

LAMPIRAN

Form Motor

unit UMotor;

interface

uses

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

type

```
TfrmMotor = class(TForm)
  MainMenu1: TMainMenu;
  Data1: TMenuItem;
  ransaksi1: TMenuItem;
  Laporan1: TMenuItem;
  Keluar1: TMenuItem;
  Barang1: TMenuItem;
  Customer1: TMenuItem;
  Supplier1: TMenuItem;
  RekanKerja1: TMenuItem;
  Penjualan1: TMenuItem;
  Pembelian1: TMenuItem;
  frmPembayaranCicilan: TMenuItem;
  PenjualanKredit1: TMenuItem;
  procedure Barang1Click(Sender: TObject);
  procedure Keluar1Click(Sender: TObject);
  procedure Customer1Click(Sender: TObject);
  procedure Supplier1Click(Sender: TObject);
  procedure RekanKerja1Click(Sender: TObject);
  procedure Penjualan1Click(Sender: TObject);
  procedure Pembelian1Click(Sender: TObject);
  procedure frmPembayaranCicilanClick(Sender: TObject);
  procedure Laporan1Click(Sender: TObject);
  procedure PenjualanKredit1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

var

frmMotor: TfrmMotor;

implementation

uses UModule, UBarang, UCustomer, USupplier, URekan,
UPembelian, ULaporan, UPembayaranCicilan, UPenjualanKredit,
UPenjualanTunai;

{ \$R *.dfm }

procedure TfrmMotor.Barang1Click(Sender: TObject);

```

begin
    frmBarang.show;
end;

procedure TfrmMotor.Keluar1Click(Sender: TObject);
begin
    application.Terminate;
end;

procedure TfrmMotor.Customer1Click(Sender: TObject);
begin
    frmCustomer.show;
end;

procedure TfrmMotor.Supplier1Click(Sender: TObject);
begin
    frmSupplier.show;
end;

procedure TfrmMotor.RekanKerja1Click(Sender: TObject);
begin
    frmRekan.show;
end;

procedure TfrmMotor.Penjualan1Click(Sender: TObject);
begin
    frmJualTunai.show;
end;

procedure TfrmMotor.Pembelian1Click(Sender: TObject);
begin
    frmBeli.show;
end;

procedure TfrmMotor.Laporan1Click(Sender: TObject);
begin
    frmLaporan.show;
end;

procedure TfrmMotor.PenjualanKredit1Click(Sender: TObject);
begin
    frmJualKredit.show;
end;

end.

```

Form Barang

```
unit UBarang;
```

```
interface
```

```
uses
```

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

Dialogs, Grids, DBGrids, StdCtrls, ExtCtrls, DB, IBCustomDataSet, IBQuery, RpRave, RpDefine, RpCon, RpConDS;

type

```
TfrmBarang = class(TForm)
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  edNoPolisi: TEdit;
  edNama: TEdit;
  edAlamat: TEdit;
  edMerek: TEdit;
  edTipe: TEdit;
  edTahunBuat: TEdit;
  edTahunRakit: TEdit;
  edSilinder: TEdit;
  edWarna: TEdit;
  edNoRangka: TEdit;
  edNoMesin: TEdit;
  edNoBPKB: TEdit;
  edHargaBeli: TEdit;
  edHargaJual: TEdit;
  Label8: TLabel;
  Label9: TLabel;
  Label10: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  Label13: TLabel;
  Label14: TLabel;
  rgCari: TRadioGroup;
  edCari: TEdit;
  btSimpan: TButton;
  btCetak: TButton;
  btKeluar: TButton;
  dbgBarang: TDBGrid;
  rvDataBarang: TRvDataSetConnection;
  rvLaporan: TRvProject;
  qDataBarang: TIBQuery;
  qDataBarangNOPOLISI: TIBStringField;
  qDataBarangNAMAPEMILIK: TIBStringField;
  qDataBarangALAMAT: TIBStringField;
  qDataBarangMEREKMOTOR: TIBStringField;
  qDataBarangTIPEMOTOR: TIBStringField;
  qDataBarangTAHUNBUAT: TIBStringField;
  qDataBarangTAHUNRAKIT: TIBStringField;
  qDataBarangSILINDER: TIBStringField;
  qDataBarangWARNA: TIBStringField;
  qDataBarangNORANGKA: TIBStringField;
  qDataBarangNOMESIN: TIBStringField;
  qDataBarangNOBPKB: TIBStringField;
  qDataBarangHARGABELI: TIBBCDField;
```

```

qDataBarangHARGAJUAL: TIBBCDField;
qDataBarangSTOK: TIBStringField;
procedure btSimpanClick(Sender: TObject);
procedure btKeluarClick(Sender: TObject);
procedure edCariChange(Sender: TObject);
procedure Tampilkan;
procedure FormShow(Sender: TObject);
procedure dbgBarangDblClick(Sender: TObject);
procedure btCetakClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmBarang: TfrmBarang;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmBarang.btSimpanClick(Sender: TObject);
begin
  //proses menyimpan data
  with dmMotor.qData do
    begin
      close;
      sql.clear;
      sql.add('insert into barang (nopolisi, namapemilik, alamat, merekmotor, tipemotor, '+
        'tahunbuat, tahunrakit, silinder, warna, norangka, nomesis, nobpkb, hargabeli,
hargajual) '+
        'values(:np, :na, :al, :mk, :tp, :tb, :tr, :sl, :wr, :nr, :nm, :nb, :hb, :hj)');
      prepare;
      parambyname('np').AsString := ednopolisi.text;
      parambyname('na').AsString := ednama.text;
      parambyname('al').AsString := edalamat.text;
      parambyname('mk').AsString := edmerek.text;
      parambyname('tp').AsString := edtipe.text;
      parambyname('tb').AsString := edtahunbuat.text;
      parambyname('tr').AsString := edtahunrakit.text;
      parambyname('sl').AsString := edsilinder.text;
      parambyname('wr').AsString := edWarna.Text;
      parambyname('nr').AsString := ednorangka.text;
      parambyname('nm').AsString := ednomesis.text;
      parambyname('nb').AsString := ednobpkb.text;
      parambyname('hb').AsString := strttoFloat(edhargabeli.text);
      parambyname('hj').AsString := strttoFloat(edhargajual.text);
      execSql;
    end;

    dmMotor.trMotor.commit;

```

```

ednopolisi.clear;
ednama.clear;
edalamat.clear;
edmerek.clear;
edtiipe.clear;
edtahunbuat.clear;
edtahunrakit.clear;
edsilinder.clear;
edwarna.Clear;
ednorangka.clear;
ednomesin.clear;
ednobpkb.clear;
edhargabeli.clear;
edhargajual.clear;
ednopolisi.SetFocus;

Tampilkan;
messagedlg('Data telah disimpan ke dalam database',mtInformation,[mbOK],0);

end;

procedure TfrmBarang.Tampilkan();
begin
    with dmMotor.qProses do
        begin
            close;
            sql.clear;
            sql.add('select nopolisi, namapemilik, alamat, merekmotor, tipemotor, '+
                'tahunbuat, tahunrakit, silinder, warna, norangka, nomesin, nobpkb, hargabeli,
                hargajual, stok from Barang');
            open;
            end;
        end;

procedure TfrmBarang.btKeluarClick(Sender: TObject);
begin
    dmMotor.qProses.close;
    close;
end;

procedure TfrmBarang.edCariChange(Sender: TObject);
begin
    with dmMotor.qProses do
        begin
            close;
            sql.clear;
            sql.add('select * from barang');

            //cek item index pada radio group yg di klik
            if rgCari.itemindex = 0 then
                begin
                    sql.Add('where nopolisi like :np');
                    prepare;
                    parambyname('np').AsString := '%'+edcari.text+'%';
                end
            end;
        end;
    end;
end;

```

```

else
if rgCari.itemindex = 1 then
begin
    sql.Add('where namapemilik like :na');
    prepare;
    parambyname('na').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 2 then
begin
    sql.Add('where merekmotor like :mk');
    prepare;
    parambyname('mk').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 3 then
begin
    sql.Add('where tipemotor like :tp');
    prepare;
    parambyname('tp').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 4 then
begin
    sql.Add('where tahunbuat like :tb');
    prepare;
    parambyname('tb').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 5 then
begin
    sql.Add('where tahunrakit like :tr');
    prepare;
    parambyname('tr').AsString := '%'+edcari.text+'%';
end
else

if rgCari.itemindex = 6 then
begin
    sql.Add('where warna like :wr');
    prepare;
    parambyname('wr').AsString := '%'+edcari.text+'%';
end;
open;
end;
end;

procedure TfrmBarang.FormShow(Sender: TObject);
begin
    Tampilkan;
end;

procedure TfrmBarang.dbgBarangDbClick(Sender: TObject);
begin
    //konfirmasi

```

```

if messagedlg('Ingin menghapus data ini?',mtConfirmation,[mbYes,mbNo],0)=mrYes
then
begin
  with dmMotor.qData do
  begin
    close;
    sql.clear;
    sql.add('delete from barang');
    sql.add('where nopolisi =:np');
    prepare;
    parambyname('np').asstring := trim(dbgbarang.Fields[0].asstring);
    execsql;
  end;
  dmMotor.trMotor.commit;
  tampilkan;
end;
end;

procedure TfrmBarang.btCetakClick(Sender: TObject);
begin
  with qDataBarang do
  begin
    close;
    open;
  end;
  rvLaporan.Close;
  rvLaporan.Execute;
end;

end.

```

Form Customer

```
unit UCustomer;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmCustomer = class(TForm)
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  edKodeCust: TEdit;
  edNamaCust: TEdit;
  edAlamatCust: TEdit;
```

```

edTelpCust: TEdit;
edHPCust: TEdit;
btSimpan: TButton;
btCetak: TButton;
btKeluar: TButton;
rgCari: TRadioGroup;
edCari: TEdit;
dbgCustomer: TDBGrid;
qDataCustomer: TIBQuery;
rvDataCustomer: TRvDataSetConnection;
rvLaporan: TRvProject;
qDataCustomerKODECUSTOMER: TIBStringField;
qDataCustomerNAMACUSTOMER: TIBStringField;
qDataCustomerALAMAT: TIBStringField;
qDataCustomerTELEPON: TIBStringField;
qDataCustomerHP: TIBStringField;
procedure btSimpanClick(Sender: TObject);
procedure btKeluarClick(Sender: TObject);
procedure edCariChange(Sender: TObject);
procedure dbgCustomerDbClick(Sender: TObject);
procedure Tampilkan;
procedure FormShow(Sender: TObject);
procedure btCetakClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmCustomer: TfrmCustomer;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmCustomer.btSimpanClick(Sender: TObject);
begin
  //proses menyimpan data
  with dmMotor.qData do
    begin
      close;
      sql.clear;
      sql.add('insert into Customer (KodeCustomer, Namacustomer, Alamat, Telepon, HP)
'+
        'values(:kc, :nc, :al, :tp, :hp)');
      prepare;
      parambyname('kc').AsString := edkodecust.text;
      parambyname('nc').AsString := ednamacust.text;
      parambyname('al').AsString := edalamatcust.text;
      parambyname('tp').AsString := edtelpcust.text;
      parambyname('hp').AsString := edhpcust.text;
      execSql;
    end;
  end;

```

```

end;

dmMotor.trMotor.commit;

edkodecust.clear;
ednamacust.clear;
edalamatcust.Clear;
edtelpcust.clear;
edhpcust.clear;
edkodecust.SetFocus;
Tampilkan;
messagedlg('Data telah disimpan ke dalam database',mtInformation,[mbOK],0);
end;

procedure TfrmCustomer.Tampilkan();
begin
  with dmMotor.qProses do
    begin
      close;
      sql.clear;
      sql.add('select kodecustomer, namacustomer, alamat, telepon, hp from Customer');
      open;
    end;
end;

procedure TfrmCustomer.btKeluarClick(Sender: TObject);
begin
  dmMotor.qProses.close;
  close;
end;

procedure TfrmCustomer.edCariChange(Sender: TObject);
begin
  with dmMotor.qProses do
    begin
      close;
      sql.clear;
      sql.add('select * from customer');

      //cek item index pada radio group yg di klik
      if rgCari.itemindex = 0 then
        begin
          sql.Add('where kodecustomer like :kc');
          parambyname('kc').AsString := '%'+edcari.text+'%';
          prepare;
        end
      else
        if rgCari.itemindex = 1 then
          begin
            sql.Add('where namacustomer like :nc');
            prepare;
            parambyname('nc').AsString := '%'+edcari.text+'%';
          end
        else
          if rgCari.itemindex = 2 then

```

```

begin
    sql.Add('where alamat like :al');
    prepare;
    parambyname('al').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 3 then
begin
    sql.Add('where telepon like :tp');
    prepare;
    parambyname('tp').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 4 then
begin
    sql.Add('where HP like :hp');
    prepare;
    parambyname('hp').AsString := '%'+edcari.text+'%';
end;
open;
end;
end;

procedure TfrmCustomer.dbgCustomerDbClick(Sender: TObject);
begin
    //konfirmasi
    if messagedlg('Ingin menghapus data ini?',mtConfirmation,[mbYes,mbNo],0)=mrYes
then
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('delete from Customer');
        sql.add('where KodeCustomer =:kc');
        prepare;
        parambyname('kc').asString := trim(dbgCustomer.Fields[0].asString);
        execsql;
    end;
    dmMotor.trMotor.commit;
    tampilkan;
end;
end;

procedure TfrmCustomer.FormShow(Sender: TObject);
begin
    Tampilkan;
end;

procedure TfrmCustomer.btCetakClick(Sender: TObject);
begin
    with qDataCustomer do
    begin
        close;
        open;
    end;
end;

```

```

end;
rvLaporan.Close;
rvLaporan.Execute;
end;

```

```

end.

```

Form Supplier

```

unit USupplier;

```

```

interface

```

```

uses

```

```

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

```

```

type

```

```

    TfrmSupplier = class(TForm)
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        Label4: TLabel;
        Label5: TLabel;
        edKodeSupp: TEdit;
        edNamaSupp: TEdit;
        edAlamatSupp: TEdit;
        edTelpSupp: TEdit;
        edHPSupp: TEdit;
        btSimpan: TButton;
        btCetak: TButton;
        btKeluar: TButton;
        rgCari: TRadioGroup;
        edCari: TEdit;
        dbgSupplier: TDBGrid;
        rvLaporan: TRvProject;
        rvDataSupplier: TRvDataSetConnection;
        qDataSupplier: TIBQuery;
        qDataSupplierKODESUPPLIER: TIBStringField;
        qDataSupplierNAMASUPPLIER: TIBStringField;
        qDataSupplierALAMAT: TIBStringField;
        qDataSupplierTELEPON: TIBStringField;
        qDataSupplierHP: TIBStringField;
        procedure btSimpanClick(Sender: TObject);
        procedure btKeluarClick(Sender: TObject);
        procedure edCariChange(Sender: TObject);
        procedure dbgSupplierDblClick(Sender: TObject);
        procedure Tampilkan;
        procedure FormShow(Sender: TObject);
        procedure btCetakClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }

```

```

end;

var
    frmSupplier: TfrmSupplier;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmSupplier.btSimpanClick(Sender: TObject);
begin
    //proses menyimpan data
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('insert into Supplier (KodeSupplier, NamaSupplier, Alamat, Telepon, HP) '+
            'values(:ks, :ns, :al, :tp, :hp)');
        prepare;
        parambyname('ks').AsString := edkodesupp.text;
        parambyname('ns').AsString := ednamasupp.text;
        parambyname('al').AsString := edalamat supp.text;
        parambyname('tp').AsString := edtelpsupp.text;
        parambyname('hp').AsString := edhpsupp.text;
        execSql;
    end;

    dmMotor.trMotor.commit;

    edkodesupp.clear;
    ednamasupp.clear;
    edalamat supp.Clear;
    edtelpsupp.clear;
    edhpsupp.clear;
    edkodesupp.SetFocus;
    Tampilkan;
    messagedlg('Data telah disimpan ke dalam database',mtInformation,[mbOK],0);
end;

procedure TfrmSupplier.Tampilkan();
begin
    with dmMotor.qProses do
    begin
        close;
        sql.clear;
        sql.add('select kodesupplier, namaSupplier, alamat, telepon, hp from supplier');
        open;
    end;
end;

procedure TfrmSupplier.btKeluarClick(Sender: TObject);
begin
    dmMotor.qProses.close;

```

```

close;
end;

procedure TfrmSupplier.edCariChange(Sender: TObject);
begin
  with dmMotor.qProses do
  begin
    close;
    sql.clear;
    sql.add('select * from Supplier');

    //cek item index pada radio group yg di klik
    if rgCari.itemindex = 0 then
    begin
      sql.Add('where kodesupplier like :ks');
      parambyname('ks').AsString := '%'+edcari.text+'%';
      prepare;
    end
    else
    if rgCari.itemindex = 1 then
    begin
      sql.Add('where namasupplier like :ns');
      prepare;
      parambyname('ns').AsString := '%'+edcari.text+'%';
    end
    else
    if rgCari.itemindex = 2 then
    begin
      sql.Add('where alamat like :al');
      prepare;
      parambyname('al').AsString := '%'+edcari.text+'%';
    end
    else
    if rgCari.itemindex = 3 then
    begin
      sql.Add('where telepon like :tp');
      prepare;
      parambyname('tp').AsString := '%'+edcari.text+'%';
    end
    else
    if rgCari.itemindex = 4 then
    begin
      sql.Add('where HP like :hp');
      prepare;
      parambyname('hp').AsString := '%'+edcari.text+'%';
    end;
    open;
  end;
end;

procedure TfrmSupplier.dbgSupplierDbClick(Sender: TObject);
begin
  //konfirmasi
  if messagedlg('Ingin menghapus data ini?',mtConfirmation,[mbYes,mbNo],0)=mrYes
  then

```

```

begin
  with dmMotor.qData do
  begin
    close;
    sql.clear;
    sql.add('delete from Supplier');
    sql.add('where KodeSupplier =:ks');
    prepare;
    parambyname('ks').asstring := trim(dbgSupplier.Fields[0].asstring);
    execsql;
  end;
  dmMotor.trMotor.commit;
  tampilkan;
end;
end;

```

```

procedure TfrmSupplier.FormShow(Sender: TObject);
begin
  tampilkan;
end;

```

```

procedure TfrmSupplier.btCetakClick(Sender: TObject);
begin
  with qDataSupplier do
  begin
    close;
    open;
  end;
  rvLaporan.Close;
  rvLaporan.Execute;
end;

end.

```

Form Rekan Kerja

```

unit URekan;

```

```

interface

```

```

uses

```

```

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

```

```

type

```

```

  TfrmRekan = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    edKodeRekan: TEdit;
    edNamaRekan: TEdit;

```

```

edAlamatRekan: TEdit;
edTelpRekan: TEdit;
edHPRekan: TEdit;
btSimpan: TButton;
btCetak: TButton;
btKeluar: TButton;
rgCari: TRadioGroup;
edCari: TEdit;
dbgRekan: TDBGrid;
qDataRekan: TIBQuery;
rvDataRekan: TRvDataSetConnection;
rvLaporan: TRvProject;
qDataRekanKODEREKAN: TIBStringField;
qDataRekanNAMAREKAN: TIBStringField;
qDataRekanALAMAT: TIBStringField;
qDataRekanTELEPON: TIBStringField;
qDataRekanHP: TIBStringField;
procedure btKeluarClick(Sender: TObject);
procedure btSimpanClick(Sender: TObject);
procedure edCariChange(Sender: TObject);
procedure dbgRekanDbClick(Sender: TObject);
procedure Tampilkan;
procedure FormShow(Sender: TObject);
procedure btCetakClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmRekan: TfrmRekan;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmRekan.btKeluarClick(Sender: TObject);
begin
  dmMotor.qProses.close;
  close;
end;

procedure TfrmRekan.btSimpanClick(Sender: TObject);
begin
  //proses menyimpan data
  with dmMotor.qData do
    begin
      close;
      sql.clear;
      sql.add('insert into RekanKerja (Koderekan, NamaRekan, Alamat, Telepon, HP) '+
        'values(:kr, :na, :al, :tp, :hp)');
      prepare;
    end;
  end;
end;

```

```

    parambyname('kr').AsString := edkoderekan.text;
    parambyname('na').AsString := ednamarekan.text;
    parambyname('al').AsString := edalamatrekan.text;
    parambyname('tp').AsString := edtelprekan.text;
    parambyname('hp').AsString := edhprekan.text;
    execSql;
end;

dmMotor.trMotor.commit;

edkoderekan.clear;
ednamarekan.clear;
edalamatrekan.Clear;
edtelprekan.clear;
edhprekan.clear;
edkoderekan.SetFocus;
Tampilkan;
messagedlg('Data telah disimpan ke dalam database',mtInformation,[mbOK],0);
end;

procedure TfrmRekan.Tampilkan();
begin
    with dmMotor.qProses do
        begin
            close;
            sql.clear;
            sql.add('select koderekan, namarekan, alamat, telepon, hp from RekanKerja');
            open;
        end;
    end;
end;

procedure TfrmRekan.edCariChange(Sender: TObject);
begin
    with dmMotor.qProses do
        begin
            close;
            sql.clear;
            sql.add('select * from RekanKerja');

            //cek item index pada radio group yg di klik
            if rgCari.itemindex = 0 then
                begin
                    sql.Add('where koderekan like :kr');
                    parambyname('kr').AsString := '%'+edcari.text+'%';
                    prepare;
                end
            else
                if rgCari.itemindex = 1 then
                    begin
                        sql.Add('where namarekan like :na');
                        prepare;
                        parambyname('na').AsString := '%'+edcari.text+'%';
                    end
                else
                    if rgCari.itemindex = 2 then

```

```

begin
    sql.Add('where alamat like :al');
    prepare;
    parambyname('al').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 3 then
begin
    sql.Add('where telepon like :tp');
    prepare;
    parambyname('tp').AsString := '%'+edcari.text+'%';
end
else
if rgCari.itemindex = 4 then
begin
    sql.Add('where HP like :hp');
    prepare;
    parambyname('hp').AsString := '%'+edcari.text+'%';
end;
open;
end;
end;

procedure TfrmRekan.dbgRekanDbClick(Sender: TObject);
begin
    //konfirmasi
    if messagedlg('Ingin menghapus data ini?',mtConfirmation,[mbYes,mbNo],0)=mrYes
then
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('delete from RekanKerja');
        sql.add('where KodeRekan =:kr');
        prepare;
        parambyname('kr').asstring := trim(dbgRekan.Fields[0].asstring);
        execsql;
    end;
    dmMotor.trMotor.commit;
    tampilkan;
end;
end;

procedure TfrmRekan.FormShow(Sender: TObject);
begin
    Tampilkan;
end;

procedure TfrmRekan.btCetakClick(Sender: TObject);
begin
    with qDataRekan do
    begin
        close;
        open;
    end;
end;

```

```

end;
rvLaporan.Close;
rvLaporan.Execute;
end;

```

```

end.

```

Form Pembelian

```

unit UPembelian;

```

```

interface

```

```

uses

```

```

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

```

```

type

```

```

  TfrmBeli = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Alamat: TLabel;
    edtanggal: TEdit;
    edNoTransaksi: TEdit;
    edKodeSupplier: TEdit;
    cmbNamaSupplier: TComboBox;
    edAlamatSupplier: TEdit;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    edMerek: TEdit;
    cmbNoPolisi: TComboBox;
    edTipe: TEdit;
    edHargaBeli: TEdit;
    dbgBeli: TDBGrid;
    btSimpan: TButton;
    btKeluar: TButton;
    btTransaksiBaru: TButton;
    btCetak: TButton;
    rvLaporan: TRvProject;
    rvBeli: TRvDataSetConnection;
    qBeli: TIBQuery;
    qBeliNOTRANSAKSIBELI: TIBStringField;
    qBeliTANGGAL: TDateField;
    qBeliKODESUPPLIER: TIBStringField;
    qBeliNAMASUPPLIER: TIBStringField;
    qBeliNOPOLISI: TIBStringField;
    qBeliMEREKMOTOR: TIBStringField;
    qBeliTIPEMOTOR: TIBStringField;
    qBeliHARGABELI: TIBBCDField;
    procedure TampilkanItem;
    procedure btKeluarClick(Sender: TObject);
    procedure btTransaksiBaruClick(Sender: TObject);

```

```

    procedure btSimpanClick(Sender: TObject);
    procedure cmbNoPolisiChange(Sender: TObject);
    procedure cmbNamaSupplierChange(Sender: TObject);
    procedure dbgBeliDbClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btCetakClick(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmBeli: TfrmBeli;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmBeli.TampilkanItem;
begin
    //menampilkan tanggal pada edtanggal.text
    edtanggal.Text := datetostr(date);

    //memasukan data no polisi ke cmbnopolisi
    cmbNoPolisi.clear;
    with dmMotor.Qproses do
    begin
        close;
        sql.clear;
        sql.add('select distinct NoPolisi from Barang');
        sql.add('where stok is null');
        sql.add('order by NoPolisi');
        open;
        while not Eof do
        begin
            cmbNoPolisi.items.add(trim(fields[0].asString));
            next;
        end;
    end;

    //Menampilkan data di dbgbeli
    with dmMotor.qProses Do
    begin
        close;
        sql.clear;
        sql.add('select pembelian.nopolisi, barang.merekmotor, barang.tipemotor,' +
            'pembelian.hargabeli from pembelian');
        sql.add('inner join Barang on barang.nopolisi = pembelian.nopolisi');
        sql.add('where pembelian.notransaksibeli =:nb');
        prepare;
        parambyname('nb').asString := trim(ednotransaksi.text);
        open;
    end;
end;

```

```

end;
end;

procedure TfrmBeli.btKeluarClick(Sender: TObject);
begin
    //Pengecekan bila keluar saat bon masih kosong
    if dbgBeli.Fields[0].IsNull then
    begin
        //Bila bon belum dibuka, maka bisa langsung keluar
        if edNoTransaksi.text = " then
        begin
            close;
        end else
        //Konfirmasi penghapusan Transaksi kosong
        begin
            if      messagedlg('Transaksi      masih      kosong!      Tetap      ingin
keluar?',mtWarning,[mbYes,mbCancel],0) = mrYes then
            //Menghapus bon kosong
            begin
                with dmMotor.qData do
                begin
                    close;
                    sql.Clear;
                    sql.Add('delete from TransaksiBeli');
                    sql.Add('where noTransaksibeli =:nb');
                    prepare;
                    parambyname('nb').AsString := trim(edNoTransaksi.text);
                    execsql;
                    dmMotor.trMotor.Commit;
                end;
                edNoTransaksi.text := "";
                close;
            end;
        end;
    end else
    //Bila Transaksi tidak kosong, maka bisa langsung keluar
    begin
        close;
    end;

    edkodesupplier.clear;
    edalamatupplier.clear;
    edtipe.clear;
    edmerek.clear;
    edhargabeli.clear;
end;

procedure TfrmBeli.btTransaksiBaruClick(Sender: TObject);
begin
    //menampilkan no transaksi pada ednotransaksi
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select notransaksibeli from transaksibeli');

```

```

open;
//pengecekan transaksi baru
if fields [0].isnull then
begin
    ednotransaksi.text := '00001';
end else
//melihat no transaksi terakhir
begin
    with dmmotor.qData do
    begin
        close;
        sql.clear;
        sql.Add('select max(NoTransaksibeli) from transaksibeli');
        open;
        edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value+1));
    end;
end;
end;

//proses penyimpanan notransaksi pada tabel transaksibeli
with dmmotor.qProses do
begin
    close;
    sql.clear;
    sql.add('insert into transaksibeli(NoTransaksibeli,Tanggal) values(:nb,:tg)');
    prepare;
    parambyname('nb').asstring := trim(ednotransaksi.text);
    parambyname('tg').AsDate := strtodate(edtanggal.Text);
    execsql;
end;

dmmotor.trmotor.Commit;
bttransaksibaru.enabled := false;
btcetak.enabled := false;
btsimpan.enabled := true;
cmbNamaSupplier.Enabled := true;
cmbnopolisi.enabled := true;
cmbNamaSupplier.Text := trim('');
edkodesupplier.Clear;
edalamatupplier.clear;
tampilkanitem;
end;

procedure TfrmBeli.btSimpanClick(Sender: TObject);
begin
    //proses memasukan data ke tabel penjualan
    with dmmotor.qData do
    begin
        close;
        sql.clear;
        sql.add('insert into pembelian(notransaksibeli, nopolisi, hargabeli) '+
            'values(:nb, :np, :hb)');
        parambyname('nb').asstring := trim(ednotransaksi.Text);
        parambyname('np').asstring := trim(cmbnopolisi.text);
        parambyname('hb').asfloat := strttoFloat(edhargabeli.text);
    end;
end;

```

```

    execsql;
end;

//menambah jumlah stok
with dmMotor.qdata do
begin
    close;
    sql.Clear;
    sql.add('select stok from barang');
    sql.add('where nopolisi =:np');
    prepare;
    open;
    dmmotor.qData.close;
    dmmotor.qData.SQL.clear;
    dmmotor.qdata.sql.Add('update barang set stok =:st');
    dmmotor.qData.sql.add('where nopolisi =:np');
    prepare;
    parambyname('np').asstring := trim(cmbnopolisi.text);
    parambyname('st').asstring := '1';
    execsql;
end;

dmMotor.trMotor.commit;

cmbNoPolisi.text := trim("");
edTipe.text := "";
edmerek.Text := "";
edhargabeli.text := "";
bttransaksibaru.enabled := true;
btcetak.enabled := true;
dbgBeli.enabled := true;
tampilkanitem;
messagedlg('Data telah masuk ke dalam database.',mtInformation,[mbOK],0);
end;

procedure TfrmBeli.cmbNoPolisiChange(Sender: TObject);
begin
    //Menampilkan Merek, Tipe, dan Harga motor sesuai No Polisi yg dipilih
    with dmMotor.qData Do
    begin
        sql.Clear;
        sql.add('select MerekMotor, TipeMotor, HargaBeli from Barang');
        sql.add('where NoPolisi =:np');
        prepare;
        paramByName('np').AsString := trim(cmbNoPolisi.Text);
        open;
        edMerek.text := trim(fields[0].value);
        edTipe.text := trim(fields[1].value);
        edHargabeli.text := trim(fields[2].value);
    end;
end;

procedure TfrmBeli.cmbNamaSupplierChange(Sender: TObject);
begin
    //Menampilkan Kode Supplier dan alamat sesuai Nama Supplier di cmbNamaSupplier

```

```

with dmMotor.qData do
begin
  close;
  sql.clear;
  sql.add('Select KodeSupplier, Alamat from Supplier');
  sql.add('where NamaSupplier =:ns');
  prepare;
  ParamByName('ns').asstring := cmbNamaSupplier.text;
  open;
  edKodeSupplier.text := trim(fields[0].asstring);
  edAlamatSupplier.Text := trim(fields[1].asstring);
end;

//Jika cmbNamaCustomer tidak kosong, maka tampilkan Kode Customer dan Alamat
//dari tabel jual sesuai dengan no transaksi yg aktif
if cmbNamaSupplier.text <> " then
begin
  with dmMotor.qproses do
  begin
    close;
    sql.clear;
    sql.add('update transaksibeli set kodeSupplier =:ks');
    sql.add('where notransaksibeli =:nb');
    prepare;
    ParamByName('nb').asstring := Trim(edNoTransaksi.text);
    ParamByName('ks').asstring := Trim(edKodeSupplier.text);
    execsql;
  end;
  dmMotor.trMotor.commit;
end
else
begin
  messagedlg('Maaf Nama Supplier Harus Terisi.',mtWarning,[mbOK],0);
  cmbNamaSupplier.SetFocus;
end;
TampilkanItem;
edtipe.clear;
edmerek.clear;
edhargabeli.clear;
end;

procedure TfrmBeli.dbgBeliDblClick(Sender: TObject);
begin
  //konfirmasi penghapusan
  if messagedlg('Ingin menghapus data ini?', mtConfirmation,[mbYes,mbNo],0) =mrYes
  then
  begin
    //update stok karena batal membeli
    with dmMotor.qData do
    begin
      close;
      sql.clear;
      sql.add('select stok from barang');
      sql.add('where nopolisi =:np');
      prepare;
    end;
  end;
end;

```

```

parambyname('np').asstring := trim(dbgbeli.fields[0].asstring);
open;
dmMotor.qdata.close;
dmMotor.qdata.sql.clear;
dmMotor.qdata.sql.add('update barang set stok =:st');
dmMotor.qdata.SQL.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgbeli.fields[0].asstring);
parambyname('st').value := null;
execsql;
end;

//menghapus record yg ingin dihapus
with dmMotor.qdata do
begin
close;
sql.clear;
sql.add('delete from Pembelian');
sql.add('where notransaksibeli =:nb and nopolisi =:np');
prepare;
parambyname('nb').asstring := trim(ednotransaksi.Text);
parambyname('np').asstring := trim(dbgbeli.fields[0].asstring);
execsql;
end;
dmMotor.trmotor.commit;
tampilkanitem;

//bila dbgjual kosong, bttransaksibaru dan btcetak tidak aktif
if dbgbeli.fields[0].isnull then
begin
bttransaksibaru.Enabled := false;
btcetak.enabled := false;
dbgbeli.Enabled := false;
end;
cmbnopolisi.setfocus;
end;
end;

procedure TfrmBeli.FormActivate(Sender: TObject);
begin
//Memasukkan Nama Supplier ke cmbNamaSupplier
cmbNamaSupplier.clear;
With dmMotor.qProses Do
begin
close;
sql.clear;
sql.add('select distinct NamaSupplier from Supplier');
sql.add('order by NamaSupplier');
open;
while not Eof do
begin
cmbNamaSupplier.items.add(trim(fields[0].asstring));
next;
end;
end;
end;

```

```

//bila tidak kosong, maka data terkakhir ditampilkan
with dmMotor.qData Do
begin
    close;
    sql.clear;
    sql.add('Select NoTransaksiBeli from TransaksiBeli');
    open;
    if not fields[0].IsNull then
    begin
        with dmMotor.qData do
        begin
            close;
            sql.Clear;
            sql.Add('select max(NoTransaksibeli) from Transaksibeli');
            open;
            edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value));
        end;
    end;
end;

tampilkanitem;
cmbNamaSupplier.enabled := false;
cmbNoPolisi.enabled := false;
dbgbeli.enabled := false;
btsimpan.enabled := false;
btCetak.enabled := false;
messagedlg('Klik tombol "Transaksi Baru" untuk membuka Transaksi
Baru.',mtlInformation,[mBOK],0);
btTransaksibaru.enabled := true;
bttransaksibaru.setfocus;
end;

procedure TfrmBeli.btCetakClick(Sender: TObject);
begin
    with qBeli do
    begin
        close;
        prepare;
        parambyname('nb').AsString := trim(edNotransaksi.Text);
        open;
    end;
    rvlaporan.Close;
    rvlaporan.Execute;
end;

end.

```

Form Penjualan Tunai

unit UPenjualanTunai;

interface

uses

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

type

```
TfrmJualTunai = class(TForm)
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  Label8: TLabel;
  Label13: TLabel;
  Label14: TLabel;
  Alamat: TLabel;
  Label15: TLabel;
  edtanggal: TEdit;
  edNoTransaksi: TEdit;
  edKodeCustomer: TEdit;
  cmbNamaCustomer: TComboBox;
  edMerek: TEdit;
  edTipe: TEdit;
  edHargaJual: TEdit;
  rgKomisi: TRadioGroup;
  cmbNamaRekan: TComboBox;
  edKodeRekan: TEdit;
  edAlamatCustomer: TEdit;
  edAlamatRekan: TEdit;
  rvJualTunai: TRvDataSetConnection;
  rvLaporan: TRvProject;
  qJualTunai: TIBQuery;
  Label9: TLabel;
  edTotal: TEdit;
  rgAsuransi: TRadioGroup;
  btTransaksiBaru: TButton;
  btCetak: TButton;
  btKeluar: TButton;
  btSimpan: TButton;
  dbgJualTunai: TDBGrid;
  qJualTunaiNOTRANSAKSIJUAL: TIBStringField;
  qJualTunaiTANGGAL: TDateField;
  qJualTunaiKODECUSTOMER: TIBStringField;
  qJualTunaiNAMACUSTOMER: TIBStringField;
  qJualTunaiNOPOLISI: TIBStringField;
  qJualTunaiMEREKMOTOR: TIBStringField;
  qJualTunaiTIPEMOTOR: TIBStringField;
  qJualTunaiHARGAJUAL: TIBBCDField;
  qJualTunaiASURANSI: TIBBCDField;
  cmbNopolisi: TComboBox;
  procedure TampilkanItem;
  procedure btKeluarClick(Sender: TObject);
  procedure btCetakClick(Sender: TObject);
```

```

    procedure btTransaksiBaruClick(Sender: TObject);
    procedure btSimpanClick(Sender: TObject);
    procedure cmbNoPolisiChange(Sender: TObject);
    procedure cmbNamaCustomerChange(Sender: TObject);
    procedure dbgJualTunaiDbClick(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure cmbNamaRekanChange(Sender: TObject);
    procedure rgAsuransiClick(Sender: TObject);
    procedure rgKomisiClick(Sender: TObject);
    { Private declarations }
public
    { Public declarations }
end;

var
    frmJualTunai: TfrmJualTunai;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmJualTunai.TampilkanItem;
begin
    //menampilkan tanggal
    edTanggal.Text := DateToStr(Date);

    //memasukan namarekan ke cmbnamarekan
    cmbnamarekan.clear;
    with dmmotor.qProses do
    begin
        close;
        sql.clear;
        sql.add('select distinct namarekan from rekankerja');
        sql.add('order by namarekan');
        open;
        while not Eof do
        begin
            cmbnamarekan.items.add(trim(fields[0].asString));
            next;
        end;
    end;

    //Memasukkan Nomor Polisi ke cmbNoPolisi
    cmbNoPolisi.clear;
    with dmMotor.Qproses do
    begin
        close;
        sql.clear;
        sql.add('select distinct NoPolisi from Barang');
        sql.add('where stok = 1');
        sql.add('order by NoPolisi');
        open;
        while not Eof do

```

```

begin
    cmbNoPolisi.items.add(trim(fields[0].asString));
    next;
end;
end;

//Menampilkan semua data penjualan
with dmMotor.qProses Do
begin
    close;
    sql.clear;
    sql.add('select penjualanTunai.nopolisi, barang.merekmotor, barang.tipemotor, ' +
        'penjualanTunai.hargajual, penjualanTunai.asuransi, penjualanTunai.komisirekan,
penjualanTunai.total from PenjualanTunai');
    sql.add('inner join Barang on barang.nopolisi = penjualanTunai.nopolisi');
    sql.add('where penjualanTunai.notransaksijual =:nj');
    prepare;
    parambyname('nj').asString := trim(ednotransaksi.text);
    open;
end;
end;

procedure TfrmJualTunai.btKeluarClick(Sender: TObject);
begin
    //Pengecekan bila keluar saat transaksi masih kosong
    if dbgjualTunai.Fields[0].IsNull then
        begin
            //Bila bon belum dibuka, maka bisa langsung keluar
            if edNoTransaksi.text = "" then
                begin
                    close;
                end else
                //Konfirmasi penghapusan Transaksi kosong
                begin
                    if messagedlg('Transaksi masih kosong! Tetap ingin
keluar?',mtWarning,[mbYes,mbCancel],0) = mrYes then
                        //Menghapus bon kosong
                        begin
                            with dmMotor.qData do
                                begin
                                    close;
                                    sql.Clear;
                                    sql.Add('delete from TransaksijualTunai');
                                    sql.Add('where noTransaksijual =:nj');
                                    prepare;
                                    parambyname('nj').AsString := trim(edNoTransaksi.text);
                                    execsql;
                                    dmMotor.trMotor.Commit;
                                end;
                                edNoTransaksi.text := "";
                                close;
                            end;
                        end;
                    end else
                        //Bila Transaksi tidak kosong, maka bisa langsung keluar

```

```

begin
    close;
end;

edkodecustomer.clear;
edalamatcustomer.clear;
edmerek.clear;
edtiipe.clear;
edhargajual.clear;
edtotal.clear;
edkoderekan.clear;
edalamatrekan.clear;
end;

procedure TfrmJualTunai.btCetakClick(Sender: TObject);
begin
    with qjualtunai do
    begin
        close;
        prepare;
        parambyname('nj').asstring := trim(ednotransaksi.Text);
        open;
    end;
    rvlaporan.close;
    rvlaporan.Execute;
end;

procedure TfrmJualTunai.btTransaksiBaruClick(Sender: TObject);
begin
    //menampilkan no transaksi pada ednotransaksi
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select notransaksijual from transaksijualtunai');
        open;
        //pengecekan transaksi baru
        if fields [0].isnull then
        begin
            ednotransaksi.text := '00001';
        end else
        //melihat no transaksi terakhir
        begin
            with dmmotor.qData do
            begin
                close;
                sql.clear;
                sql.Add('select max(NoTransaksiJual) from transaksijualtunai');
                open;
                edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value+1));
            end;
        end;
    end;
    dmmotor.trmotor.Commit;
end;

```

```

bttransaksibaru.enabled := false;
btcetak.enabled := false;
btsimpan.enabled := true;
cmbNamacustomer.Enabled := true;
cmbnopolisi.enabled := true;
cmbNamarekan.enabled := false;
cmbNamacustomer.Text := trim("");
rgasuransi.enabled := true;
rgkomisi.enabled := true;
rgasuransi.itemindex := 1;
rgKomisi.itemindex := 1;
edkodecustomer.clear;
edalamatcustomer.Clear;
tampilkanitem;
end;

```

```

procedure TfrmJualTunai.btSimpanClick(Sender: TObject);
begin
    //proses memasukan data ke tabel penjualan tunai
    with dmmotor.qData do
    begin
        close;
        sql.clear;
        sql.add('insert into penjualantunai(notransaksijual, nopolisi, hargajual, asuransi,
        komisirekan, total) '+
        'values(:nj, :np, :hj, :as, :kr, :to)');
        prepare;
        parambyname('nj').asstring := trim(ednotransaksi.Text);
        parambyname('np').asstring := trim(cmbnopolisi.text);
        parambyname('hj').asfloat := strtofloat(edhargajual.text);

        //asuransi, komisi, total
        if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 0) then
        begin
            parambyname('as').asfloat := 300000;
            parambyname('kr').asfloat := 200000;
            parambyname('to').asfloat := strtofloat(edtotal.text);
        end
        else
        if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 0) then
        begin
            parambyname('as').asfloat := 0;
            parambyname('kr').asfloat := 200000;
            parambyname('to').asfloat := strtofloat(edtotal.text);
        end
        else
        if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 1) then
        begin
            parambyname('as').asfloat := 300000;
            parambyname('kr').asfloat := 0;
            parambyname('to').asfloat := strtofloat(edtotal.text);
        end
        else
        if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 1) then
        begin

```

```

    parambyname('as').asfloat := 0;
    parambyname('kr').asfloat := 0;
    parambyname('to').asfloat := strtofloat(edtotal.text);
end;
execsql;
end;

```

```

//proses memasukan data ke tabel transaksi penjualan tunai
with dmmotor.qData do
begin
    close;
    sql.clear;
    sql.add('insert into transaksijualtunai(notransaksijual, tanggal, kodecustomer,
komisirekan, koderekan, asuransi) '+
        'values(:nt, :tg, :kc, :km, :kk, :ar)');
    prepare;
    parambyname('nt').asstring := trim(ednotransaksi.text);
    parambyname('tg').AsDate := strtodate(edtanggal.Text);
    parambyname('kc').asstring := trim(edkodecustomer.text);
    if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 0) then
    begin
        parambyname('ar').asstring := 'Ya';
        parambyname('km').asstring := 'Ya';
        parambyname('kk').asstring := trim(edkoderekan.text);
    end
    else
    if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 0) then
    begin
        parambyname('ar').asstring := 'Tidak';
        parambyname('km').asstring := 'Ya';
        parambyname('kk').asstring := trim(edkoderekan.text);
    end
    else
    if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 1) then
    begin
        parambyname('ar').asstring := 'Ya';
        parambyname('km').asstring := 'Tidak';
        parambyname('kk').asstring := '-';
    end
    else
    if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 1) then
    begin
        parambyname('ar').asstring := 'Tidak';
        parambyname('km').asstring := 'Tidak';
        parambyname('kk').asstring := '-';
    end;
    execsql;
end;

```

```

//mengurangi jumlah stok
with dmmotor.qData do
begin
    close;
    sql.Clear;
    sql.add('select stok from barang');

```

```

sql.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(cmbnopolisi.text);
open;
dmmotor.qData.close;
dmmotor.qdata.SQL.clear;
dmmotor.qdata.sql.add('update barang set stok =:st');
dmmotor.qdata.sql.add('where nopolisi =:np');
prepare;
parambyname('np').AsString := trim(cmbnopolisi.Text);
parambyname('st').asstring := '0';
execsql;
end;

dmmotor.trmotor.commit;
cmbNoPolisi.text := trim("");
cmbnamarekan.text := trim("");
edkoderekan.clear;
edalamatrekan.clear;
tampilkanitem;
dbgjualTunai.enabled := true;
bttransaksibaru.enabled := true;
btcetak.enabled := true;
messagedlg('Data telah masuk ke dalam database.',mtInformation,[mbOK],0);
end;

```

```

procedure TfrmJualTunai.cmbNoPolisiChange(Sender: TObject);
begin
    //Menampilkan Merek, Tipe, dan Harga motor sesuai No Polisi yg dipilih
    with dmMotor.qData Do
    begin
        sql.Clear;
        sql.add('select MerekMotor, TipeMotor, HargaJual from Barang');
        sql.add('where NoPolisi =:np');
        prepare;
        paramByName('np').AsString := trim(cmbNoPolisi.Text);
        open;
        edMerek.text := trim(fields[0].value);
        edTipe.text := trim(fields[1].value);
        edHargaJual.text := trim(fields[2].value);
        edtotal.text := edhargajual.text;
    end;
    rgasuransi.ItemIndex := 1;
    rgkomisi.itemindex := 1;
    edtotal.text := edhargajual.text;
end;

```

```

procedure TfrmJualTunai.cmbNamaCustomerChange(Sender: TObject);
begin
    //Menampilkan Kode Customer dan alamat sesuai Nama Customer di
    cmbNamaCustomer
    with dmMotor.qData do
    begin
        close;
        sql.clear;
    end;

```

```

sql.add('Select KodeCustomer, Alamat from Customer');
sql.add('where NamaCustomer =:nc');
prepare;
ParamByName('nc').asstring := cmbNamaCustomer.text;
open;
edKodeCustomer.text := trim(fields[0].asstring);
edAlamatCustomer.Text := trim(fields[1].asstring);
end;

//Jika cmbNamaCustomer tidak kosong, maka tampilkan Kode Customer dan Alamat
//dari tabel jual sesuai dengan no transaksi yg aktif
if cmbNamaCustomer.text <> " then
begin
  with dmMotor.qproses do
  begin
    close;
    sql.clear;
    sql.add('update transaksijual set kodecustomer =:kc');
    sql.add('where notransaksijual =:nj');
    prepare;
    ParamByName('nj').asstring := Trim(edNoTransaksi.text);
    ParamByName('kc').asstring := Trim(edKodeCustomer.text);
    execsql;
  end;
  dmMotor.trMotor.commit;
end
else
begin
  messagedlg('Nama Customer Harus Terisi.',mtWarning,[mbOK],0);
  cmbNamaCustomer.SetFocus;
end;
cmbnopolisi.clear;
edmerek.clear;
ed tipe.clear;
edhargajual.clear;
edtotal.clear;
edkoderekan.clear;
edalamatrekan.clear;
rgasuransi.itemindex := 1;
rgKomisi.itemindex := 1;
TampilkanItem;
end;

procedure TfrmJualTunai.dbgJualTunaiDbClick(Sender: TObject);
begin
  //konfirmasi penghapusan
  if messagedlg('Ingin menghapus data ini?', mtConfirmation,[mbYes,mbNo],0) =mrYes
  then
  begin
    //update stok karena batal membeli
    with dmMotor.qData do
    begin
      close;
      sql.clear;
      sql.add('select stok from barang');

```

```

sql.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgjualtunai.fields[0].asstring);
open;
dmMotor.qdata.close;
dmMotor.qdata.sql.clear;
dmMotor.qdata.sql.add('update barang set stok =:st');
dmMotor.qdata.SQL.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgjualtunai.fields[0].asstring);
parambyname('st').asstring := '1';
execsql;
end;

//menghapus record yg ingin dihapus
with dmmotor.qData do
begin
close;
sql.clear;
sql.add('delete from transaksijualtunai');
sql.Add('where notransaksijual =:nj');
prepare;
parambyname('nj').asstring := trim(ednotransaksi.text);
execsql;
end;

with dmMotor.qdata do
begin
close;
sql.clear;
sql.add('delete from PenjualanTunai');
sql.add('where notransaksijual =:nj and nopolisi =:np');
prepare;
parambyname('nj').asstring := trim(ednotransaksi.Text);
parambyname('np').asstring := trim(dbgJualtunai.fields[0].asstring);
execsql;
end;

dmMotor.trmotor.commit;
tampilkanitem;

//bila dbgjual kosong, bttransaksibaru dan btcetak tidak aktif
if dbgjualtunai.fields[0].isnull then
begin
bttransaksibaru.Enabled := false;
btcetak.enabled := false;
dbgjualtunai.Enabled := false;
end;
cmbnopolisi.setfocus;
end;
end;

procedure TfrmJualTunai.FormActivate(Sender: TObject);
begin
//Memasukkan Nama Customer ke cmbNamaCustomer

```

```

cmbNamaCustomer.clear;
With dmMotor.qProses Do
begin
    close;
    sql.clear;
    sql.add('select distinct NamaCustomer from Customer');
    sql.add('order by NamaCustomer');
    open;
    while not Eof do
        begin
            cmbNamaCustomer.items.add(trim(fields[0].asString));
        end;
    next;
end;

//bila tidak kosong, maka data terkakhir ditampilkan
with dmMotor.qData Do
begin
    close;
    sql.clear;
    sql.add('Select NoTransaksiJual from TransaksiJualtunai');
    open;
    if not fields[0].IsNull then
        begin
            with dmMotor.qData do
            begin
                close;
                sql.Clear;
                sql.Add('select max(NoTransaksiJual) from TransaksiJualtunai');
                open;
                edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value));
            end;
        end;
    end;

    tampilkanitem;
    cmbNamaCustomer.enabled := false;
    cmbNoPolisi.enabled := false;
    cmbNamarekan.enabled := false;
    dbgJualtunai.enabled := false;
    rgasuransi.enabled := false;
    rgkomisi.enabled := false;
    btsimpan.enabled := false;
    btCetak.enabled := false;
    messagedlg('Klik tombol "Transaksi Baru" untuk membuka Transaksi
    Baru.',mtInformation,[mBOK],0);
    btTransaksibaru.enabled := true;
    bttransaksibaru.setfocus;
end;

procedure TfrmJualTunai.cmbNamaRekanChange(Sender: TObject);
begin
    //Menampilkan Kode rekan dan alamat sesuai Nama rekan di cmbNamarekan
    with dmMotor.qData do
    begin

```

```

close;
sql.clear;
sql.add('Select Koderekan, Alamat from RekanKerja');
sql.add('where NamaRekan =:nr');
prepare;
ParamByName('nr').asstring := cmbNamarekan.text;
open;
edKoderekan.text := trim(fields[0].asstring);
edAlamatrekan.Text := trim(fields[1].asstring);
end;
end;

procedure TfrmJualTunai.rgAsuransiClick(Sender: TObject);
begin
  if rgAsuransi.ItemIndex = 0 then
    begin
      edtotal.text := floattostr(strtfloat(edtotal.text) + 300000);
    end
  else
    if rgasuransi.itemindex = 1 then
      begin
        edtotal.text := floattostr(strtfloat(edtotal.text) - 300000);
      end;
    end;
end;

procedure TfrmJualTunai.rgKomisiClick(Sender: TObject);
begin
  if rgKomisi.ItemIndex = 0 then
    begin
      cmbnamarekan.enabled := true;
      edtotal.text := floattostr(strtfloat(edtotal.text) - 200000);
    end
  else
    if rgKomisi.itemindex = 1 then
      begin
        cmbnamarekan.Enabled := false;
        cmbnamarekan.text := trim("");
        edkoderekan.text := "";
        edalamatrekan.text := "";
        edtotal.text := floattostr(strtfloat(edtotal.text) + 200000);
      end;
    end;
end.

```

Form Penjualan Kredit

unit UPenjualanKredit;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, Grids, DBGrids, ExtCtrls, DB, IBCustomDataSet,

IBQuery, RpRave, RpDefine, RpCon, RpConDS, ComCtrls;

type

```
TfrmJualKredit = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    edtanggal: TEdit;
    edNoTransaksi: TEdit;
    Label3: TLabel;
    Label4: TLabel;
    edKodeCustomer: TEdit;
    cmbNamaCustomer: TComboBox;
    Label5: TLabel;
    edMerek: TEdit;
    cmbNoPolisi: TComboBox;
    Label6: TLabel;
    edTipe: TEdit;
    Label7: TLabel;
    dbgJualKredit: TDBGrid;
    Label8: TLabel;
    edHargaJual: TEdit;
    edTotal: TEdit;
    Label9: TLabel;
    btTransaksiBaru: TButton;
    btCetak: TButton;
    btKeluar: TButton;
    Label10: TLabel;
    edUangMuka: TEdit;
    edCicilan: TEdit;
    rgKomisi: TRadioGroup;
    Label12: TLabel;
    cmbNamaRekan: TComboBox;
    edKodeRekan: TEdit;
    Label13: TLabel;
    Label14: TLabel;
    btSimpan: TButton;
    Alamat: TLabel;
    edAlamatCustomer: TEdit;
    rgAsuransi: TRadioGroup;
    Label15: TLabel;
    edAlamatRekan: TEdit;
    rvJualKredit: TRvDataSetConnection;
    rvLaporan: TRvProject;
    qJualKredit: TIBQuery;
    rgLamakredit: TRadioGroup;
    qJualKreditNOTRANSAKSIJUAL: TIBStringField;
    qJualKreditTANGGAL: TDateField;
    qJualKreditKODECUSTOMER: TIBStringField;
    qJualKreditNAMACUSTOMER: TIBStringField;
    qJualKreditNOPOLISI: TIBStringField;
    qJualKreditMEREKMOTOR: TIBStringField;
    qJualKreditTIPEMOTOR: TIBStringField;
    qJualKreditHARGAJUAL: TIBBCDField;
    qJualKreditUANGMUKA: TIBBCDField;
    qJualKreditLAMAKREDIT: TIBStringField;
```

```

qJualKreditCICILAN: TIBBCDField;
qJualKreditASURANSI: TIBBCDField;
Label11: TLabel;
dtpJatuhTempo: TDateTimePicker;
procedure btKeluarClick(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure TampilkanItem;
procedure cmbNamaCustomerChange(Sender: TObject);
procedure cmbNoPolisiChange(Sender: TObject);
procedure dbgJualKreditDbClick(Sender: TObject);
procedure btCetakClick(Sender: TObject);
procedure btTransaksiBaruClick(Sender: TObject);
procedure btSimpanClick(Sender: TObject);
procedure cmbNamaRekanChange(Sender: TObject);
procedure rgAsuransiClick(Sender: TObject);
procedure rgKomisiClick(Sender: TObject);
procedure rgLamakreditClick(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

var
  frmJualKredit: TfrmJualKredit;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmJualKredit.TampilkanItem;
begin
  //menampilkan tanggal
  edTanggal.Text := DateToStr(Date);

  //memasukan namarekan ke cmbnamarekan
  cmbnamarekan.clear;
  with dmmotor.qProses do
  begin
    close;
    sql.clear;
    sql.add('select distinct namarekan from rekankerja');
    sql.add('order by namarekan');
    open;
    while not Eof do
    begin
      cmbnamarekan.items.add(trim(fields[0].asString));
    end;
  end;
end;

//Memasukkan Nomor Polisi ke cmbNoPolisi
cmbNoPolisi.clear;

```

```

with dmMotor.Qproses do
begin
  close;
  sql.clear;
  sql.add('select distinct NoPolisi from Barang');
  sql.add('where stok = 1');
  sql.add('order by NoPolisi');
  open;
  while not Eof do
  begin
    cmbNoPolisi.items.add(trim(fields[0].asString));
  next;
  end;
end;

//Menampilkan semua data penjualan kredit
with dmMotor.qProses Do
begin
  close;
  sql.clear;
  sql.add('select penjualankredit.nopolisi, barang.merekmotor, barang.tipemotor, ' +
    'penjualankredit.hargajual, penjualankredit.uangmuka, penjualankredit.asuransi,
penjualankredit.komisirekan, penjualankredit.total from Penjualankredit');
  sql.add('inner join Barang on barang.nopolisi = penjualankredit.nopolisi');
  sql.add('where penjualankredit.notransaksijual =:nj');
  prepare;
  parambyname('nj').asString := trim(ednotransaksi.text);
  open;
  end;
end;

procedure TfrmJualKredit.btKeluarClick(Sender: TObject);
begin
  //Pengecekan bila keluar saat transaksi masih kosong
  if dbgjualkredit.Fields[0].IsNull then
  begin
    //Bila bon belum dibuka, maka bisa langsung keluar
    if edNoTransaksi.text = '' then
    begin
      close;
    end else
    //Konfirmasi penghapusan Transaksi kosong
    begin
      if messagedlg('Transaksi masih kosong! Tetap ingin
keluar?',mtWarning,[mbYes,mbCancel],0) = mrYes then
      //Menghapus bon kosong
      begin
        with dmMotor.qData do
        begin
          close;
          sql.Clear;
          sql.Add('delete from Transaksijualkredit');
          sql.Add('where noTransaksijual =:nj');
          prepare;
          parambyname('nj').AsString := trim(edNoTransaksi.text);

```

```

        execsql;
        dmMotor.trMotor.Commit;
    end;
    edNoTransaksi.text := '';
    close;
end;
end;
end else
//Bila Transaksi tidak kosong, maka bisa langsung keluar
begin
    close;
end;

edkodecustomer.clear;
edalamatcustomer.clear;
edmerek.clear;
edtiipe.clear;
edhargajual.clear;
eduangmuka.clear;
edcicilan.clear;
edttotal.clear;
edkoderekan.clear;
edalamatrekan.clear;
end;

procedure TfrmJualKredit.btCetakClick(Sender: TObject);
begin
    with qJualkredit do
    begin
        close;
        prepare;
        parambyname('nj').AsString := trim(edNotransaksi.Text);
        open;
    end;
    rvlaporan.Close;
    rvlaporan.Execute;
end;

procedure TfrmJualKredit.btTransaksiBaruClick(Sender: TObject);
begin
    //menampilkan no transaksi pada ednotransaksi
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select notransaksijual from transaksijualkredit');
        open;
        //pengecekan transaksi baru
        if fields [0].IsNull then
        begin
            ednotransaksi.text := '00001';
        end else
        //melihat no transaksi terakhir
        begin
            with dmmotor.qData do

```

```

begin
  close;
  sql.clear;
  sql.Add('select max(NoTransaksiJual) from transaksijualkredit');
  open;
  edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value+1));
end;
end;
end;

dmmotor.trmotor.Commit;
bttransaksibaru.enabled := false;
btcetak.enabled := false;
btsimpan.enabled := true;
cmbNamacustomer.Enabled := true;
cmbnopolisi.enabled := true;
cmbNamarekan.enabled := false;
cmbNamacustomer.Text := trim("");
rglamakredit.Enabled := true;
rgasuransi.enabled := true;
rgkomisi.enabled := true;
rgasuransi.itemindex := 1;
rgKomisi.itemindex := 1;
rglamakredit.ItemIndex := -1;
edkodecustomer.clear;
edalamatcustomer.Clear;
tampilkanitem;
end;

procedure TfrmJualKredit.btSimpanClick(Sender: TObject);
begin
  //memasukan data ke tabel cicilan
  with dmmotor.qData do
  begin
    close;
    sql.clear;
    sql.add('insert into cicilan(kodecustomer, nopolisi, sisakredit, tgljatuhtempo, cicilan,
totaldenda) '+
      'values(:km, :ni, :sk, :jt, :cl, :td)');
    prepare;
    parambyname('km').asstring := trim(edkodecustomer.text);
    parambyname('ni').asstring := trim(cmbnopolisi.text);
    parambyname('sk').asstring := '24';
    parambyname('jt').asdate := dtpjatuhtempo.date;
    parambyname('cl').asfloat := strttoFloat(edcicilan.text);
    parambyname('td').asfloat := 0;
    execsql;
  end;

  //proses memasukan data ke tabel penjualan kredit
  with dmmotor.qData do
  begin
    close;
    sql.clear;

```

```

    sql.add('insert into penjualankredit(notransaksijual, nopolisi, hargajual, uangmuka,
asuransi, komisirekan, total) '+
        'values(:nj, :np, :hj, :um, :as, :kr, :to)');
    prepare;
    parambyname('nj').asstring := trim(ednotransaksi.Text);
    parambyname('np').asstring := trim(cmbnopolisi.text);
    parambyname('hj').asfloat := strtofloat(edhargajual.text);
    parambyname('um').asfloat := strtofloat(eduangmuka.text);

//asuransi, komisi, total
if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 0) then
begin
    parambyname('as').asfloat := 300000;
    parambyname('kr').asfloat := 200000;
    parambyname('to').asfloat := strtofloat(edtotal.text);
end
else
if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 0) then
begin
    parambyname('as').asfloat := 0;
    parambyname('kr').asfloat := 200000;
    parambyname('to').asfloat := strtofloat(edtotal.text);
end
else
if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 1) then
begin
    parambyname('as').asfloat := 300000;
    parambyname('kr').asfloat := 0;
    parambyname('to').asfloat := strtofloat(edtotal.text);
end
else
if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 1) then
begin
    parambyname('as').asfloat := 0;
    parambyname('kr').asfloat := 0;
    parambyname('to').asfloat := strtofloat(edtotal.text);
end;
execsql;
end;

//proses memasukan data ke tabel transaksi penjualan kredit
with dmmotor.qData do
begin
    close;
    sql.clear;
    sql.add('insert into transaksijualkredit(notransaksijual, tanggal, kodecustomer,
lamakredit, cicilan, komisirekan, koderekan, asuransi) '+
        'values(:nt, :tg, :kc, :lk, :ci, :km, :kk, :ar)');
    prepare;
    parambyname('nt').asstring := trim(ednotransaksi.text);
    parambyname('tg').asstring := trim(edtanggal.text);
    parambyname('kc').asstring := trim(edkodecustomer.text);

    if rglamakredit.itemindex = 0 then
begin

```

```

    parambyname('lk').asstring := '12 Bulan';
    parambyname('ci').asfloat := strtoint(edCicilan.Text);
end
else
if rglamakredit.itemindex = 1 then
begin
    parambyname('lk').asstring := '24 Bulan';
    parambyname('ci').asfloat := strtoint(edCicilan.Text);
end
else
if rglamakredit.itemindex = 2 then
begin
    parambyname('lk').asstring := '36 Bulan';
    parambyname('ci').asfloat := strtoint(edCicilan.Text);
end;

if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 0) then
begin
    parambyname('ar').asstring := 'Ya';
    parambyname('km').asstring := 'Ya';
    parambyname('kk').asstring := trim(edkoderekan.text);
end
else
if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 0) then
begin
    parambyname('ar').asstring := 'Tidak';
    parambyname('km').asstring := 'Ya';
    parambyname('kk').asstring := trim(edkoderekan.text);
end
else
if (rgasuransi.itemindex = 0) and (rgkomisi.itemindex = 1) then
begin
    parambyname('ar').asstring := 'Ya';
    parambyname('km').asstring := 'Tidak';
    parambyname('kk').asstring := '-';
end
else
if (rgasuransi.itemindex = 1) and (rgkomisi.itemindex = 1) then
begin
    parambyname('ar').asstring := 'Tidak';
    parambyname('km').asstring := 'Tidak';
    parambyname('kk').asstring := '-';
end;
execsql;
end;

//mengurangi jumlah stok
with dmmotor.qData do
begin
close;
sql.Clear;
sql.add('select stok from barang');
sql.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(cmbnopolisi.text);

```

```

open;
dmmotor.qData.close;
dmmotor.qdata.SQL.clear;
dmmotor.qdata.sql.add('update barang set stok =:st');
dmmotor.qdata.sql.add('where nopolisi =:np');
prepare;
parambyname('np').AsString := trim(cmbnopolisi.Text);
parambyname('st').AsString := '0';
execsql;
end;

dmmotor.trmotor.commit;
cmbNoPolisi.text := trim("");
cmbnamarekan.text := trim("");
edkoderekan.clear;
edalamatrekan.clear;
tampilkanitem;
dbgjualkredit.enabled := true;
bttransaksibaru.enabled := true;
btcetak.enabled := true;
messagedlg('Data telah masuk ke dalam database.',mtInformation,[mbOK],0);
end;

```

```

procedure TfrmJualKredit.cmbNoPolisiChange(Sender: TObject);
begin
    //Menampilkan Merek, Tipe, dan Harga motor sesuai No Polisi yg dipilih
    with dmMotor.qData Do
    begin
        sql.Clear;
        sql.add('select MerekMotor, TipeMotor, HargaJual from Barang');
        sql.add('where NoPolisi =:np');
        prepare;
        paramByName('np').AsString := trim(cmbNoPolisi.Text);
        open;
        edMerek.text := trim(fields[0].value);
        edTipe.text := trim(fields[1].value);
        edHargaJual.text := trim(fields[2].value);
        eduangmuka.text := floattostr(0.2 * (strtofloat(edhargajual.text)));
        edtotal.text := eduangmuka.text;
    end;
    rglamakredit.ItemIndex := -1;
    edcicilan.clear;
    rgasuransi.ItemIndex := 1;
    rgkomisi.itemindex := 1;
    edtotal.text := eduangmuka.text;
end;

```

```

procedure TfrmJualKredit.cmbNamaCustomerChange(Sender: TObject);
begin
    //Menampilkan Kode Customer dan alamat sesuai Nama Customer di
    cmbNamaCustomer
    with dmMotor.qData do
    begin
        close;
        sql.clear;
    end;

```

```

sql.add('Select KodeCustomer, Alamat from Customer');
sql.add('where NamaCustomer =:nc');
prepare;
ParamByName('nc').asstring := cmbNamaCustomer.text;
open;
edKodeCustomer.text := trim(fields[0].asstring);
edAlamatCustomer.Text := trim(fields[1].asstring);
end;

//Jika cmbNamaCustomer tidak kosong, maka tampilkan Kode Customer dan Alamat
//dari tabel jual sesuai dengan no transaksi yg aktif
if cmbNamaCustomer.text <> " then
begin
  with dmMotor.qproses do
  begin
    close;
    sql.clear;
    sql.add('update transaksijualkredit set kodecustomer =:kc');
    sql.add('where notransaksijual =:nj');
    prepare;
    ParamByName('nj').asstring := Trim(edNoTransaksi.text);
    ParamByName('kc').asstring := Trim(edKodeCustomer.text);
    execsql;
  end;
  dmMotor.trMotor.commit;
end
else
begin
  messagedlg('Maaf Nama Customer Harus Terisi.',mtWarning,[mbOK],0);
  cmbNamaCustomer.SetFocus;
end;
cmbnopolisi.clear;
edmerek.clear;
ed tipe.clear;
edhargajual.clear;
eduangmuka.clear;
edcicilan.clear;
edttotal.clear;
edkoderekan.clear;
edalamatrekan.clear;
rgasuransi.itemindex := 1;
rgKomisi.itemindex := 1;
TampilkanItem;
end;

procedure TfrmJualKredit.dbgJualKreditDbClick(Sender: TObject);
begin
  //konfirmasi penghapusan
  if messagedlg('Ingin menghapus data ini?', mtConfirmation,[mbYes,mbNo],0) =mrYes
then
  begin
    //update stok karena batal membeli
    with dmMotor.qData do
    begin
      close;

```

```

sql.clear;
sql.add('select stok from barang');
sql.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgjualkredit.fields[0].asstring);
open;
dmMotor.qdata.close;
dmMotor.qdata.sql.clear;
dmMotor.qdata.sql.add('update barang set stok =:st');
dmMotor.qdata.SQL.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgjualkredit.fields[0].asstring);
parambyname('st').asstring := '1';
execsql;
end;

```

```

//menghapus record yg ingin dihapus
//table transaksijualkredit
with dmmotor.qData do
begin
close;
sql.clear;
sql.add('delete from transaksijualkredit');
sql.Add('where notransaksijual =:nj');
prepare;
parambyname('nj').asstring := trim(ednotransaksi.text);
execsql;
end;

```

```

//table penjualankredit
with dmMotor.qdata do
begin
close;
sql.clear;
sql.add('delete from Penjualankredit');
sql.add('where notransaksijual =:nj and nopolisi =:np');
prepare;
parambyname('nj').asstring := trim(ednotransaksi.Text);
parambyname('np').asstring := trim(dbgJualkredit.fields[0].asstring);
execsql;
end;

```

```

with dmMotor.qdata do
begin
close;
sql.clear;
sql.add('delete from cicilan');
sql.add('where nopolisi =:np');
prepare;
parambyname('np').asstring := trim(dbgJualkredit.fields[0].asstring);
execsql;
end;

```

```

dmMotor.trmotor.commit;
tampilkanitem;

```

```

//bila dbgjual kosong, bttransaksibaru dan btcetak tidak aktif
if dbgjualkredit.fields[0].isnull then
begin
    bttransaksibaru.Enabled := false;
    btcetak.enabled := false;
    dbgjualkredit.Enabled := false;
end;
cmbnopolisi.setfocus;
end;
end;

procedure TfrmJualKredit.FormActivate(Sender: TObject);
begin
    //menampilkan tanggal di dtpjatuhtempo
    dtpjatuhtempo.Date := date;

    //Memasukkan Nama Customer ke cmbNamaCustomer
    cmbNamaCustomer.clear;
    With dmMotor.qProses Do
    begin
        close;
        sql.clear;
        sql.add('select distinct NamaCustomer from Customer');
        sql.add('order by NamaCustomer');
        open;
        while not Eof do
            begin
                cmbNamaCustomer.items.add(trim(fields[0].asString));
                next;
            end;
        end;
    end;

    //bila tidak kosong, maka data terkakhir ditampilkan
    with dmMotor.qData Do
    begin
        close;
        sql.clear;
        sql.add('Select NoTransaksiJual from TransaksiJualkredit');
        open;
        if not fields[0].IsNull then
            begin
                with dmMotor.qData do
                begin
                    close;
                    sql.Clear;
                    sql.Add('select max(NoTransaksiJual) from TransaksiJualkredit');
                    open;
                    edNoTransaksi.Text := FormatFloat('00000',StrToFloat(Fields[0].Value));
                end;
            end;
        end;
    end;

    tampilkanitem;
    cmbNamaCustomer.enabled := false;

```

```

cmbNoPolisi.enabled := false;
cmbNamarekan.enabled := false;
dbgJualkredit.enabled := false;
rglamakredit.Enabled := false;
rgasuransi.enabled := false;
rgkomisi.enabled := false;
btsimpan.enabled := false;
btCetak.enabled := false;
messagedlg('Klik tombol "Transaksi Baru" untuk membuka Transaksi
Baru.',mtInformation,[mBOK],0);
btTransaksibaru.enabled := true;
bttransaksibaru.setfocus;
end;

```

```

procedure TfrmJualKredit.cmbNamaRekanChange(Sender: TObject);
begin
    //Menampilkan Kode rekan dan alamat sesuai Nama rekan di cmbNamarekan
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('Select Koderekan, Alamat from RekanKerja');
        sql.add('where NamaRekan =:nr');
        prepare;
        ParamByName('nr').asstring := cmbNamarekan.text;
        open;
        edKoderekan.text := trim(fields[0].asstring);
        edAlamatrekan.Text := trim(fields[1].asstring);
    end;
end;

```

```

procedure TfrmJualKredit.rgAsuransiClick(Sender: TObject);
begin
    if rgAsuransi.ItemIndex = 0 then
    begin
        edtotal.text := floattostr(strtfloat(edtotal.text) + 300000);
    end
    else
    if rgasuransi.itemindex = 1 then
    begin
        edtotal.text := floattostr(strtfloat(edtotal.text) - 300000);
    end;
end;

```

```

procedure TfrmJualKredit.rgKomisiClick(Sender: TObject);
begin
    if rgKomisi.ItemIndex = 0 then
    begin
        cmbnamarekan.enabled := true;
        edtotal.text := floattostr(strtfloat(edtotal.text) - 200000);
    end
    else
    if rgKomisi.itemindex = 1 then
    begin
        cmbnamarekan.Enabled := false;
    end;
end;

```

```

    cmbnamarekan.text := trim("");
    edkoderekan.text := "";
    edalamatrekan.text := "";
    edttotal.text := floattostr(strtfloat(edtotal.text) + 200000);
end;
end;

procedure TfrmJualKredit.rgLamakreditClick(Sender: TObject);
begin
    if rglamakredit.itemindex = 0 then
    begin
        edcicilan.text := floattostr(Round((strtfloat(edhargajual.text) -
strtfloat(eduangmuka.Text))/12));
    end
    else
    if rglamakredit.itemindex = 1 then
    begin
        edcicilan.text := floattostr(Round((strtfloat(edhargajual.text) -
strtfloat(eduangmuka.Text))/24));
    end
    else
    if rglamakredit.itemindex = 2 then
    begin
        edcicilan.text := floattostr(Round((strtfloat(edhargajual.text) -
strtfloat(eduangmuka.Text))/36));
    end;
end;

end.

```

Form Laporan

unit ULaporan;

interface

uses

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

type

```

TfrmLaporan = class(TForm)
    rgLaporan: TRadioGroup;
    gbTanggal: TGroupBox;
    rgMotor: TRadioGroup;
    dtpTglDari: TDateTimePicker;
    dtpTglHingga: TDateTimePicker;
    Label1: TLabel;
    Label2: TLabel;
    gbCustomerTunai: TGroupBox;
    gbSupplier: TGroupBox;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;

```

```

Label6: TLabel;
cmbNamaCustomerTunai: TComboBox;
cmbKodeCustomerTunai: TComboBox;
cmbNamaSupplier: TComboBox;
cmbKodeSupplier: TComboBox;
btCetak: TButton;
btKeluar: TButton;
qLaporanBarang: TIBQuery;
qLaporanJualTunai: TIBQuery;
qLaporanBeli: TIBQuery;
rvLaporanBarang: TRvDataSetConnection;
rvLaporanJualTunai: TRvDataSetConnection;
rvLaporanBeli: TRvDataSetConnection;
rvLaporan: TRvProject;
gbCustomerKredit: TGroupBox;
Label7: TLabel;
Label8: TLabel;
cmbNamaCustomerKredit: TComboBox;
cmbKodeCustomerKredit: TComboBox;
qLaporanJualKredit: TIBQuery;
rvLaporanJualKredit: TRvDataSetConnection;
qLaporanBeliNOTRANSAKSIBELI: TIBStringField;
qLaporanBeliTANGGAL: TDateField;
qLaporanBeliKODESUPPLIER: TIBStringField;
qLaporanBeliNAMASUPPLIER: TIBStringField;
qLaporanBeliALAMAT: TIBStringField;
qLaporanBeliNOPOLISI: TIBStringField;
qLaporanBeliMEREKMOTOR: TIBStringField;
qLaporanBeliTIPEMOTOR: TIBStringField;
qLaporanBeliTAHUNRAKIT: TIBStringField;
qLaporanBeliHARGABELI: TIBBCDField;
rvLaporanCicilan: TRvDataSetConnection;
qLaporanCicilan: TIBQuery;
gbCustomerCicilan: TGroupBox;
Label9: TLabel;
Label10: TLabel;
cmbNamaCustomerCicilan: TComboBox;
cmbKodeCustomerCicilan: TComboBox;
qLaporanJualTunaiNOTRANSAKSIJUAL: TIBStringField;
qLaporanJualTunaiTANGGAL: TDateField;
qLaporanJualTunaiKODECUSTOMER: TIBStringField;
qLaporanJualTunaiNAMACUSTOMER: TIBStringField;
qLaporanJualTunaiALAMAT: TIBStringField;
qLaporanJualTunaiNOPOLISI: TIBStringField;
qLaporanJualTunaiMEREKMOTOR: TIBStringField;
qLaporanJualTunaiTIPEMOTOR: TIBStringField;
qLaporanJualTunaiTAHUNBUAT: TIBStringField;
qLaporanJualTunaiTOTAL: TIBBCDField;
qLaporanJualKreditNOTRANSAKSIJUAL: TIBStringField;
qLaporanJualKreditTANGGAL: TDateField;
qLaporanJualKreditKODECUSTOMER: TIBStringField;
qLaporanJualKreditNAMACUSTOMER: TIBStringField;
qLaporanJualKreditALAMAT: TIBStringField;
qLaporanJualKreditNOPOLISI: TIBStringField;
qLaporanJualKreditMEREKMOTOR: TIBStringField;

```

```

qLaporanJualKreditTIPEMOTOR: TIBStringField;
qLaporanJualKreditTAHUNBUAT: TIBStringField;
qLaporanJualKreditHARGAJUAL: TIBBCDField;
qLaporanJualKreditUANGMUKA: TIBBCDField;
qLaporanJualKreditLAMAKREDIT: TIBStringField;
qLaporanJualKreditTOTAL: TIBBCDField;
procedure TampilkanItem;
procedure btCetakClick(Sender: TObject);
procedure btKeluarClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure cmbNamaCustomerTunaiChange(Sender: TObject);
procedure cmbKodeCustomerTunaiChange(Sender: TObject);
procedure cmbNamaSupplierChange(Sender: TObject);
procedure cmbKodeSupplierChange(Sender: TObject);
procedure rgLaporanClick(Sender: TObject);
procedure cmbNamaCustomerKreditChange(Sender: TObject);
procedure cmbKodeCustomerKreditChange(Sender: TObject);
procedure cmbKodeCustomerCicilanChange(Sender: TObject);
procedure cmbNamaCustomerCicilanChange(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmLaporan: TfrmLaporan;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmLaporan.TampilkanItem;
begin
  cmbNamaCustomerTunai.Text := '';
  cmbKodeCustomerTunai.Text := '';
  cmbNamaCustomerKredit.Text := '';
  cmbKodeCustomerKredit.Text := '';
  cmbNamaSupplier.Text := '';
  cmbKodeSupplier.Text := '';
  dtpTglDari.date := date;
  dtpTglHingga.date := date;
  rgMotor.visible := false;
  gbCustomerTunai.visible := false;
  gbCustomerKredit.Visible := false;
  gbSupplier.visible := false;
  if rgLaporan.ItemIndex = 0 then
    begin
      rgMotor.Visible := true;
    end else
    if rgLaporan.ItemIndex = 1 then
      begin
        gbCustomerTunai.Visible := true;
      end
    end
  end;

```

```

end else
if rgLaporan.ItemIndex = 2 then
begin
    gbCustomerKredit.Visible := true;
end else
if rgLaporan.ItemIndex = 3 then
begin
    gbSupplier.Visible := true;
end;
end;

procedure TfrmLaporan.rgLaporanClick(Sender: TObject);
begin
    TampilkanItem;
end;

procedure TfrmLaporan.btCetakClick(Sender: TObject);
begin
    //pemilihan jenis laporan
    //laporan barang
    if rgLaporan.ItemIndex = 0 then
    begin
        with qLaporanBarang do
        begin
            close;
            prepare;
            parambyname('st').asString := trim(inttostr(rgMotor.itemindex));
            open;
        end;
        rvLaporan.close;
        rvLaporan.projectfile := '..\laporan\LaporanBarang.rav'
    end else
    //laporan penjualan tunai
    if rgLaporan.ItemIndex = 1 then
    begin
        with qLaporanjualTunai do
        begin
            close;
            prepare;
            parambyname('tgldari').AsDate := dtptgldari.Date;
            parambyname('tglhingga').AsDate := dtptglhingga.Date;
            parambyname('nc').AsString := trim('%'+cmbnamacustomertunai.Text+'%');
            open;
        end;
        rvlaporan.Close;
        rvlaporan.ProjectFile := '..\laporan\Laporanpenjualantunai.rav';
    end else
    //laporan penjualan kredit
    if rgLaporan.ItemIndex = 2 then
    begin
        with qLaporanjualKredit do
        begin
            close;
            prepare;
            parambyname('tgldari').AsDate := dtptgldari.Date;

```

```

        parambyname('tglhingga').AsDate := dtptglhingga.Date;
        parambyname('nc').AsString := trim('%'+cmbnamacustomerkredit.Text+'%');
        open;
    end;
    rvlaporan.Close;
    rvlaporan.ProjectFile := '..\laporan\Laporanpenjualankredit.rav';
end else
//laporan pembelian
if rglaporan.ItemIndex = 3 then
begin
    with qLaporanbeli do
    begin
        close;
        prepare;
        parambyname('tgldari').AsDate := dtptgldari.Date;
        parambyname('tglhingga').AsDate := dtptglhingga.Date;
        parambyname('ns').AsString := trim('%'+cmbnamasupplier.Text+'%');
        open;
    end;
    rvlaporan.Close;
    rvlaporan.ProjectFile := '..\laporan\Laporanbeli.rav';
end;

procedure TfrmLaporan.btKeluarClick(Sender: TObject);
begin
    close;
end;

procedure TfrmLaporan.FormShow(Sender: TObject);
begin
    //memasukan kode dan nama customer ke cmbnamacustomertunai dan
    cmbkodecustomertunai
    with dmMotor.qProses do
    begin
        close;
        sql.clear;
        sql.add('Select distinct customer.namacustomer, transaksijualtunai.kodecustomer from
transaksijualtunai');
        sql.add('inner join customer on customer.kodecustomer =
transaksijualtunai.kodecustomer');
        sql.add('order by Kodecustomer');
        open;
        while not Eof do
        begin
            cmbNamaCustomerTunai.Items.add(trim(fields[0].asstring));
            cmbKodeCustomerTunai.Items.add(trim(fields[1].asstring));
        next;
        end;
    end;

    //memasukan kode dan nama customer ke cmbnamacustomerkredit dan
    cmbkodecustomerkredit
    with dmMotor.qProses do
    begin
        close;

```

```

    sql.clear;
    sql.add('Select distinct customer.namacustomer, transaksijualkredit.kodecustomer
from transaksijualkredit');
    sql.add('inner join customer on customer.kodecustomer =
transaksijualkredit.kodecustomer');
    sql.add('order by Kodecustomer');
    open;
    while not Eof do
    begin
        cmbNamaCustomerkredit.Items.add(trim(fields[0].asstring));
        cmbKodeCustomerkredit.Items.add(trim(fields[1].asstring));
    next;
    end;
end;

```

```

//memasukan kode dan nama supplier ke cmbnamasupplier dan cmbkodesupplier
with dmMotor.qProses do
begin
    close;
    sql.clear;
    sql.add('select distinct kodesupplier, namasupplier from supplier');
    sql.add('order by Namasupplier');
    open;
    while not Eof do
    begin
        cmbKodesupplier.Items.add(trim(fields[0].asstring));
        cmbNamasupplier.Items.add(trim(fields[1].asstring));
    next;
    end;
end;

```

```

procedure TfrmLaporan.cmbNamaCustomerTunaiChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select distinct kodecustomer from customer');
        sql.add('where namacustomer =:nc');
        prepare;
        parambyname('nc').asstring := trim(cmbnamacustomertunai.text);
        open;
        cmbkodecustomertunai.Text := trim(fields[0].asstring);
    end;
end;

```

```

procedure TfrmLaporan.cmbKodeCustomerTunaiChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select distinct namacustomer from customer');
        sql.add('where kodecustomer =:kc');
        prepare;
    end;
end;

```

```

    parambyname('kc').asstring := trim(cmbkodecustomertunai.text);
    open;
    cmbnamacustomertunai.Text := trim(fields[0].asstring);
end;
end;

procedure TfrmLaporan.cmbNamaSupplierChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select distinct kodesupplier from supplier');
        sql.add('where namasupplier =:ns');
        prepare;
        parambyname('ns').asstring := trim(cmbnamasupplier.text);
        open;
        cmbkodesupplier.Text := trim(fields[0].asstring);
    end;
end;

procedure TfrmLaporan.cmbKodeSupplierChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select distinct namasupplier from supplier');
        sql.add('where kodesupplier =:ks');
        prepare;
        parambyname('ks').asstring := trim(cmbkodesupplier.text);
        open;
        cmbnamasupplier.Text := trim(fields[0].asstring);
    end;
end;

procedure TfrmLaporan.cmbNamaCustomerKreditChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin
        close;
        sql.clear;
        sql.add('select distinct kodecustomer from customer');
        sql.add('where namacustomer =:nc');
        prepare;
        parambyname('nc').asstring := trim(cmbnamacustomerkredit.text);
        open;
        cmbkodecustomerkredit.Text := trim(fields[0].asstring);
    end;
end;

procedure TfrmLaporan.cmbKodeCustomerKreditChange(Sender: TObject);
begin
    with dmMotor.qData do
    begin

```

```
close;
sql.clear;
sql.add('select distinct namacustomer from customer');
sql.add('where kodecustomer =:kc');
prepare;
parambyname('kc').asstring := trim(cmbkodecustomercredit.text);
open;
cmbnamacustomercredit.Text := trim(fields[0].asstring);
end;
end;

end.
```