateKriteriaOn:=False;
    StateKriteriaOff:=True;
    ModalResult:=mrOK;
  end;

  procedure TfmKriteriaJB.FormShow(Sender: TObject);
  begin
    Load;
  end;
  end.

  unit LapAnal;

  interface

  uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    ExtCtrls, quickrpt, Qrctrls, Db, DBTables;

  type
    TfmLapAnalisa = class(TForm)
      QRAnalisa: TQuickRep;
      QRBand1: TQRBand;
      Judul1: TQRLabel;
      Judul2: TQRLabel;
      QRLabel1: TQRLabel;
      ChildBand1: TQRChildBand;
      QRLabel3: TQRLabel;
      QRLabel4: TQRLabel;
      QRLabel5: TQRLabel;
      QAnalisa: TQuery;
      QRLabel7: TQRLabel;
      QRLabel11: TQRLabel;
      Kode: TQRLabel;
      QRLabel12: TQRLabel;
      QRSysData2: TQRSysData;
      QRBand2: TQRBand;
      QRBand3: TQRBand;
      QRDBText1: TQRDBText;
```

```
  QRDBText2: TQRDBText;
  QRDBText3: TQRDBText;
  QRDBText4: TQRDBText;
  QRExpr1: TQRExpr;
  QRExpr2: TQRExpr;
  QRExpr3: TQRExpr;
  QRLabel2: TQRLabel;
  QRLabel6: TQRLabel;
  QRKassa: TQRLabel;
  procedure QRExpr1Print(sender: TObject; var Value: String);
 private
  { Private declarations }
 public
  { Public declarations }
 end;

var
 fmLapAnalisa: TfmLapAnalisa;

implementation
{$R *.DFM}
procedure TfmLapAnalisa.QRExpr1Print(sender: TObject; var Value: String);
begin
 if QAnalisa.RecordCount<1 then
   Value:='0.00';
end;
end.

unit Lap;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, Buttons, ComCtrls, Db, DBTables;

type
 TfmLaporan = class(TForm)
   Panel1: TPanel;
   Panel2: TPanel;
   Panel3: TPanel;
   Panel4: TPanel;
   TujuanCetak: TRadioGroup;
   GroupBox2: TGroupBox;
   Label1: TLabel;
   EKode: TEdit;
   sbCari: TSpeedButton;
   btnBatal: TSpeedButton;
   SpeedButton2: TSpeedButton;
   ENama: TEdit;
   SpeedButton1: TSpeedButton;
   QBeli: TQuery;
   QUpdate: TQuery;
   QJual: TQuery;
   Label3: TLabel;
   cbKassa: TComboBox;
   QKassa: TQuery;
   GroupBox3: TGroupBox;
   DAwal: TDateTimePicker;
   DAkhir: TDateTimePicker;
   Label5: TLabel;
   Label4: TLabel;
   Label6: TLabel;
   EPemasok: TEdit;
   SpeedButton3: TSpeedButton;
   ENamaPemasok: TEdit;
   GroupBox4: TGroupBox;
   rb1: TRadioButton;
   rb2: TRadioButton;
   rb3: TRadioButton;
```

```
    rb4: TRadioButton;
    rb5: TRadioButton;
    rb6: TRadioButton;
    rb7: TRadioButton;
    rb8: TRadioButton;
    cbStockMinimal: TCheckBox;
    procedure btnBatalClick(Sender: TObject);
    procedure sbCariClick(Sender: TObject);
    procedure EKodeKeyDown(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure FormActivate(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure EKodeExit(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure FormShow(Sender: TObject);
    procedure SpeedButton3Click(Sender: TObject);
    procedure EPemasokExit(Sender: TObject);
  private
    procedure CetakLaporan;
    Procedure Hit_Analisa;
    procedure LaporanPenjualan;
    procedure LaporanPembelian;
    procedure LoadKassa;
    procedure LaporanHPP;
    procedure LaporanBarang;
  public
  end;

var
  fmLaporan: TfmLaporan;

implementation

uses LuBarang, LapBrg, LapBeli, LapJual, LapAnal, UDM, Analisa, LapHPP,
  LuVendor, LapVend, LRetbeli, LRetJual;

{$R *.DFM}

procedure TfmLaporan.LoadKassa;
begin
  with QKassa do
  begin
    Close;
    SQL.Clear;
    SQL.Add('Select * From Kassa');
    cbKassa.Clear;
    cbKassa.Items.Add('----semua----');
    Open;
    First;
    while not EOF do
    begin
      cbKassa.Items.Add(Fields[0].AsString+'   '+Fields[1].AsString);
      Next;
    end;
  end;
end;

procedure TfmLaporan.LaporanPenjualan;
var
  WhereUsed: boolean;

  function SQLPrefix:string;
  begin
    if WhereUsed then
      Result:='and '
    else
      begin
        WhereUsed:=True;
        Result:='Where ';
```

```
        if Length(EPemasok.Text)>0 then
        begin
          SQL.Add(SQLPrefix+'Beli.Pemasok=:P4');
          ParamByName('P4').AsString:=EPemasok.Text;
          Open;
        end;
      end;
    end;
  end;

procedure TfmLaporan.LaporanHPP;
var
  WhereUsed: boolean;

 begin
  WhereUsed:=False;
  with fmLapHPP do
  begin
    with QHPP do
    begin
      Close;
      SQL.Clear;
      SQL.Add('Select Strook,Tgl,(Jumlah*Harga_Jual) As Penjualan,(Jumlah*HPP) As JumHPP,');
      SQL.Add('(Jumlah*Harga_Jual)-(Jumlah*HPP) As Saldo');
      SQL.Add('From Jual');
      Open;
      if EKode.Text<>'' then
      begin
        SQL.Add(SQLPrefix+'Kode_Barang=:P1');
        ParamByName('P1').AsString:=EKode.Text;
        Open;
      end;
      if (DAwal.Checked)and(DAkhir.Checked) then
      begin
        SQL.Add(SQLPrefix+'Tgl Between :P2 and :P3');
        ParamByName('P2').AsDate:=DAwal.Date;
        ParamByName('P3').AsDate:=DAkhir.Date;
        Open;
      end;
      if (cbKassa.ItemIndex>0) then
      begin
        SQL.Add(SQLPrefix+'Kode_Kasa=:P4');
        ParamByName('P4').AsString:=copy(cbKassa.Text,1,3);
        Open;
      end;
      SQL.Add('Order By Tgl');
      Open;
    end;
  end;
 end;

 Procedure TfmLaporan.Hit_Analisa;
 var
   i : byte;
 begin
   fmAnalisa.CekTabelAnalisa;
   //analisa penjualan
   with QBeli do
   begin
    for i:=1 to 12 do
    begin
      Close;
      SQL.Clear;
      SQL.Add('Select Sum(Jumlah*HPP) From  Jual');
      SQL.Add('Where Extract(Month From Tgl)=:P1');
      Close;
      ParamByName('P1').AsInteger:=i;
      if (cbKassa.ItemIndex>0) then
      begin
        SQL.Add('And Kode_Kasa=:P2');
```

```
      ParamByName('P2').AsString:=copy(cbKassa.Text,1,3);
    end;
    if EKode.Text<>'' then
    begin
     SQL.Add('And Kode_Barang=:P3');
     ParamByName('P3').AsString:=EKode.Text;
    end;
    Open;
    with QUpdate do
    begin
     Close;
     SQL.Clear;
     SQL.Add('Update Analisa Set Beli=:p1 Where Bulan=:P2');
     Params[0].AsCurrency:=QBeli.Fields[0].AsCurrency;
     Params[1].AsInteger:=i;
     ExecSQL;
    end;
   end;
  end;
  with QJual do
  begin
   for i:=1 to 12 do
   begin
     Close;
     SQL.Clear;
     SQL.Add('Select Sum((Jumlah*Harga_Jual)-((Jumlah*Harga_Jual)*(Diskon/100))) From  Jual');
     SQL.Add('Where Extract(Month From Tgl )=:P1');
     Close;
     ParamByName('P1').AsInteger:=i;
     if (cbKassa.ItemIndex>0)then
     begin
       SQL.Add('And Kode_Kasa=:P2');
       ParamByName('P2').AsString:=copy(cbKassa.Text,1,3);
     end;
     if EKode.Text<>'' then
     begin
       SQL.Add('And Kode_Barang=:P3');
       ParamByName('P3').AsString:=EKode.Text;
     end;
     Open;
     with QUpdate do
     begin
       Close;
       SQL.Clear;
       SQL.Add('Update Analisa Set Jual=:p1 Where Bulan=:P2');
       Params[0].AsCurrency:=QJual.Fields[0].AsCurrency;
       Params[1].AsInteger:=i;
       ExecSQL;
     end;
    end;
    with QUpdate do
    begin
     for i:=1 to 12 do
     begin
       Close;
       SQL.Clear;
       SQL.Add('Update Analisa Set Laba=Jual-Beli');
       SQL.Add('Where Bulan=:P1');
       Params[0].AsInteger:=i;
       ExecSQL;
     end;
    end;
  end;

procedure TfmLaporan.CetakLaporan;
var
  Nama, ALamat : string;
begin
  DM.ID_Usaha(Nama,Alamat);
```

```
        begin
          Awal.Caption:=DateToStr(DAwal.Date);
          Akhir.Caption:=DateToStr(DAkhir.Date);;
        end;
        QRKassa.Caption:=cbKassa.Text;

        Judul1.Caption:=Nama;
        Judul2.Caption:=Alamat;
        case TujuanCetak.ItemIndex of
          0 : QRJual.Preview;
          1 : QRJual.Print;
        end;
      end;
  end else
  if rb5.Checked then
  begin
      with fmLapAnalisa do
      begin
        if (EKode.Text='') then
          Kode.Caption:='~'
        else
          Kode.Caption:=EKode.Text+', '+ENama.Text;
        try
          Screen.Cursor:=crHourGlass;
          Hit_Analisa;
        finally
          Screen.Cursor:=crDefault;
        end;
        QAnalisa.Close;
        QAnalisa.Open;

        QRKassa.Caption:=cbKassa.Text;
        Judul1.Caption:=Nama;
        Judul2.Caption:=Alamat;
        case TujuanCetak.ItemIndex of
          0 : QRAnalisa.Preview;
          1 : QRAnalisa.Print;
        end;
      end;
  end else
  if rb6.Checked then
  begin
      LaporanHPP;
      with fmLapHPP do
      begin
        if EKode.Text='' then
          Kode.Caption:='~' else
          Kode.Caption:=EKode.Text;
        if (not DAwal.Checked)and (not DAkhir.Checked) then
          QRPeriode.Caption:='~'
        else
          QRPeriode.Caption:=DateToStr(DAwal.Date)+' s.d. '+
          DateToStr(DAkhir.Date);

        QRKassa.Caption:=cbKassa.Text;

        Judul1.Caption:=Nama;
        Judul2.Caption:=Alamat;
        case TujuanCetak.ItemIndex of
          0 : QRHPP.Preview;
          1 : QRHPP.Print;
        end;
      end;
    end else
    if rb7.Checked then
    begin
      with fmLapReturBeli do
      begin
        QReturBeli.Close;
        QReturBeli.SQL.Clear;
```

```
          QReturBeli.SQL.Add('Select * From Retbeli, Barang');
          QReturBeli.SQL.Add('Where RetBeli.KodeBarang=Barang.Kode_Barang');

          if (not DAwal.Checked)and (not DAkhir.Checked) then
            QRPeriode.Caption:='~'
          else
          begin
            QReturBeli.SQL.Add('and RetBeli.Tanggal Between :P1 and :P2');
            QReturBeli.Params[0].AsDate:=DAwal.Date;
            QReturBeli.Params[1].AsDate:=DAkhir.Date;
            QRPeriode.Caption:=DateToStr(DAwal.Date)+' s.d. '+
            DateToStr(DAkhir.Date);
          end;
          QReturBeli.Open;
          Judul1.Caption:=Nama;
          Judul2.Caption:=Alamat;
          case TujuanCetak.ItemIndex of
            0 : QRRetBeli.Preview;
            1 : QRRetbeli.Print;
          end;
        end;
      end else
      if rb8.Checked then
      begin
        with fmLapReturJual do
        begin
          QReturJual.Close;
          QReturJual.SQL.Clear;
          QReturJual.SQL.Add('Select * From RetJual, Barang');
          QReturJual.SQL.Add('Where RetJual.KodeBarang=Barang.Kode_Barang');

          if (not DAwal.Checked)and (not DAkhir.Checked) then
            QRPeriode.Caption:='~'
          else
          QReturJual.Open;
          Judul1.Caption:=Nama;
          Judul2.Caption:=Alamat;
          case TujuanCetak.ItemIndex of
            0 : QRRetJual.Preview;
            1 : QRRetJual.Print;
          end;
        end;
    end;
end;

procedure TfmLaporan.btnBatalClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmLaporan.sbCariClick(Sender: TObject);
begin
  with fmLuBarang do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      Ekode.Text:=KodeCari;
      ENama.Text:=NamaCari;
    end;
  end;
end;

procedure TfmLaporan.EKodeKeyDown(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if Key=VK_DOWN then
    sbCariClick(Sender);
end;
```

```
procedure TfmLaporan.FormActivate(Sender: TObject);
begin
  DM.TBarang.Open;
end;

procedure TfmLaporan.SpeedButton2Click(Sender: TObject);
begin
  CetakLaporan;
end;

procedure TfmLaporan.FormCreate(Sender: TObject);
begin
  DAwal.Date:=Now;
  DAkhir.Date:=Now;
  DAwal.Checked:=False;
  DAkhir.Checked:=False;
end;

procedure TfmLaporan.EKodeExit(Sender: TObject);
begin
 if Length(EKode.Text)>0 then
 begin
  if DM.TBarang.FindKey([EKode.Text]) then
    ENama.Text:=DM.TBarang.FieldByName('Nama_Barang').AsString
  else
  begin
    MessageDlg('Data barang belum ada !',mtInformation,[mbOK],0);
    ActiveControl:=EKode;
  end;
 end;
end;

procedure TfmLaporan.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DM.TBarang.Close;
end;

procedure TfmLaporan.FormShow(Sender: TObject);
begin
  EKode.Clear;
  ENama.Clear;
  LoadKassa;
end;



procedure TfmLaporan.SpeedButton3Click(Sender: TObject);
begin
  with fmLuVendor do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      EPemasok.Text:=KodeVendorCari;
      EPemasokExit(Sender);
    end;
  end;
end;

procedure TfmLaporan.EPemasokExit(Sender: TObject);
begin
 if Length(EPemasok.Text)>0 then
 begin
  DM.TVendor.Filtered:=False;
  DM.TVendor.IndexName:='';
  if DM.TVendor.FindKey([EPemasok.Text]) then
   ENamaPemasok.Text:=DM.TVendor.FieldByName('Nama').AsString
  else
  begin
    MessageDlg('Pemasok belum ada !',mtInformation,[mbOK],0);
```

```
    ActiveControl:=EPemasok;
  end;
 end;
end;

unit LuBarang;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, Db, Grids, DBGrids, ComCtrls, StdCtrls, DBTables, Buttons;

type
 TfmLuBarang = class(TForm)
   Panel1: TPanel;
   Panel2: TPanel;
   Panel4: TPanel;
   GridBarang: TDBGrid;
   DsBarang: TDataSource;
   QDeskripsi: TQuery;
   Label1: TLabel;
   EKodeCari: TEdit;
   btnPilih: TBitBtn;
   procedure btnPilihClick(Sender: TObject);
   procedure GridBarangKeyDown(Sender: TObject; var Key: Word;
    Shift: TShiftState);
   procedure btnBatalClick(Sender: TObject);
   procedure FormActivate(Sender: TObject);
   procedure EKodeCariKeyUp(Sender: TObject; var Key: Word;
    Shift: TShiftState);
 private
  { Private declarations }
 public
   KodeCari,NamaCari : String;
   Harga : Currency;
   Diskon : real;
 end;

var
 fmLuBarang: TfmLuBarang;

implementation

uses UDM;

{$R *.DFM}

procedure TfmLuBarang.btnPilihClick(Sender: TObject);
begin
  KodeCari:=GridBarang.Fields[0].AsString;
  with QDeskripsi do
  begin
    Close;
    Params[0].AsString:=KodeCari;
    Open;
    NamaCari:=Fields[0].AsString;
    Harga:=Fields[1].AsCurrency;
    Diskon:=Fields[2].AsFloat;
    Close;
  end;
  ModalResult:=mrOK;
end;

procedure TfmLuBarang.GridBarangKeyDown(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  if Key=13 then
    btnPilihClick(Sender);
end;

procedure TfmLuBarang.btnBatalClick(Sender: TObject);
```

```
begin
  Close;
end;

procedure TfmLuBarang.FormActivate(Sender: TObject);
begin
  DSBarang.DataSet.Open;
  DSBarang.DataSet.Refresh;
  ActiveControl:=GridBarang;
end;

procedure TfmLuBarang.EKodeCariKeyUp(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  if Length(EKodeCari.Text)>0 then
    DM.TBarang.FindNearest([EKodeCari.Text]);
end;
end.

unit LuVendor;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, Db, Grids, DBGrids, ComCtrls, StdCtrls, DBTables, Buttons;

type
 TfmLuVendor = class(TForm)
  Panel1: TPanel;
  Panel4: TPanel;
  TabelVendor: TDBGrid;
  DSVendor: TDataSource;
  Panel2: TPanel;
  Label1: TLabel;
  EKodeCari: TEdit;
  btnPilih: TBitBtn;
  procedure btnPilihClick(Sender: TObject);
  procedure TabelVendorKeyDown(Sender: TObject; var Key: Word;
   Shift: TShiftState);
  procedure FormActivate(Sender: TObject);
  procedure EKodeCariKeyUp(Sender: TObject; var Key: Word;
   Shift: TShiftState);
 private
  { Private declarations }
 public
  KodeVendorCari: String;
 end;

var
 fmLuVendor: TfmLuVendor;

implementation

uses UDM;

{$R *.DFM}

procedure TfmLuVendor.btnPilihClick(Sender: TObject);
begin
  KodeVendorCari:=TabelVendor.Fields[0].AsString;
  ModalResult:=mrOK;
end;

procedure TfmLuVendor.TabelVendorKeyDown(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  if Key=13 then
    btnPilihClick(Sender);
end;

procedure TfmLuVendor.FormActivate(Sender: TObject);
```

```
begin
  DsVendor.DataSet.Open;
  DsVendor.DataSet.Refresh;
  ActiveControl:=TabelVendor;
end;
end.

unit Graph;
interface
uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 StdCtrls, ExtCtrls, Buttons;

type
 TfmPilihanGrafik = class(TForm)
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  btnBatal: TSpeedButton;
  sbBar: TSpeedButton;
  sbLine: TSpeedButton;
  GroupBox1: TGroupBox;
  Pilihan1: TCheckBox;
  Pilihan2: TCheckBox;
  Pilihan3: TCheckBox;
  procedure sbBarClick(Sender: TObject);
  procedure sbLineClick(Sender: TObject);
  procedure btnBatalClick(Sender: TObject);
 private
 public

 end;

var
 fmPilihanGrafik: TfmPilihanGrafik;

implementation

uses Grafik;

{$R *.DFM}

procedure TfmPilihanGrafik.sbBarClick(Sender: TObject);
begin
  if (Pilihan1.Checked) or (Pilihan2.Checked) or (Pilihan3.Checked) then
  begin
   with fmGrafik do
   begin
    ShowGraph(0);
    ShowModal;
   end;
  end else
   MessageDlg('Pilih dulu jenis transaksi !',mtInformation,[mbOK],0);
end;

procedure TfmPilihanGrafik.sbLineClick(Sender: TObject);
begin
  if (Pilihan1.Checked) or (Pilihan2.Checked) or (Pilihan3.Checked) then
  begin
   with fmGrafik do
   begin
    ShowGraph(1);
    ShowModal;
   end;
  end else
   MessageDlg('Pilih dulu jenis transaksi !',mtInformation,[mbOK],0);
end;

procedure TfmPilihanGrafik.btnBatalClick(Sender: TObject);
```

```
 begin
   ModalResult:=mrOK;
 end;
 end.

 unit RetBeli;

 interface
 uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Buttons, ExtCtrls, ComCtrls, StdCtrls, Grids, DBGrids, Db, DBTables,
  Menus;

 type
  TfmRetBeli = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    SpeedButton1: TSpeedButton;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    ENoBuktiRetur: TEdit;
    ETanggal: TDateTimePicker;
    GroupBox2: TGroupBox;
    TabelBeli: TDBGrid;
    Label3: TLabel;
    ENoBuktiBeli: TEdit;
    DSBeli: TDataSource;
    Label4: TLabel;
    EJumlahRetur: TEdit;
    Panel3: TPanel;
    btnSimpan: TBitBtn;
    btnBatal: TBitBtn;
    QNoBukti: TQuery;
    QUpdateStock: TQuery;
    BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    PopupMenu1: TPopupMenu;
    HapusRetur1: TMenuItem;
    QSelect: TQuery;
    QDel: TQuery;
    procedure ENoBuktiBeliChange(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure ENoBuktiReturKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure ENoBuktiReturChange(Sender: TObject);
    procedure btnSimpanClick(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure HapusRetur1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

 var
  fmRetBeli: TfmRetBeli;

 implementation

 uses UDM;

 {$R *.DFM}

 procedure TfmRetBeli.ENoBuktiBeliChange(Sender: TObject);
 begin
   if Length(ENoBuktiBeli.Text)>0 then
     DSBeli.DataSet.Filter:='Strook='''+ENoBuktiBeli.Text+''''
```

```
  else
    DSBeli.DataSet.Filter:='Strook="*"';
  DSBeli.DataSet.Filtered:=True;
end;

procedure TfmRetBeli.ENoBuktiReturKeyUp(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  btnSimpan.Enabled:=Length(ENoBuktiRetur.Text)>0;
end;

procedure TfmRetBeli.ENoBuktiReturChange(Sender: TObject);
begin
  if Length(ENoBuktiRetur.Text)>0 then
  begin
    with QNoBukti do
    begin
      Close;
      Params[0].AsString:=ENoBuktiRetur.Text;
      Open;
      if Fields[0].AsInteger>0 then
      begin
        MessageDlg('No. Bukti sudah digunakan !,ganti dengan yang lain',mtInformation,[mbOK],0);
        ENoBuktiRetur.Clear;
        ActiveControl:=ENoBuktiRetur;
      end else
        btnSimpan.Enabled:=False;
    end;
  end;
end;

procedure TfmRetBeli.btnSimpanClick(Sender: TObject);
begin
  if MessageDlg('Data retur sudah benar ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
  begin
    with DM.TReturBeli do
    begin
      if not active then Open;
      Append;
      FieldByName('NoBukti').AsString:=ENoBuktiRetur.Text;
      FieldByname('Tanggal').AsDateTime:=ETanggal.Date;
      FieldByName('NoFakbeli').AsString:=ENoBuktiBeli.Text;
      FieldByName('KodeBarang').AsString:=TabelBeli.Fields[0].AsString;
      FieldByName('Jumlah').AsInteger:=StrToInt(EJumlahRetur.Text);
      FieldByName('Harga').AsCurrency:=TabelBeli.Fields[3].AsCurrency;
      Post;
    end;
    with QUpdateStock do
    begin
      Close;
      Params[0].AsInteger:=StrToInt(EJumlahRetur.Text);
      Params[1].AsString:=TabelBeli.Fields[0].AsString;
      ExecSQL;
    end;
    EJumlahRetur.Text:='0';
  end;
end;

procedure TfmRetBeli.btnBatalClick(Sender: TObject);
begin
  FormActivate(Sender);
end;

procedure TfmRetBeli.BitBtn1Click(Sender: TObject);
begin
 Close;
end;

procedure TfmRetBeli.BitBtn2Click(Sender: TObject);
begin
```

```
  FormActivate(Sender);
end;

procedure TfmRetBeli.HapusRetur1Click(Sender: TObject);
var
  Jumlah: integer;
  KodeBrg,s: string;
begin
  with QSelect do
  begin
    s:=INputBox('Masukan kode retur pembelian ','Kode: ','');
    if Length(s)>0 then
    begin
      Close;
      Params[0].asString:=s;
      Open;
      Jumlah:=Fields[1].AsInteger;
      KodeBrg:=Fields[0].asstring;
      if RecordCount>0 then
      begin
        with QUpdateStock do
        begin
          Close;
          Params[0].AsInteger:=Jumlah*-1;
          Params[1].AsString:=KodeBrg;
          ExecSQL;
        end;
        QDel.Close;
        QDel.Params[0].AsString:=s;
        QDel.execSQL;
      end else
      MessageDlg('Kode retur belum ada ',mtInformation,[mbOK],0);
    end;

  end;
end;
end.

unit RetJual;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Buttons, ExtCtrls, ComCtrls, StdCtrls, Grids, DBGrids, Db, DBTables,
  Menus;

type
  TfmRetJual = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    SpeedButton1: TSpeedButton;
    GB: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    ENoBuktiRetur: TEdit;
    ETanggal: TDateTimePicker;
    GroupBox2: TGroupBox;
    TabelJual: TDBGrid;
    Label3: TLabel;
    ENoBuktiBeli: TEdit;
    DSJual: TDataSource;
    Label4: TLabel;
    EJumlahRetur: TEdit;
    Panel3: TPanel;
    btnSimpan: TBitBtn;
    btnBatal: TBitBtn;
    QNoBukti: TQuery;
    QUpdateStock: TQuery;
    BitBtn1: TBitBtn;
```

```
    BitBtn2: TBitBtn;
    PopupMenu1: TPopupMenu;
    HapusRetur1: TMenuItem;
    QSelect: TQuery;
    QDel: TQuery;
    procedure ENoBuktiBeliChange(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure ENoBuktiReturKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure ENoBuktiReturChange(Sender: TObject);
    procedure btnSimpanClick(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure HapusRetur1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  fmRetJual: TfmRetJual;

implementation

uses UDM;

{$R *.DFM}

procedure TfmRetJual.ENoBuktiBeliChange(Sender: TObject);
begin
  if Length(ENoBuktiBeli.Text)>0 then
    DSJual.DataSet.Filter:='Strook='''+ENoBuktiBeli.Text+''''
  else
    DSJual.DataSet.Filter:='Strook=0';
  DSJual.DataSet.Filtered:=True;
end;

procedure TfmRetJual.FormActivate(Sender: TObject);
begin
  btnSimpan.Enabled:=False;
  ENoBuktiRetur.Clear;
  ENoBuktiBeli.Clear;
  EJumlahRetur.Text:='0';
  ENoBuktiBeliChange(Sender);
  ActiveControl:=ENoBuktiRetur;
  DSJual.DataSet.Open;
  DSJual.DataSet.Refresh;
end;

procedure TfmRetJual.ENoBuktiReturKeyUp(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  btnSimpan.Enabled:=Length(ENoBuktiRetur.Text)>0;
end;

procedure TfmRetJual.ENoBuktiReturChange(Sender: TObject);
begin
  if Length(ENoBuktiRetur.Text)>0 then
  begin
    with QNoBukti do
    begin
      Close;
      Params[0].AsString:=ENoBuktiRetur.Text;
      Open;
      if Fields[0].AsInteger>0 then
      begin
        MessageDlg('No. Bukti sudah digunakan !,ganti dengan yang lain',mtInformation,[mbOK],0);
        ENoBuktiRetur.Clear;
```

```
        ActiveControl:=ENoBuktiRetur;
      end else
        btnSimpan.Enabled:=False;
    end;
  end;
end;

procedure TfmRetJual.btnSimpanClick(Sender: TObject);
begin
  if MessageDlg('Data retur sudah benar ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
  begin
    with DM.TReturJual do
    begin
      if not active then Open;
      Append;
      FieldByName('NoBukti').AsString:=EnoBuktiRetur.Text;
      FieldByname('Tanggal').AsDateTime:=ETanggal.Date;
      FieldByName('NoFakJual').AsString:=ENoBuktiBeli.Text;
      FieldByName('KodeBarang').AsString:=TabelJual.Fields[0].AsString;
      FieldByName('Kassa').AsString:=TabelJual.Fields[3].AsString;
      FieldByName('Jumlah').AsInteger:=StrToInt(EJumlahRetur.Text);
      FieldByName('Harga').AsCurrency:=TabelJual.Fields[3].AsCurrency;
      Post;
    end;
    with QUpdateStock do
    begin
      Close;
      Params[0].AsInteger:=StrToInt(EJumlahRetur.Text);
      Params[1].AsString:=TabelJual.Fields[0].AsString;
      ExecSQL;
    end;
    EJumlahRetur.Text:='0';
  end;
end;

procedure TfmRetJual.btnBatalClick(Sender: TObject);
begin
  FormActivate(Sender);
end;

procedure TfmRetJual.BitBtn1Click(Sender: TObject);
begin
  Close;
end;

procedure TfmRetJual.BitBtn2Click(Sender: TObject);
begin
  FormActivate(Sender);
end;

procedure TfmRetJual.HapusRetur1Click(Sender: TObject);
var
  Jumlah: integer;
  s,KodeBrg: string;
begin
  gb.Enabled:=False;
  with QSelect do
  begin
    s:=INputBox('Masukan kode retur penjualan ','Kode: ','');
    if Length(s)>0 then
    begin
      Close;
      Params[0].asString:=s;
      Open;
      Jumlah:=Fields[1].AsInteger;
      KodeBrg:=Fields[0].asstring;
      if RecordCount>0 then
      begin
        with QUpdateStock do
        begin
```

```
            Close;
            Params[0].AsInteger:=Jumlah*-1;
            Params[1].AsString:=KodeBrg;
            ExecSQL;
          end;
          QDel.Close;
          QDel.Params[0].AsString:=s;
          QDel.execSQL;
        end else
        MessageDlg('Kode retur belum ada ',mtInformation,[mbOK],0);
      end;
    end;
    gb.Enabled:=True
  end;
  end.

  unit Tanggal;

  interface

  uses
   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
   Buttons, ComCtrls, StdCtrls, ExtCtrls ;

  type
   TfmTanggal = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    GroupBox1: TGroupBox;
    TanggalSekarang: TDateTimePicker;
    button2: TSpeedButton;
    btnBatal: TSpeedButton;
    procedure btnBatalClick(Sender: TObject);
    procedure button2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
   private
    { Private declarations }
   public
    { Public declarations }
   end;

  var
   fmTanggal: TfmTanggal;

  implementation

  {$R *.DFM}

  procedure TfmTanggal.btnBatalClick(Sender: TObject);
  begin
    ModalResult:=mrOK;
  end;

  procedure TfmTanggal.FormCreate(Sender: TObject);
  begin
    TanggalSekarang.Date:=Now;
  end;

  end.

  unit TBeli;

  interface

  uses
   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
   ExtCtrls, Db, Grids, DBGrids, ComCtrls, Buttons, StdCtrls, DBTables;
```

```
type
 TfmTBeli = class(TForm)
   Panel1: TPanel;
   Panel2: TPanel;
   Panel4: TPanel;
   Dsbeli: TDataSource;
   SpeedButton1: TSpeedButton;
   QKriteria: TQuery;
   QTotal: TQuery;
   QDafBeli: TQuery;
   Panel5: TPanel;
   SpeedButton2: TSpeedButton;
   btnTutup: TSpeedButton;
   GroupBox2: TGroupBox;
   Label1: TLabel;
   Label2: TLabel;
   Label3: TLabel;
   lbKodebarang: TLabel;
   lbUser: TLabel;
   lbtanggal: TLabel;
   SpeedButton3: TSpeedButton;
   SpeedButton4: TSpeedButton;
   QItemPembelian: TQuery;
   TabelBarangBeli: TDBGrid;
   btnHapusTransaksi: TSpeedButton;
   QUpdate: TQuery;
   QDelete: TQuery;
   procedure FormActivate(Sender: TObject);
   procedure FormClose(Sender: TObject; var Action: TCloseAction);
   procedure SpeedButton1Click(Sender: TObject);
   procedure SpeedButton2Click(Sender: TObject);
   procedure DsbeliDataChange(Sender: TObject; Field: TField);
   procedure TabelItemBeliDblClick(Sender: TObject);
   procedure TabelBarangBeliDblClick(Sender: TObject);
   procedure btnHapusTransaksiClick(Sender: TObject);
 private
   procedure Tampilkan;
   procedure Update_Persediaan(Kode: string;Jumlah: real);
   procedure CekStatusTombolHapus;
 public
   { Public declarations }
 end;
var
 fmTBeli: TfmTBeli;

implementation

uses KriteJB, UDM, EditBeli, Beli;

{$R *.DFM}

procedure TfmTBeli.CekStatusTombolHapus;
begin
  btnHapusTransaksi.Enabled:=DSBeli.DataSet.RecordCount>0;
end;

procedure TfmTBeli.Update_Persediaan(Kode: string;Jumlah: real);
begin
  with QUpdate do
  begin
    Close;
    ParamByName('P1').AsFloat:=Jumlah;
    ParamByName('P2').AsString:=Kode;
    ExecSQL;
  end;
end;

procedure TfmTBeli.FormActivate(Sender: TObject);
begin
  with DSBeli.DataSet do
```

```
  begin
    Close;
    Open;
    Refresh;
  end;
  CekStatusTombolHapus;
end;

procedure TfmTBeli.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DSBeli.DataSet.Close;
end;

procedure TfmTBeli.SpeedButton1Click(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmTBeli.TampilKan;
var
  WhereUsed: boolean;

begin
  with fmKriteriaJB do
  begin
    Kolom4.Enabled:=False;
    ENilai4.Enabled:=False;
    ShowModal;
    if ModalResult=mrOK then
    begin
    if StateKriteriaON then
    begin
      with QKriteria do
      begin
        Close;
        SQL.Clear;
        SQL.Add('Select Distinct Strook,Tanggal,Jumlah*Harga As JumlahBeli, ID');
        SQL.Add('From Beli');
        Open;
        if Kolom1.Checked then
        begin
          SQL.Add(SQLPrefix+'Kode_Barang=:P1');
          ParamByName('P1').AsString:=ENilai1.Text;
          Open;
          lbKodeBarang.Caption:=ENilai1.Text;
        end else lbKodeBarang.Caption:='';
        if Kolom2.Checked then
        begin
          SQL.Add(SQLPrefix+'ID=:P2');
          ParamByName('P2').AsString:=ENilai2.Text;
          Open;
          lbUser.Caption:=ENilai2.Text;
        end else lbUser.Caption:='';
        if Kolom3.Checked then
        begin
          SQL.Add(SQLPrefix+'Tanggal'+SBK3.Caption+':P3');
          ParamByName('P3').AsDate:=ENilai3.Date;
          Open;
          lbTanggal.Caption:=DateToStr(ENilai3.Date);
        end else lbTanggal.Caption:='';
        SQL.Add('Order By Tanggal');
        Open;
      end;
      DSBeli.DataSet:=QKriteria;
    end else
      DSBeli.DataSet:=DM.QDafBeli;
  end;
  end;
  DSBeli.DataSet.Refresh;
end;
```

```
procedure TfmTBeli.SpeedButton2Click(Sender: TObject);
begin
  Tampilkan;
end;

procedure TfmTBeli.DsbeliDataChange(Sender: TObject; Field: TField);
begin
  with QItemPembelian do
  begin
    Close;
    ParamByName('P1').AsString:='001';
    Open;
  end;
end;

procedure TfmTBeli.TabelItemBeliDblClick(Sender: TObject);
begin
 with fmEditBeli do
  begin
    with QDafBeli do
    begin
      Close;
      ParamByName('Faktur').AsString:=TabelBarangBeli.Fields[0].AsString;
      ParamByName('Kode').AsString:=TabelBarangBeli.Fields[1].AsString;
      Open;
      JumlahBeliAsal:=FieldByName('Jumlah').AsInteger;
      ENoFaktur.Text:=FieldByName('Strook').AsString;
      EKodeBarang.Text:=FieldByName('Kode_Barang').AsString;
      ETanggal.Date:=FieldByName('Tanggal').AsDateTime;
      EJumlah.Text:=IntToStr(FieldByName('Jumlah').AsInteger);
      EHarga.Text:=CurrToStr(FieldByName('Harga').AsCurrency);
      EID.Text:=FieldByName('ID').AsString;
    end;
    EKodeBarangExit(Sender);
    ShowModal;
    DSBeli.DataSet.Close;
    DSBeli.DataSet.Open;
  end;
end;


procedure TfmTBeli.TabelBarangBeliDblClick(Sender: TObject);
begin
  if Length(TabelBarangBeli.Fields[0].AsString)>0 then
  begin
   with fmBeli do
   begin
    StateEdit:=True;
    ENoFak.Text:=TabelBarangBeli.Fields[0].AsString;
    DTanggal.Date:=TabelBarangBeli.Fields[1].AsDateTime;
    EPemasok.Text:=TabelbarangBeli.Fields[4].AsString;
    EKasir.Caption:=TabelbarangBeli.Fields[3].AsString;
    DM.TempBeli.Open;
    with QItemPembelian do
    begin
      Params[0].AsString:=Tabelbarangbeli.Fields[0].AsString;
      begin
        DM.TempBeli.Append;
        DM.TempBeli.Post;
        Next;
      end;
    end;
   ShowModal;
   DSBeli.DataSet.Close;
   DSBeli.DataSet.Open;
   end;
  end;
end;
```

```
procedure TfmTBeli.btnHapusTransaksiClick(Sender: TObject);
begin
  if MessageDlg('Yakin transaksi ini akan dihapus ?',mtConfirmation,[mbYes,mbNO],0)=mrYes then
  begin
    with QItemPembelian do
    begin
      Params[0].AsString:=TabelBarangBeli.Fields[0].AsString;
      Open;First;
      while not EOF do
      begin
        Update_Persediaan(Fields[1].AsString,Fields[3].AsFloat);
        Next;
      end;
    end;
    with QDelete do
    begin
      Close;
      Params[0].AsString:=TabelBarangBeli.Fields[0].AsString;
      ExecSQL;
    end;
    CekStatusTombolHapus;
    DSBeli.DataSet.Close;
    DSBeli.DataSet.Open;
  end;
end;

end.
unit TJual;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, Db, Grids, DBGrids, ComCtrls, Buttons, DBTables, StdCtrls;

type
  TfmTJual = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel4: TPanel;
    TabelBarang: TDBGrid;
    DsJual: TDataSource;
    QKriteria: TQuery;
    SpeedButton1: TSpeedButton;
    GroupBox1: TGroupBox;
    DBGrid1: TDBGrid;
    DSItemPenjualan: TDataSource;
    QItemPenjualan: TQuery;
    Panel5: TPanel;
    SpeedButton2: TSpeedButton;
    btnTutup: TSpeedButton;
    GroupBox2: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    lbKodebarang: TLabel;
    lbKodekassa: TLabel;
    lbtanggal: TLabel;
    QInsertItemTrans: TQuery;
    btnHapusTransaksi: TSpeedButton;
    QDelete: TQuery;
    QUpdate: TQuery;
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure btnTutupClick(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure DsJualDataChange(Sender: TObject; Field: TField);
    procedure TabelBarangDblClick(Sender: TObject);
    procedure btnHapusTransaksiClick(Sender: TObject);
  private
```

```
    procedure Tampilkan;
    procedure Update_Persediaan(Kode: string;Jumlah: real);
    procedure CekStatusTombolHapus;
  public
  end;

var
  fmTJual: TfmTJual;

implementation

uses detil, UDM, KriteJB, Jual;
{$R *.DFM}

procedure TfmTJual.CekStatusTombolHapus;
begin
  btnHapusTransaksi.Enabled:=DSJual.DataSet.RecordCount>0;
end;

procedure TfmTJual.Update_Persediaan(Kode: string;Jumlah: real);
begin
  with QUpdate do
  begin
    Close;
    ParamByName('P1').AsFloat:=Jumlah;
    ParamByName('P2').AsString:=Kode;
    ExecSQL;
  end;
end;

procedure TfmTJual.FormActivate(Sender: TObject);
begin
  with DSJual.DataSet do
  begin
   Close;
   Open;
   Refresh;
  end;
  CekStatusTombolHapus;
end;

procedure TfmTJual.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DSJual.DataSet.Close;
end;

procedure TfmTJual.btnTutupClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmTJual.TampilKan;
var
  WhereUsed: boolean;

  function SQLPrefix:string;
  begin
    if WhereUsed then
      Result:='and '
    else
    begin
      WhereUsed:=True;
      Result:='Where ';
    end;
  end;

begin
  WhereUsed:=False;
  with fmKriteriaJB do
    if StateKriteriaON then
```

```
    begin
      with QKriteria do
      begin
        Close;
        SQL.Clear;
        SQL.Add('From Jual');
        Open;
        if Kolom1.Checked then
        begin
          SQL.Add(SQLPrefix+'Kode_Kasa='"+ENilai2.Text+"");
          Open;
          lbKodeKassa.Caption:=ENilai2.Text;
        end else lbKodeKassa.Caption:='';
        if Kolom3.Checked then
        begin
          SQL.Add(SQLPrefix+'Tgl'+SBK3.Caption+':P3');
          ParamByName('P3').AsDate:=ENilai3.Date;
          Open;
          lbTanggal.Caption:=DateToStr(ENilai3.Date);
        end else lbTanggal.Caption:='';
        if Kolom4.Checked then
        SQL.Add('Order By Tgl');
        Open;
      end;
      DSJual.DataSet:=QKriteria;
    end else
      DSJual.DataSet:=DM.QDafJual;
  end;
  end;
  DSJual.DataSet.Refresh;
end;

procedure TfmTJual.SpeedButton2Click(Sender: TObject);
begin
  Tampilkan;
end;

procedure TfmTJual.DsJualDataChange(Sender: TObject; Field: TField);
begin
  with QItemPenjualan do
  begin
    Close;
    Params[0].AsInteger:=TabelBarang.Fields[0].AsInteger;
    Open;
  end;
end;

procedure TfmTJual.TabelBarangDblClick(Sender: TObject);
begin
  if Length(TabelBarang.Fields[0].AsString)>0 then
  begin
   with fmJual do
   begin
    StateEdit:=True;
    ENoFak.Caption:=TabelBarang.Fields[0].AsString;
    DTanggal.Date:=TabelBarang.Fields[1] AsDateTime;
    EKassa.Caption:=Tabelbarang.Fields[3].AsString;
    EOperator.Caption:=Tabelbarang.Fields[4].AsString;
    DM.TempJual.Open;
    Init;
    with QItemPenjualan do
    begin
      Open;First;
      while not EOF do
      begin
        DM.TempJual.Append;
        DM.TempJualKode.Value:=Fields[1].AsString;
        DM.TempJualNama.Value:=Fields[2].AsString;
        DM.TempJualBanyak.Value:=Fields[3].AsFloat;
        DM.TempJualHarga.Value:=Fields[4].AsCurrency;
```

```
        DM.TempJual.Post;
        Next;
      end;
    end;
   ShowModal;
   DSJual.DataSet.Close;
   DSJual.DataSet.Open;
   end;
  end;
end;


procedure TfmTJual.btnHapusTransaksiClick(Sender: TObject);
begin
  if MessageDlg('Yakin transaksi ini akan dihapus ?',mtConfirmation,[mbYes,mbNO],0)=mrYes then
  begin
    with QItemPenjualan do
    begin
      First;
      while not EOF do
      begin
        Update_Persediaan(Fields[1].AsString,Fields[3].AsFloat);
        Next;
      end;
    end;
    with QDelete do
    begin
      Close;
      Params[0].AsInteger:=TabelBarang.Fields[0].AsInteger;
      ExecSQL;
    end;
    CekStatusTombolHapus;
    DSJual.DataSet.Close;
    DSJual.DataSet.Open;
  end;
end;
end.

unit Util;

interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, Buttons, FileCtrl, ComCtrls, Db, DBTables;
type
  TfmUtilitas = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    rgJenis: TRadioGroup;
    gbTujuan: TGroupBox;
    GroupBox1: TGroupBox;
    Hapus1: TCheckBox;
    Hapus2: TCheckBox;
    Hapus3: TCheckBox;
    btnBatal: TSpeedButton;
    button2: TSpeedButton;
    Label1: TLabel;
    lbDrive: TLabel;
    drive: TDriveComboBox;
    Label3: TLabel;
    lbDirektori: TLabel;
    directory: TDirectoryListBox;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    Progress: TProgressBar;
    QDel: TQuery;
    QUtilitas: TQuery;
    Hapus4: TCheckBox;
    BM: TBatchMove;
```

```
    TTujuan: TTable;
    TSumber: TTable;
    Hapus5: TCheckBox;
    Hapus6: TCheckBox;
    GroupBox2: TGroupBox;
    PMulai: TDateTimePicker;
    PSampai: TDateTimePicker;
    Label2: TLabel;
    Label4: TLabel;
    procedure driveChange(Sender: TObject);
    procedure directoryChange(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure rgJenisClick(Sender: TObject);
    procedure button2Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure SpeedButton2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    procedure Delete;
    procedure Backup;
    procedure Restore;
  public
  end;
var
  fmUtilitas: TfmUtilitas;

implementation

uses UDM;

{$R *.DFM}

procedure TfmUtilitas.Delete;
begin
  with QDel do
  begin
    if Hapus2.Checked then
    begin
      Close;
      SQL.Clear;
      SQL.Add('Delete From Jual');
      SQL.Add('Where Tgl Between :P1 and :P2');
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQl;
    end;
    if Hapus3.Checked then
    begin
      Close;
      SQL.Clear;
      SQL.Add('Delete From Beli');
      SQL.Add('Where Tanggal Between :P1 and :P2');
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQl;
    end;
    if Hapus1.Checked then
    begin
      Close;
      SQL.Clear;
      SQL.Add('Delete From Barang ');
      ExecSQL;
    end;
    if Hapus4.Checked then
    begin
      Close;
      SQL.Clear;
      SQL.Add('Update Barang Set Sisa=0');
      ExecSQL;
```

```
    end;
    if Hapus5.Checked then
    begin
      Close;
      SQL.Clear;
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQL;
    end;
    if Hapus6.Checked then
    begin
      Close;
      SQL.Clear;
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQL;
    end;
  end;
end;

procedure TfmUtilitas.driveChange(Sender: TObject);
begin
  lbDrive.Caption:=Drive.Text;
  Directory.Drive:=Drive.Drive;
end;

procedure TfmUtilitas.directoryChange(Sender: TObject);
begin
  lbDirektori.Caption:=Directory.Directory;
end;

procedure TfmUtilitas.btnBatalClick(Sender: TObject);
begin
  ModalResult:=mrOk;
end;

procedure TfmUtilitas.rgJenisClick(Sender: TObject);
begin
  case rgJenis.ItemIndex of
    0 : gbTujuan.Caption:='Tujuan Backup :';
    1 : gbTujuan.Caption:='Sumber Data :';
  end;
end;



procedure TfmUtilitas.Backup;
begin
   Dir:=Directory.Directory;
   if Dir[length(Dir)]='\' then
     Dir:=Dir
   else
     Dir:=Dir+'\';

   Nama1:=Dir+'Barang.DB';
   Nama2:=Dir+'Jual.DB';
   Nama3:=Dir+'Beli.DB';
   Nama4:=Dir+'Vendor.DB';
   Nama5:=Dir+'RetBeli.DB';
   Nama6:=Dir+'RetJual.DB';

   with QUtilitas do
   begin
     if FileExists(Nama1) then
     begin
       Close;
       SQL.Clear;
       SQL.Add('Delete From "'+Nama1+'"');
       ExecSQL;
     end else
```

```
begin
  Close;
  SQL.Clear;
  SQL.Add('Create Table "'+Nama1+'"');
  SQL.Add('(Kode_Barang      Char(20),');
  SQL.Add(' Nama_Barang      Char(50),');
  SQL.Add(' Harga_Jual       Money,');
  SQL.Add(' Diskon           Numeric(10,2),');
  SQL.Add(' Sisa             Numeric(10,2),');
  SQL.Add(' Unit             Char(20),');
  SQL.Add(' Kassa            Char(3),');
  SQL.Add(' Stock_Minimum    Numeric(10,2),');
  SQL.Add(' Stock_Maksimum   Numeric(10,2))');
  ExecSQL;
end;
Close;
SQL.Clear;
SQL.Add('Insert Into "'+Nama1+'"');
SQL.Add('Select * From ":CASHREG:BARANG.DB"');
ExecSQL;

for i:=1 to dm.TBarang.RecordCount do
  progress.Position:=round(i/JumRec)*100;
Dir:=Directory.Directory;
//create file penjualan
if FileExists(Nama2) then
begin
  Close,
  SQL.Clear;
  SQL.Add('Delete From "'+Nama2+'"');
  ExecSQL;
end else
begin
  Close;
  SQL.Clear;
  SQL.Add('Create Table "'+Nama2+'"');
  SQL.Add('(Strook           Integer,');
  SQL.Add(' Kode_Barang      Char(20),');
  SQL.Add(' Tgl              Date,');
  SQL.Add(' Jumlah           Numeric(10,2),');
  SQL.Add(' Diskon           Numeric(10,2),');
  SQL.Add(' Kode_Kasa        Char(3),');
  SQL.Add(' ID               Char(15),');
  SQL.Add(' Jenis_Kode       Char(20),');
  SQL.Add(' Jenis_Nama       Char(10),');
  SQL.Add(' Harga_Jual       Money,');
  SQL.Add(' HPP              Money)');
  ExecSQL;
end;
Close;
SQL.Clear;
SQL.Add('Insert Into "'+Nama2+'"');
SQL.Add('Select * From ":CASHREG:JUAL.DB"');
SQL.Add('Where tgl between :P1 and :P2');
Params[0].AsDate:=PMulai.date;
Params[1].AsDate:=PSampai.date;
ExecSQL;

for i:=dm.TBarang.RecordCount to dm.Tbarang.RecordCount+dm.TJual.RecordCount do
  progress.Position.=round(i/JumRec)*100;

//create file pembelian
if FileExists(Nama3) then
begin
  Close;
  SQL.Clear;
  SQL.Add('Delete From "'+Nama3+'"');
  ExecSQL;
end else
begin
```

```
    Close;
    SQL.Clear;
    SQL.Add('Delete From "'+Nama4+'"');
    ExecSQL;
  end else
  begin
   Close;
   SQL.Clear;
   SQL.Add('Create Table "'+Nama4+'"');
   SQL.Add('(Kode          Char(11),');
   SQL.Add(' Nama          Char(50),');
   SQL.Add(' Alamat1       Char(50),');
   SQL.Add(' Alamat2       Char(50),');
   SQL.Add(' Kota          Char(30),');
   SQL.Add(' Provinsi      Char(30),');
   SQL.Add(' KodeWilayah   Char(10),');
   SQL.Add(' Kontak        Char(25),');
   SQL.Add(' Telp          Char(20),');
   SQL.Add(' Fax           Char(20))');


   ExecSQL;
  end;

  Close;
  SQL.Clear;
  SQL.Add('Insert Into "'+Nama4+'"');
  SQL.Add('Select * From ":CASHREG:VENDOR.DB"');
  ExecSQL;

  //create file reture beli
  if FileExists(Nama5) then
  begin
    Close;
    SQL.Clear;
    SQL.Add('Delete From "'+Nama5+'"');
    ExecSQL;
  end else
  begin
   Close;
   SQL.Clear;
   SQL.Add('Create Table "'+Nama5+'"');
   SQL.Add('(Nobukti       Char(20),');
   SQL.Add(' Tanggal       Date,');
   SQL.Add(' NoFakturBeli  Char(20),');
   SQL.Add(' KodeBarang    Char(20),');
   SQL.Add(' Jumlah        Integer,');
   SQL.Add(' Harga         Money)');
   ExecSQL;
  end;
  Close;
  SQL.Clear;
  SQL.Add('Insert Into "'+Nama5+'"');
  SQL.Add('Where tanggal between :P1 and :P2');
  Params[0].AsDate:=PMulai.date;
  Params[1].AsDate:=PSampai.date;
  ExecSQL;

  if FileExists(Nama6) then
  begin
    Close;
    SQL.Clear;
    SQL.Add('Delete From "'+Nama6+'"');
    ExecSQL;
  end else
  begin
   Close;
   SQL.Clear;
   SQL.Add('Create Table "'+Nama6+'"');
   SQL.Add('(Nobukti       Char(20),');
```

```
        SQL.Add(' Tanggal        Date,');
        SQL.Add(' NoFakturJual   Integer,');
        SQL.Add(' KodeBarang     Char(20),');
        SQL.Add(' Kassa          Char(3),');
        SQL.Add(' Jumlah         Integer,');
        SQL.Add(' Harga          Money)');
        ExecSQL;
      end;
      Close;
      SQL.Clear;
      SQL.Add('Insert Into "'+Nama5+'"');
      SQL.Add('Where tanggal between :P1 and :P2');
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQL;

      for i:=dm.Tbarang.RecordCount+dm.TJual.RecordCount to JumRec do
        progress.Position:=round(i/JumRec)*100;
      progress.Position:=0;
    end;
end;

procedure TfmUtilitas.Restore;
  Dir   : string;
  I: INTEGER;
begin
  progress.Position:=0;
  Dir:=Directory.Directory;
  if Dir[length(Dir)]='\' then
    Dir:=Dir
  else
    Dir:=Dir+'\';

  TSumber.DatabaseName:=Dir;

  with QUtilitas do
  begin
    if FileExists(Nama1) then
    begin
      TTujuan.TableName:='BARANG.DB';
      TSumber.TableName:=Nama1;
      BM.Execute;
    end;
    if FileExists(Nama2) then
    begin
      Close;
      SQL.Add('Select * From "'+Nama2+'"');
      SQL.Add('Where tgl between :P1 and :P2');
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQL;
    end;
    if FileExists(Nama3) then
    begin
      Close;
      SQL.Clear;
      SQL.Add('Where tanggal between :P1 and :P2');
      Params[0].AsDate:=PMulai.date;
      Params[1].AsDate:=PSampai.date;
      ExecSQL;
    end;
    if FileExists(Nama4) then
    begin
      TTujuan.TableName:='VENDOR.DB';
      TSumber.TableName:=Nama4;
      BM.Execute;
    end;
```

```
        SQL.Clear;
        SQL.Add('Insert Into ":CASHREG:RETBELI.DB"');
        SQL.Add('Select * From "'+Nama5+'"');
        SQL.Add('Where tanggal between :P1 and :P2');
        Params[0].AsDate:=PMulai.date;
        Params[1].AsDate:=PSampai.date;
        ExecSQL;
      end;
      if FileExists(Nama6) then
      begin
        Close;
        SQL.Clear;
        SQL.Add('Insert Into ":CASHREG:RETJUAL.DB"');
        SQL.Add('Select * From "'+Nama6+'"');
        SQL.Add('Where tanggal between :P1 and :P2');
        Params[0].AsDate:=PMulai.date;
        Params[1].AsDate:=PSampai.date;
        ExecSQL;
      end;
        progress.Position:=round(i/1000)*100;
    progress.Position:=0;
  end;
end;

procedure TfmUtilitas.FormActivate(Sender: TObject);
begin
  with DM do
  begin
    TBarang.Open;
    TJual.Open;
    TBeli.Open;
  end;
end;

procedure TfmUtilitas.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  with DM do
  begin
    TBarang.Close;
    TJual.Close;
    TBeli.Close;
  end;
end;

procedure TfmUtilitas.SpeedButton2Click(Sender: TObject);
begin
  Application.HelpFile:='CASHREG.HLP';
  Application.HelpContext(13);
end;

procedure TfmUtilitas.FormCreate(Sender: TObject);
begin
  PMulai.Date:=Now;
  PSampai.Date:=Now;
end;

end.

unit Vendor;

interface

uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 Spin, StdCtrls, ExtCtrls, Buttons, DBTables, Db;

type
```

```
    Panel4: TPanel;
    ScrollBox1: TScrollBox;
    GroupBox5: TGroupBox;
    Label15: TLabel;
    Label16: TLabel;
    Label17: TLabel;
    Label18: TLabel;
    Label19: TLabel;
    Label20: TLabel;
    Label21: TLabel;
    Label22: TLabel;
    Label23: TLabel;
    Label24: TLabel;
    EKode: TEdit;
    ENama: TEdit;
    EAlamat1: TEdit;
    EAlamat2: TEdit;
    EKota: TEdit;
    EProvinsi: TEdit;
    EKodeArea: TEdit;
    EKontak: TEdit;
    ETelepon: TEdit;
    EFax: TEdit;
    SpeedButton1: TSpeedButton;
    QVendor: TQuery;
    VendorSQL: TUpdateSQL;
    sbKode: TSpeedButton;
    Panel3: TPanel;
    btnEdit: TBitBtn;
    btnTambah: TBitBtn;
    btnHapus: TBitBtn;
    btnSimpan: TBitBtn;
    btnBatal: TBitBtn;
    procedure EBatasKreditKeyPress(Sender: TObject; var Key: Char);
    procedure SpeedButton1Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btnSimpanClick(Sender: TObject);
    procedure btnTambahClick(Sender: TObject);
    procedure btnEditClick(Sender: TObject);
    procedure btnHapusClick(Sender: TObject);
    procedure sbKodeClick(Sender: TObject);
    procedure EKodeExit(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure EKodeKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure FormShow(Sender: TObject);
  private
    JumSaldoawal,JumHutang: currency;
    procedure Initial_State;
    procedure Toggle(State:boolean);
    procedure Reset;
    procedure Clear;
    procedure CekStatus;
  public
    StateEdit,StateTambah, StateHapus : boolean;
  end;

var
  fmVendor: TfmVendor;

implementation

uses LuVendor, UDM;

{$R *.DFM}
```

```
      begin
        btnEdit.Enabled:=False;
        btnTambah.Enabled:=False;
        btnHapus.Enabled:=False;
      end else
      begin
        btnEdit.Enabled:=True;
        btnTambah.Enabled:=True;
        btnHapus.Enabled:=True;
      end;
    end;

    procedure TfmVendor.Initial_State;
    begin
      StateTambah:=False;
      StateEdit:=False;
      StateHapus:=False;
    end;

    procedure TfmVendor.Reset;
    begin
      Initial_State;
      Clear;
      Toggle(False);
      sbKode.Visible:=False;
      CekStatus;
    end;

    procedure TfmVendor.Toggle(State:boolean);
    begin
      if State then
      begin
        EKode.Enabled:=True;
        ENama.Enabled:=True;
        EAlamat1.Enabled:=True;
        EAlamat2.Enabled:=True;
        EKota.Enabled:=True;
        EProvinsi.Enabled:=True;
        EKodeArea.Enabled:=True;
        EKontak.Enabled:=True;
        ETelepon.Enabled:=True;
        EFax.Enabled:=True;
        ActiveControl:=EKode;
      end else
      begin
        EKode.Enabled:=False;
        ENama.Enabled:=False;
        EAlamat1.Enabled:=False;
        EAlamat2.Enabled:=False;
        EKota.Enabled:=False;
        EProvinsi.Enabled:=False;
        EKodeArea.Enabled:=False;
        EKontak.Enabled:=False;
        ETelepon.Enabled:=False;
        EFax.Enabled:=False;
      end;
    end;

    procedure TfmVendor.Clear;
    begin
      EKode.Clear;
      ENama.Clear;
      EAlamat1.Clear;
      EAlamat2.Clear;
      EKota.Clear;
      EProvinsi.Clear;
      EKodeArea.Clear;
```

```
  end;

  procedure TfmVendor.EBatasKreditKeyPress(Sender: TObject; var Key: Char);
  begin
    if not (Key in['0'..'9',',','.']) then
      Key:=#0;
  end;

  procedure TfmVendor.SpeedButton1Click(Sender: TObject);
  begin
    ModalResult:=mrOK;
  end;

  procedure TfmVendor.FormActivate(Sender: TObject);
  begin
    DM.TVendor.Open;
    sbKode.Visible:=False;
    btnSimpan.Enabled:=False;
    if not StateEdit then
    begin
      Clear;
      Toggle(False);
    end else
      Toggle(True);
    CekStatus;
  end;

procedure TfmVendor.btnSimpanClick(Sender: TObject);
begin
  if StateTambah then
  begin
    if MessageDlg('Data akan disimpan ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
    with VendorSQL.Query[ukInsert] do
    begin
      ParamByName('Kode').AsString:=EKode.Text;
      ParamByName('Nama').AsString:=ENama.Text;
      ParamByName('Alamat1').AsString:=EAlamat1.Text;
      ParamByName('Alamat2').AsString:=EAlamat2.Text;
      ParamByName('Kota').AsString:=EKota.Text;
      ParamByName('Provinsi').AsString:=EProvinsi.Text;
      ParamByName('KodeWilayah').AsString:=EKodeArea.Text;
      ParamByName('Kontak').AsString:=EKontak.Text;
      ParamByName('Telp').AsString:=ETelepon.Text;
      ParamByName('Fax').AsString:=EFax.Text;
    end;
    VendorSQL.ExecSQL(ukInsert);
  end else
  if StateEdit then
  begin
    with VendorSQL.Query[ukModify] do
    begin
      ParamByName('OLD_Kode').AsString:=EKode.Text;
      ParamByName('Nama').AsString:=ENama.Text;
      ParamByName('Alamat1').AsString:=EAlamat1.Text;
      ParamByName('Alamat2').AsString:=EAlamat2.Text;
      ParamByName('Kota').AsString:=EKota.Text;
      ParamByName('Provinsi').AsString:=EProvinsi.Text;
      ParamByName('KodeWilayah').AsString:=EKodeArea.Text;
      ParamByName('Kontak').AsString:=EKontak.Text;
      ParamByName('Telp').AsString:=ETelepon.Text;
      ParamByName('Fax').AsString:=EFax.Text;
    end;
    Reset;
  end;
  btnEdit.Enabled:=True;
  btnTambah.Enabled:=True;
  btnHapus.Enabled:=True;
```

```
procedure TfmVendor.btnTambahClick(Sender: TObject);
begin
  StateTambah:=True;
  StateEdit:=False;
  StateHapus:=False;
  Toggle(True);
  Clear;
end;

procedure TfmVendor.btnEditClick(Sender: TObject);
begin
  StateTambah:=False;
  StateEdit:=True;
  StateHapus:=False;
  Toggle(True);
  Clear;
  sbKode.Visible:=True;
end;

procedure TfmVendor.btnHapusClick(Sender: TObject);
begin
  StateTambah:=False;
  StateEdit:=False;
  StateHapus:=True;
  Toggle(True);
  Clear;
  sbKode.Visible:=True;
end;

procedure TfmVendor.sbKodeClick(Sender: TObject);
begin
  with fmLuVendor do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      EKode.Text:=KodeVendorCari;
      EKodeExit(Sender);
    end;
  end;
end;

procedure TfmVendor.EKodeExit(Sender: TObject);
begin
 if Length(EKode.Text)>0 then
 begin
  if StateTambah then
  begin
    DM.TVendor.IndexName:='';
    if DM.TVendor.FindKey([EKode.Text]) then
    begin
      MessageDlg('Kode sudah digunakan, ganti dengan yang lain !',mtInformation,[mbOK],0);
      ActiveControl:=EKode;
    end;
  end else
  if StateEdit or StateHapus then
  begin
    DM.TVendor.IndexName:='';
    if DM.TVendor.FindKey([EKode.Text]) then
    begin
      with QVendor do
      begin
        Close;
        Params[0].AsString:=EKode.Text;
        Open;
        ENama.Text:=FieldByName('Nama').AsString;
        EAlamat1.Text:=FieldByName('Alamat1').AsString;
```

```
        EKodeArea.Text:=FieldByName('KodeWilayah').AsString;
        EKontak.Text:=FieldByName('Kontak').AsString;
        ETelepon.Text:=FieldByName('Telp').AsString;
        EFax.Text:=FieldByName('Fax').AsString;
        Close;
      end;
      if StateHapus then
      begin
      MessageBeep(9);
      MessageDlg('Vendor belum ada dalam daftar !',mtInformation,[mbOK],0);
      ActiveControl:=EKode;
    end;
  end;
 end;
end;

procedure TfmVendor.btnBatalClick(Sender: TObject);
begin
  Reset;
  btnSimpan.Enabled:=False;
end;

procedure TfmVendor.EKodeKeyUp(Sender: TObject; var Key: Word;
 Shift: TShiftState);
begin
  btnSimpan.Enabled:=EKode.Text<>'';
end;

procedure TfmVendor.FormKeyPress(Sender: TObject; var Key: Char);
begin
  if Key=#13 then
  begin
    SelectNext(ActiveControl, True, True);
    Key:=#0;
  end;
end;

procedure TfmVendor.FormShow(Sender: TObject);
begin
  if StateEdit then
    EKodeExit(Sender);
end;
end.


unit EditPass;

interface

uses
 Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 ExtCtrls, StdCtrls, DB, DBTables, Buttons;

type
 TFormEditPass = class(TForm)
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Panel4: TPanel;
  ScrollBox1: TScrollBox;
  UserID: TEdit;
  Psw: TEdit;
  Label1: TLabel;
  LPassword: TLabel;
  QUpdate: TQuery;
  SpeedButton1: TSpeedButton;
```

```
    procedure FormKeyPress(Sender: TObject; var Key: Char);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure SpeedButton1Click(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
  private
    function CekPassWord:Boolean;
  end;

var
  FormEditPass: TFormEditPass;
  Jenis : string;
  PilihJenis, Stat : Boolean;


implementation

uses UDM;

{$R *.DFM}

function TFormEditPass.CekPassWord:Boolean;
var
  i,t1  : byte;
  s,s1  : string;
  t2    :real;
begin
  with DM.TPassWord do
  begin
    if FindKey([UserID.Text]) then
     begin
       t1:=ord(s[i]);
       t2:=t1-254;
       s1:=s1+chr(round(t2));
     end;
     if (UserID.Text<>'')and(Psw.Text=S1) then
     begin
       Result:=True;
     end else
       Result:=False;
   end;
  end;
end;

procedure TFormEditPass.btnOKClick(Sender: TObject);
begin
  if Stat then
  begin
    LPassword.Caption:='Password Lama :';
    LPassword.Repaint;
    OK:=CekPassword;
    PsW.Clear;
    if OK then
    begin
     stat:=False;
     LPassword.Caption:='Password Baru :';
     LPassword.Repaint;
    end
    else
     begin
       MessageDlg('Password salah !...',mtinformation,[mbOK],0);
       ActiveControl:=Psw;
       Stat:=True;
     end;
  end
  else
  begin
```

```
            t2:=t1+254;
            s1:=s1+chr(round(t2));
          end;
          with QUpdate do
          begin
            Close;
            ParamByName('Psw').AsString:=S1;
            ParamByName('ID').AsString:=UserID.Text;
            ExecSQL;
          end;
          Stat:=True;
        end;
        Close;
    end;
    DM.TPassword.Refresh;
  end;

  procedure TFormEditPass.FormActivate(Sender: TObject);
  begin
    DM.TPassword.Refresh;
    Psw.Text:='';
    LPassword.Caption:='Password Lama :';
    LPassWord.Repaint;
    ActiveControl:=Psw;
  end;

  procedure TFormEditPass.FormKeyPress(Sender: TObject; var Key: Char);
  begin
    if Key=#13 then
    begin
      Key:=#0;
      SelectNext(ActiveControl, True, True);
    end;
  end;

  procedure TFormEditPass.FormClose(Sender: TObject;
    var Action: TCloseAction);
  begin
    DM.TPassword.Close;
  end;

  procedure TFormEditPass.SpeedButton1Click(Sender: TObject);
  begin
    ModalResult:=mrOK;
  end;

  procedure TFormEditPass.btnBatalClick(Sender: TObject);
  begin
    ModalResult:=mrOK;
  end;
  end.

  unit Password;
  interface
  uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    ExtCtrls, StdCtrls, DB, DBTables, Buttons, checklst, Grids, DBGrids;

  type
    TFormPassword = class(TForm)
      Panel1: TPanel;
      Panel2: TPanel;
      Panel3: TPanel;
      Panel4: TPanel;
      ScrollBox1: TScrollBox;
      QInsert: TQuery;
      btnBrowse: TSpeedButton;
```

```
    GroupBox2: TGroupBox;
    CLMenu: TCheckListBox;
    TabelUser: TDBGrid;
    DSUser: TDataSource;
    GroupBox3: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    UserID: TEdit;
    Password: TEdit;
    SpeedButton1: TSpeedButton;
    sbClear: TSpeedButton;
    QEdit: TQuery;
    Button1: TButton;
    procedure btnOKClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure BtnBrowseClick(Sender: TObject);
    procedure UserIDExit(Sender: TObject);
    procedure btnBatalClick(Sender: TObject);
    procedure SpeedButton1Click(Sender: TObject);
    procedure sbClearClick(Sender: TObject);
    procedure TabelUserDblClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
  private
    Fasilitas: string;
    procedure CekFasilitas_Penggunaan;
  public
  end;

var
  FormPassword: TFormPassword;
  Jenis : char;
  PilihJenis : Boolean;
  StatusEdit: boolean=False;
uses UDM, U_LPass;
{$R *.DFM}
procedure TFormPassword.CekFasilitas_Penggunaan;
var
  i : byte;
  s : string;
begin
  s:='';
  with CLMenu do
  begin
    for i:=0 to 23 do
      if Checked[i] then s:=s+Chr(65+i);
  end;
  Fasilitas:=s;
end;

procedure TFormPassword.btnOKClick(Sender: TObject);
  begin
    with QEdit do
    begin
      Close;
      ParamByName('ID').AsString:=UserID.Text;
      ParamByName('Pass').AsString:=s1;
      ParamByName('Menu').AsString:=Fasilitas;
      ExecSQL;
    end;
  end else
  begin
    with QInsert do
    begin
      Close;
      ParamByName('ID').AsString:=UserID.Text;
```

```
  begin
    if DM.TBarang.FindKey([EKode.Text]) then
    begin
      MessageDlg('Kode barang sudah digunakan, ganti dengan yang lain !',
        mtinformation,[mbOK],0);
      ActiveControl:=EKode;
    end;
  end else
  if StateEdit or StateDel then
  begin
    with DM.TBarang do
    begin
      if FindKey([EKode.Text]) then
      begin
        EKode.Text:=FieldByName('Kode_Barang').AsString;
        EKeterangan.Text:=FieldByName('Nama_Barang').AsString;
        EHargaJual.Text:=CurrToStr(FieldByName('Harga_Jual').AsCurrency);
        EUnit.Text:=FieldByName('Unit').AsString;
        EDiskon.Text:=FloatToStr(FieldByName('Diskon').AsFloat);
        EJumlah.Text:=FloatToStr(FieldByName('Sisa').AsFloat);
        cbKassa.Text:=FieldByName('Kassa').AsString;
        EStockMinimum.Text:=FloatToStr(FieldByName('Stock_Minimum').AsFloat);
        EStockMaksimum.Text:=FloatToStr(FieldByName('Stock_Maksimum').AsFloat);
        OK:=True;
      end else
      begin
        MessageDlg('Data barang belum ada !',mtinformation,[mbOK],0);
        OK:=False;
      end;
    end;
    if StateDel and OK then
    begin
      if MessageDlg('Benar data ini akan dihapus ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
      begin
        SQLBarang.Query[ukDelete].ParamByName('OLD_Kode_Barang').AsString:=EKode.Text;
        SQLBarang.ExecSQL(ukDelete);
        //update transaksi penjualan dan pembelian barang
        UpdateTransaksi(EKode.Text);
        StateOff;
        Clear;
      end else
      begin
        StateOff;
        FormActivate(Sender);
        Clear;
      end;
    end;
  end;
 end;
end;

procedure TfmDetil.UpdateTransaksi(Kode:string);
begin
  with QUpdateTransaksi do
  begin
    Close;
    SQL.Clear;
    SQL.Add('Delete From Beli Where Kode_Barang=:Kode');
    Params[0].AsString:=Kode;
    ExecSQL;
    Close;
    SQL.Clear;
    SQL.Add('Delete From Jual Where Kode_Barang=:Kode');
    Params[0].AsString:=Kode;
    ExecSQL;
  end;
end;
```

```
  LoadKassa;
end;

procedure TfmDetil.btnEditClick(Sender: TObject);
begin
  StateAdd:=False;
  StateDel:=False;
  StateEdit:=True;
  sbKode.Visible:=True;
  Toggle(True);
  Clear;
  ActiveControl:=EKode;
  Status.Visible:=True;
  Pesan:='<< Edit Data >>';
  i:=1;
  Timer1.Enabled:=True;
end;

procedure TfmDetil.btnHapusClick(Sender: TObject);
begin
  StateAdd:=False;
  StateDel:=True;
  StateEdit:=False;
  sbKode.Visible:=True;
  Toggle(True);
  Clear;
  ActiveControl:=EKode;
  Status.Visible:=True;
  Pesan:='<< Hapus Data >>';
  i:=1;
  Timer1.Enabled:=True;
end;

procedure TfmDetil.btnBatalClick(Sender: TObject);
begin
  Clear;
  StateOff;
  FormActivate(Sender);
  btnTambah.Enabled:=True;
  btnHapus.Enabled:=True;
end;

procedure TfmDetil.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  DM.TBarang.Refresh;
  Timer1.Enabled:=False;
end;

procedure TfmDetil.btnTutupClick(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmDetil.sbKodeClick(Sender: TObject);
begin
  with fmLuBarang do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      Ekode.Text:=KodeCari;
      EKodeExit(Sender);
    end;
  end;
end;


procedure TfmDetil.EKodeKeyUp(Sender: TObject; var Key: Word;
```

```
  if StateAdd then
    btnSimpan.Enabled:=not(DM.TBarang.FindKey([EKode.Text]));

  if StateEdit then
    btnSimpan.Enabled:=EKode.Text<>'';

end;

procedure TfmDetil.Timer1Timer(Sender: TObject);
begin
    Status.Repaint;
    if (i mod 2)=1 then
      Status.Caption:=Pesan
    else
      Status.Caption:='';
   Status.Repaint;
   inc(i);
end;

procedure TfmDetil.btnBantuanClick(Sender: TObject);
begin
  Application.HelpFile:='CASHREG.HLP';
  Application.HelpContext(20);
end;

procedure TfmDetil.EHargaJualKeyPress(Sender: TObject; var Key: Char);
begin
 if not (Key in['0'..'9',',','.']) then
    Key:=#0;
end;

procedure TfmDetil.EDiskonKeyPress(Sender: TObject; var Key: Char);
begin
  if not (Key in['0'..'9',',','.']) then
    Key:=#0;
end;
procedure TfmDetil.EJumlahKeyPress(Sender: TObject; var Key: Char);
begin
  if not (Key in['0'..'9',',','.']) then
    Key:=#0;
end;
procedure TfmDetil.EStockMaksimumExit(Sender: TObject);
begin
  if btnSimpan.Enabled then
     btnSimpanClick(Sender);
end;
end.
unit EditBeli;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, ComCtrls, Buttons, DBTables, Db;

type
  TfmEditBeli = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    GroupBox1: TGroupBox;
    ENoFaktur: TEdit;
    EKodeBarang: TEdit;
    ENamaBarang: TEdit;
    ETanggal: TDateTimePicker;
    EJumlah: TEdit;
    EHarga: TEdit;
    EID: TEdit;
    Label1: TLabel;
```

```
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  SpeedButton1: TSpeedButton;
  btnHapus: TSpeedButton;
  btnSImpan: TSpeedButton;
  btnBatal: TSpeedButton;
  sbKode: TSpeedButton;
  Query1: TQuery;
  BeliSQL: TUpdateSQL;
  QUpdateStock: TQuery;
  procedure EJumlahKeyPress(Sender: TObject; var Key: Char);
  procedure sbKodeClick(Sender: TObject);
  procedure EKodeBarangExit(Sender: TObject);
  procedure SpeedButton1Click(Sender: TObject);
  procedure FormKeyPress(Sender: TObject; var Key: Char);
  procedure btnHapusClick(Sender: TObject);
  procedure btnSImpanClick(Sender: TObject);
 private
  procedure UpdateStock(Jenis:byte);
 public
  JumlahBeliAsal: Real;
 end;

var
 fmEditBeli: TfmEditBeli;

implementation

uses LuBarang, UDM;

{$R *.DFM}


procedure TfmEditBeli.EJumlahKeyPress(Sender: TObject; var Key: Char);
begin
  if not (Key in['0'..'9',',','.']) then
    Key:=#0;
end;

procedure TfmEditBeli.UpdateStock(Jenis:byte);
var
  JumUpdate,
  TotalUpdate : Real;
begin
  case Jenis of
  1 : begin
      JumUpdate:=StrToFloat(EJumlah.Text);
      TotalUpdate:=JumUpdate-JumlahBeliAsal;
      with QUpdateStock do
      begin
        Close;
        SQL.Clear;
        SQL.Add('Update Barang Set Sisa=Sisa+:P1');
        SQL.Add('Where Kode_Barang=:P2');
        ParamByName('P1').AsFloat:=TotalUpdate;
        ParamByName('P2').AsString:=EKodeBarang.Text;
        ExecSQL;
      end;
    end;
  2 : begin
      TotalUpdate:=JumlahBeliAsal;
      with QUpdateStock do
      begin
        Close;
        SQL.Clear;
        SQL.Add('Update Barang Set Sisa=Sisa-:P1');
```

```
          SQL.Add('Where Kode_Barang=:P2');
          ParamByName('P1').AsFloat:=TotalUpdate;
          ParamByName('P2').AsString:=EKodeBarang.Text;
          ExecSQL;
        end;
      end;
    end;
end;

procedure TfmEditBeli.sbKodeClick(Sender: TObject);
begin
  with fmLuBarang do
  begin
    ShowModal;
    if ModalResult=mrOK then
    begin
      EKodeBarang.Text:=KodeCari;
      ENamaBarang.Text:=NamaCari;
      EKodeBarangExit(Sender);
    end;
  end;
end;

procedure TfmEditBeli.EKodeBarangExit(Sender: TObject);
begin
  if (DM.TBarang.FindKey([EKodeBarang.Text]))and(EKodebarang.Text<>'') then
    ENamaBarang.Text:=DM.TBarang.FieldByName('Nama_Barang').AsString
  else
  begin
    MessageDlg('Data barang belum ada !',mtinformation,[mbOK],0);
    ActiveControl:=EKodeBarang;
  end;
end;

procedure TfmEditBeli.SpeedButton1Click(Sender: TObject);
begin
  ModalResult:=mrOK;
end;

procedure TfmEditBeli.FormKeyPress(Sender: TObject; var Key: Char);
begin
 if Key=#13 then
 begin
   SelectNext(ActiveControl, True, True);
   Key:=#0;
 end;
end;

procedure TfmEditBeli.btnHapusClick(Sender: TObject);
begin
 if MessageDlg('Benar data ini akan dihapus ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
 begin
   with BeliSQL.Query[ukDelete] do
     ParamByName('OLD_Strook').AsString:=ENoFaktur.Text;
   BeliSQL.ExecSQL(ukDelete);
   UpdateStock(2);
   ModalResult:=mrOK;
 end;
end;

procedure TfmEditBeli.btnSImpanClick(Sender: TObject);
begin
 if MessageDlg('Benar data ini akan disimpan ?',mtConfirmation,[mbYes,mbNo],0)=mrYes then
 begin
   with BeliSQL.Query[ukModify] do
   begin
     ParamByName('OLD_Strook').AsString:=ENoFaktur.Text;
     ParamByName('Kode_Barang').AsString:=EKodeBarang.Text;
     ParamByName('Tanggal').AsDate:=ETanggal.Date;
     ParamByName('Jumlah').AsFloat:=StrToFloat(EJumlah.Text);
```

```
        ParamByName('Harga').AsCurrency:=StrToCurr(EHarga.Text);
      end;
      BeliSQL.ExecSQL(ukModify);
      UpdateStock(1);
      ModalResult:=mrOK;
    end;
  end;
  end.


  unit Grafik;

  interface

  uses
   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
   ExtCtrls, Buttons, TeeProcs, TeEngine, Chart, Db, DBTables, Series,
   DBChart;

  type
   TfmGrafik = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    Panel3: TPanel;
    Panel4: TPanel;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    TGrafik: TTable;
    nbGrafik: TNotebook;
    Grafik1: TDBChart;
    Series1: TBarSeries;
    Series2: TBarSeries;
    Series3: TBarSeries;
    Grafik2: TDBChart;
    BarSeries1: TLineSeries;
    BarSeries2: TLineSeries;
    BarSeries3: TLineSeries;
    btnTutup: TSpeedButton;
    SpeedButton3: TSpeedButton;
    SpeedButton4: TSpeedButton;
    SpeedButton5: TSpeedButton;
    procedure SpeedButton1Click(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure btnTutupClick(Sender: TObject);
    procedure SpeedButton4Click(Sender: TObject);
    procedure SpeedButton5Click(Sender: TObject);
   private
    { Private declarations }
   public
    procedure ShowGraph(Jenis: byte);
   end;

  var
   fmGrafik: TfmGrafik;

  implementation

  uses Graph, UDM;

  {$R *.DFM}

  procedure TfmGrafik.ShowGraph(Jenis: byte);
  var
    Nama,Alamat : string;
  begin
    DM.ID_Usaha(Nama,Alamat);
    case Jenis of
    0 : begin
        with Grafik1 do
        begin
```