

LAMPIRAN I

Listing Program

Pembuatan Database:

1. Pembuatan table Barang

```
CREATE TABLE "BARANG"  
(  
  "KODEBRG" CHAR(15) NOT NULL,  
  "NAMABRG" CHAR(25),  
  "MERKBRG" CHAR(30),  
  "HRGJUAL" NUMERIC(10, 2),  
  "HRGBELI" NUMERIC(10, 2),  
  "STOK" CHAR(3),  
  CONSTRAINT "PKKODEBRG" PRIMARY KEY ("KODEBRG")  
);
```

2. Pembuatan table Beli

```
CREATE TABLE "BELI"  
(  
  "BONBELI" CHAR(8) NOT NULL,  
  "KODESUPP" CHAR(10),  
  "TGL" DATE,  
  CONSTRAINT "PKBONBELI" PRIMARY KEY ("BONBELI")  
);  
ALTER TABLE "BELI" ADD CONSTRAINT "FKKODESUPP" FOREIGN KEY  
("KODESUPP") REFERENCES SUPPLIER ("KODESUPP");
```

3. Pembuatan table beliitem

```
CREATE TABLE "BELIITEM"  
(  
  "BONBELI" CHAR(8) NOT NULL,  
  "KODEBRG" CHAR(15) NOT NULL,  
  "QTY" NUMERIC(10, 2),  
  "HRGBELI" NUMERIC(10, 2),  
  CONSTRAINT "PKBeliBRG" PRIMARY KEY ("BONBELI", "KODEBRG")  
);
```

4. Pembuatan Tabel Jual

```
CREATE TABLE "JUAL"  
(  
  "BONJUAL" CHAR(8) NOT NULL,  
  "KODECUST" CHAR(10),  
  "TGL" DATE,  
  CONSTRAINT "PKBONJUAL" PRIMARY KEY ("BONJUAL")  
);  
ALTER TABLE "JUAL" ADD CONSTRAINT "FKKODECUST" FOREIGN KEY  
("KODECUST") REFERENCES CUSTOMER ("KODECUST");
```

5. Pembuatan Tabel JualItem

```
CREATE TABLE "JUALITEM"  
(  
  "BONJUAL" CHAR(8) NOT NULL,  
  "KODEBRG" CHAR(15) NOT NULL,  
  "QTY" NUMERIC(10, 2),  
  "HRGJUAL" NUMERIC(10, 2),  
  CONSTRAINT "PKBONBRG" PRIMARY KEY ("BONJUAL", "KODEBRG")  
);
```

6. Pembuatan Tabel Customer

```
CREATE TABLE "CUSTOMER"  
(  
  "KODECUST" CHAR(10) NOT NULL,  
  "NAMACUST" CHAR(25),  
  "ALAMATCUST" CHAR(25),  
  "TELEPON" CHAR(12),  
  CONSTRAINT "PKKODECUST" PRIMARY KEY ("KODECUST")  
);
```

7. Pembuatan Tabel Supplier

```
CREATE TABLE "SUPPLIER"  
(  
  "KODESUPP" CHAR(10) NOT NULL,  
  "NAMASUPP" CHAR(25),  
  "ALAMATSUPP" CHAR(25),  
  "TELEPON" CHAR(12),  
  CONSTRAINT "PKKODESUPP" PRIMARY KEY ("KODESUPP")  
);
```

);

Pembuatan Program:

a. Pembuatan form awal:

```
unit UMenu;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, Menus, jpeg, ExtCtrls, XPMan, ComCtrls;
type
TfrmToko = class(TForm)
    MainMenu1: TMainMenu;
    Data1: TMenuItem;
    Barang1: TMenuItem;
    Customer1: TMenuItem;
    Supplier1: TMenuItem;
    ransaksi1: TMenuItem;
    PembelianBarang1: TMenuItem;
    PenjualanBarang1: TMenuItem;
    Laporan1: TMenuItem;
    jual1: TMenuItem;
    Beli1: TMenuItem;
    Keluar1: TMenuItem;
    JualBeli1: TMenuItem;
    JualBeli2: TMenuItem;
    XPManifest1: TXPManifest;
    Keluar2: TMenuItem;
    Timer1Timer: TTimer;
    Timer2Timer: TTimer;
    StatusBar1: TStatusBar;
    Image1: TImage;
    Beli2: TMenuItem;
    procedure Keluar1Click(Sender: TObject);
    procedure Barang1Click(Sender: TObject);
    procedure Customer1Click(Sender: TObject);
    procedure Supplier1Click(Sender: TObject);
    procedure PembelianBarang1Click(Sender: TObject);
    procedure PenjualanBarang1Click(Sender: TObject);
    procedure jual1Click(Sender: TObject);
    procedure Beli1Click(Sender: TObject);
    procedure JualBeli1Click(Sender: TObject);
    procedure JualBeli2Click(Sender: TObject);
    procedure Keluar2Click(Sender: TObject);
    procedure Timer1TimerTimer(Sender: TObject);
    procedure Timer2TimerTimer(Sender: TObject);
    procedure FormCreate(Sender: TObject);
```

```

    procedure Image1Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmToko: TfrmToko;
implementation
uses UBarang, UCustomer, USupplier, UJual, UBeli, UlaporanBarang,
    ULaporanCustomer, ULaporanSupplier, ULaporanJualBeli;
{$R *.dfm}
procedure TfrmToko.Keluar1Click(Sender: TObject);
begin
close;
end;
procedure TfrmToko.Barang1Click(Sender: TObject);
begin
frmBarang.Show;
end;
procedure TfrmToko.Customer1Click(Sender: TObject);
begin
frmCustomer.show;
end;
procedure TfrmToko.Supplier1Click(Sender: TObject);
begin
frmSupplier.show;
end;
procedure TfrmToko.PembelianBarang1Click(Sender: TObject);
begin
frmBeli.show;
end;
procedure TfrmToko.PenjualanBarang1Click(Sender: TObject);
begin
frmJual.show;
end;
procedure TfrmToko.jual1Click(Sender: TObject);
begin
frmLaporanBarang.show;
end;
procedure TfrmToko.Beli1Click(Sender: TObject);
begin
frmLaporanCustomer.show;
end;
procedure TfrmToko.JualBeli1Click(Sender: TObject);

```

```

begin
frmLaporanSupplier.show;
end;
procedure TfrmToko.JualBeli2Click(Sender: TObject);
begin
frmLaporanJualBeli.show;
end;
procedure TfrmToko.Keluar2Click(Sender: TObject);
begin
close;
end;
procedure TfrmToko.Timer1TimerTimer(Sender: TObject);
begin
// Perintah Untuk Mejalankan Jam dan Tanggal pada Status Bar
StatusBar1.Panels[1].Text := FormatDateTime('dd mmmm yyyy',now);
StatusBar1.Panels[2].Text := FormatDateTime('hh:mm:ss', now);
end;
procedure TfrmToko.Timer2TimerTimer(Sender: TObject);
var
Temp:String;
begin
Temp:= Application.Title;
Temp:= Temp+Temp[1];
Temp:= Copy(Temp,2,length(Temp)-1);
Application.Title:= Temp;
frmToko.Caption:= Temp;
end;
procedure TfrmToko.FormCreate(Sender: TObject);
begin
Application.Title:='Program Jual Beli Distro   ';
//Title:=Application.Title;
end;
end.

```

b. Pembuatan Form Module:

```

unit UModule;

```

```

interface

```

```

uses

```

```

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

```

```

type

```

```

Tdm = class(TDataModule)

```

```

dbToko: TIBDatabase;

```

```

trToko: TIBTransaction;

```

```

qData: TIBQuery;

```

```
qProses: TIBQuery;  
qsData: TDataSource;  
qsProses: TDataSource;  
private  
  { Private declarations }  
public  
  { Public declarations }  
end;
```

```
var  
  dm: Tdm;
```

```
implementation
```

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

```
end.
```

c. Pembuatan program data Barang:

```
unit UBarang;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
  Dialogs, Grids, DBGrids, StdCtrls, ExtCtrls, Buttons;
```

```
type
```

```
TfrmBarang = class(TForm)  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  Label4: TLabel;  
  Label5: TLabel;  
  Label6: TLabel;  
  edKodeBrg: TEdit;  
  edNamaBrg: TEdit;  
  edMerkBrg: TEdit;  
  edHrgJual: TEdit;  
  edHrgBeli: TEdit;  
  edCari: TEdit;  
  btSimpan: TButton;  
  btHapus: TButton;  
  dbgCari: TDBGrid;  
  rdgCari: TRadioGroup;  
  Label7: TLabel;
```

```

edStok: TEdit;
btCari: TButton;
BitKeluar: TBitBtn;
procedure btKeluarClick(Sender: TObject);
procedure btSimpanClick(Sender: TObject);
procedure btHapusClick(Sender: TObject);
// procedure btCariClick(Sender: TObject);
procedure edCariChange(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure btCariClick(Sender: TObject);
procedure BitKeluarClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmBarang: TfrmBarang;

implementation

uses UModule;

{$R *.dfm}

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

procedure TfrmBarang.btSimpanClick(Sender: TObject);
begin
  With dm.qData Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Insert Into Barang(KodeBrg>NamaBrg>Stok>MerkBrg>HrgJual>HrgBeli)'+
    'Values(:kb,:nb,:st,:mb,:hj,:hb)');

    Prepare;

    ParamByName('kb').AsString:=edKodeBrg.Text;
    ParamByName('nb').AsString:=edNamaBrg.Text;
    ParamByName('mb').AsString:=edMerkBrg.Text;
    ParamByName('st').AsString := EdStok.Text;

```

```
ParamByName('hj').AsFloat:=StrToFloat(edHrgJual.Text);
ParamByName('hb').AsFloat:=StrToFloat(edHrgBeli.Text);
```

```
ExecSql;
```

```
end;
dm.trToko.Commit;
```

```
edKodeBrg.Clear;
edNamaBrg.Clear;
edMerkBrg.Clear;
EdStok.Clear;
edHrgJual.Text:='0';
edHrgBeli.Text:='0';
```

```
edKodeBrg.SetFocus;
```

```
with dm.qProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select *from Barang');
Open;
end;
end;
```

```
procedure TfrmBarang.btHapusClick(Sender: TObject);
```

```
begin
With dm.qProses do
Begin
Close;
Sql.Clear;
Sql.Add('Delete From Barang');
Sql.Add('Where KodeBrg=:kb');
Prepare;
ParamByName('kb').AsString := edKodeBrg.Text;
ExecSql;
End;
```

```
dm.trToko.Commit;
```

```
With dm.qProses Do
Begin
Close;
```



```
Sql.Clear;
Sql.Add('Select *From Barang');
Open;
End;
```

```
end;
```

```
{procedure TfrmBarang.btCariClick(Sender: TObject);
begin
```

```
    With dm.qProses Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select KodeBrg,NamaBrg,MerkBrg,hrgJual,hrgBeli from Barang');
    Sql.Add('Where KodeBrg Like :kb');
    Prepare;
    ParamByName('kb').AsString := '%'+edKodeBrg.Text+'%';
    Open;
    End;
end; }
```

```
procedure TfrmBarang.edCariChange(Sender: TObject);
begin
```

```
With dm.qProses Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select *From Barang');

    //Pencarian berdasarkan Kode Barang
    If rdgCari.ItemIndex = 0 Then
    Begin
    Sql.Add('Where KodeBrg Like :kb');
    Prepare;
    ParamByName('kb').AsString := '%'+edCari.Text+'%';
    End
    Else

    //Pencarian berdasarkan Nama Barang
    If rdgCari.ItemIndex = 1 Then
    Begin
    Sql.Add('Where NamaBrg Like :nb');
```

```

    Prepare;
    ParamByName('nb').AsString := '%'+edCari.Text+'%';
End
Else

//Pencarian berdasarkan Merk Barang
If rdgCari.ItemIndex = 2 Then
Begin
    Sql.Add('Where MerkBrg Like :mb');
    Prepare;
    ParamByName('mb').AsString := '%'+edCari.Text+'%';
End;
    Open;
end;
end;

```

```

procedure TfrmBarang.FormActivate(Sender: TObject);
begin
With dm.qProses Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select *From Barang');
    Open;
    end;
end;

```

```

procedure TfrmBarang.btCariClick(Sender: TObject);
begin
with dm.qProses Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select *from Barang');
    Sql.Add('Where KodeBrg Like :kb');
    Prepare;
    ParamByName('kb').AsString:='%'+edKodeBrg.Text+'%';
    Open;
    end;
end;

```

```

procedure TfrmBarang.BitKeluarClick(Sender: TObject);
begin

```

```
close;  
end;
```

```
end.
```

d. Pembuatan program menu data customer:

```
unit UCustomer;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmCustomer = class(TForm)
```

```
Label1: TLabel;
```

```
Label2: TLabel;
```

```
Label3: TLabel;
```

```
Label4: TLabel;
```

```
edKodeCust: TEdit;
```

```
edNamaCust: TEdit;
```

```
edAlamat: TEdit;
```

```
edTlp: TEdit;
```

```
dbgCustomer: TDBGrid;
```

```
rdgCari: TRadioGroup;
```

```
lblCari: TLabel;
```

```
edCari: TEdit;
```

```
btSimpan: TButton;
```

```
btHapus: TButton;
```

```
btCari: TButton;
```

```
btnClose: TBitBtn;
```

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

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

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

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

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

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

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

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

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```

var
  frmCustomer: TfrmCustomer;

implementation

uses UModule;

{$R *.dfm}

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

procedure TfrmCustomer.edCariChange(Sender: TObject);
begin
  With dm.qProses Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select * From Customer');

      //Pencarian berdasarkan kode customer
      If rdgCari.ItemIndex = 0 Then
        Begin
          Sql.Add('Where KodeCust Like :kc');
          Prepare;
          ParamByName('kc').AsString := '%'+edCari.Text+'%';
        End
      Else

      //Pencarian berdasarkan nama customer
      If rdgCari.ItemIndex = 1 Then
        Begin
          Sql.Add('Where NamaCust Like :nc');
          Prepare;
          ParamByName('nc').AsString := '%'+edCari.Text+'%';
        End;
      Open;
    end;
  end;

procedure TfrmCustomer.btSimpanClick(Sender: TObject);
begin
  With dm.qData Do

```

```

Begin
Close;
Sql.Clear;
Sql.Add('Insert Into Customer(KodeCust, NamaCust, AlamatCust, Telepon) '+
'Values(:kc, :nc, :al, :tl)');
Prepare;
ParamByName('kc').AsString := edKodeCust.Text;
ParamByname('nc').AsString := edNamaCust.Text;
ParamByName('al').AsString := edAlamat.Text;
ParamByName('tl').AsString := edTlp.Text;
ExecSql;
End;

```

```
dm.trToko.Commit;
```

```

edKodeCust.Clear;
edNamaCust.Clear;
edAlamat.Clear;
edTlp.Clear;
edKodeCust.SetFocus;

```

```

With dm.qProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select *From Customer');
Open;
End;
end;

```

```
procedure TfrmCustomer.btHapusClick(Sender: TObject);
```

```

begin
With dm.qProses do
Begin
Close;
Sql.Clear;
Sql.Add('Delete From Customer');
Sql.Add('Where KodeCust=:kc');
Prepare;
ParamByName('kc').AsString := edKodecust.Text;
ExecSql;
End;

```

```
dm.trToko.Commit;
```

```
With dm.qProses Do
```

```
Begin
Close;
Sql.Clear;
Sql.Add('Select *From Customer');
Open;
End;
```

```
end;
```

```
procedure TfrmCustomer.btCariClick(Sender: TObject);
begin
With dm.qProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select * From Customer');
Sql.Add('Where KodeCust Like :kc');
Prepare;
//ParamByName('kb').AsString := edKodeBrg.Text;
ParamByName('kc').AsString := '%'+edKodeCust.Text+'%';
Open;
End;
end;
```

```
procedure TfrmCustomer.FormActivate(Sender: TObject);
begin
With dm.qProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select *From Customer');
Open;
End;
end;
```

```
procedure TfrmCustomer.btCetakClick(Sender: TObject);
begin
{
with QCustomer Do
Begin
Close;
Prepare;
ParamByName('kc').AsString := '%'+EdKodeCust.Text+'%';
//Trim(EdKodeCust.Text);
Open;
```

```
    End;  
    }  
//rvLapCust.Execute;  
end;
```

```
procedure TfrmCustomer.btnCloseClick(Sender: TObject);  
begin  
close;  
end;
```

```
end.
```

e. Pembuatan daftar menu Supplier:

```
unit USupplier;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmSupplier = class(TForm)  
Label1: TLabel;  
Label2: TLabel;  
Label3: TLabel;  
Label6: TLabel;  
edKodeSupp: TEdit;  
edNamaSupp: TEdit;  
edAlamatSupp: TEdit;  
edCari: TEdit;  
rdgCari: TRadioGroup;  
dbgSupplier: TDBGrid;  
btSimpan: TButton;  
btHapus: TButton;  
btCari: TButton;  
Label4: TLabel;  
edTelepon: TEdit;  
btnClose: TBitBtn;  
procedure edCariChange(Sender: TObject);  
procedure btSimpanClick(Sender: TObject);  
procedure btHapusClick(Sender: TObject);  
procedure btCariClick(Sender: TObject);
```

```

    procedure FormActivate(Sender: TObject);
    procedure btKeluarClick(Sender: TObject);
    procedure btnCloseClick(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmSupplier: TfrmSupplier;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmSupplier.edCariChange(Sender: TObject);
begin
    With dm.qProses Do
        Begin
            Close;
            Sql.Clear;
            Sql.Add('Select * From Supplier');

            //Pencarian berdasarkan kode Supplier
            If rdgCari.ItemIndex = 0 Then
                Begin
                    Sql.Add('Where KodeSupp Like :kc');
                    Prepare;
                    ParamByName('ks').AsString := '%'+edCari.Text+'%';
                End
            Else

            //Pencarian berdasarkan nama Supplier
            If rdgCari.ItemIndex = 1 Then
                Begin
                    Sql.Add('Where NamaSupp Like :ns');
                    Prepare;
                    ParamByName('ns').AsString := '%'+edCari.Text+'%';
                End;
                Open;
            end;
        end;
end;

```



```

procedure TfrmSupplier.btSimpanClick(Sender: TObject);
begin
With dm.qData Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Insert Into Supplier(KodeSupp, NamaSupp, AlamatSupp, Telepon) '+
'Values(:ks, :ns, :al, :tl)');
  Prepare;
  ParamByName('ks').AsString := edKodeSupp.Text;
  ParamByname('ns').AsString := edNamaSupp.Text;
  ParamByName('al').AsString := edAlamatSupp.Text;
  ParamByName('tl').AsString := edTelepon.Text;
  ExecSql;
End;

```

```

dm.trToko.Commit;

```

```

edKodeSupp.Clear;
edNamaSupp.Clear;
edAlamatSupp.Clear;
edTelepon.Clear;
edKodeSupp.SetFocus;

```

```

With dm.qProses Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select *From Supplier');
  Open;
  End;
end;

```

```

procedure TfrmSupplier.btHapusClick(Sender: TObject);
begin
With dm.qProses do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Delete From Supplier');
  Sql.Add('Where KodeSupp=:ks');
  Prepare;
  ParamByName('ks').AsString := edKodeSupp.Text;
  ExecSql;
  End;

```

```
dm.trToko.Commit;
```

```
With dm.qProses Do  
  Begin  
  Close;  
  Sql.Clear;  
  Sql.Add('Select *From Supplier');  
  Open;  
  End;
```

```
end;
```

```
procedure TfrmSupplier.btCariClick(Sender: TObject);  
begin  
  With dm.qProses Do  
    Begin  
    Close;  
    Sql.Clear;  
    Sql.Add('Select * From Supplier');  
    Sql.Add('Where KodeSupp Like :ks');  
    Prepare;  
    //ParamByName('kb').AsString := edKodeBrg.Text;  
    ParamByName('ks').AsString := '%' + edKodeSupp.Text + '%';  
    Open;  
    End;  
  end;
```

```
procedure TfrmSupplier.FormActivate(Sender: TObject);  
begin  
  With dm.qProses Do  
    Begin  
    Close;  
    Sql.Clear;  
    Sql.Add('Select *From Supplier');  
    Open;  
    End;  
  end;
```

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

```
procedure TfrmSupplier.btnCloseClick(Sender: TObject);
begin
close;
end;

end.
```

f. Pembuatan program menu penjualan:

```
unit UJual;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmJual = class(TForm)
Label1: TLabel;
edTgl: TEdit;
Label2: TLabel;
edNoBon: TEdit;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
edNamaCust: TEdit;
cmbKodeCust: TComboBox;
edAlamatCust: TEdit;
Label6: TLabel;
edNamabrg: TEdit;
cmbKodeBrg: TComboBox;
Label7: TLabel;
edMerkBrg: TEdit;
Label8: TLabel;
edQty: TEdit;
Label9: TLabel;
edHarga: TEdit;
Label10: TLabel;
dbgJual: TDBGrid;
btTambah: TButton;
btSimpan: TButton;
Label11: TLabel;
edJumlahTotal: TEdit;
btnClose: TBitBtn;
Button1: TButton;
```

```

procedure FormActivate(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure cmbKodeCustChange(Sender: TObject);
Procedure JumlahTotal;
procedure TampilkanItem;
procedure TampilkanSemua;
procedure btkeluarClick(Sender: TObject);
procedure btSimpanClick(Sender: TObject);
procedure btTambahClick(Sender: TObject);
procedure dbgJualDbClick(Sender: TObject);
procedure edNamabrgExit(Sender: TObject);
procedure cmbKodeBrgChange(Sender: TObject);
procedure btnCloseClick(Sender: TObject);
// procedure cmbKodeBrgChange(Sender: TObject);
// procedure edMerkBrgChange(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmJual: TfrmJual;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmJual.FormActivate(Sender: TObject);
begin
  With dm.qData Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select KodeBrg From jualItem');
      Sql.Add('Where BonJual =:Bj');
      Prepare;
      ParamByName('Bj').AsString := Trim(ednoBon.Text);
      Open;
      edNamaCust.Text := Fields[0].AsString;
    End;
    TampilkanSemua;
  end;
end;

```

```

procedure TfrmJual.FormShow(Sender: TObject);
begin
  //Ambil nilai bon akhir lalu tampilkan di edBon
  { With dm.qData Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select Max(BonBeli) From Beli');
      Open;
      ednoBon.Text := FormatFloat('00000',StrToFloat(Fields[0].Value));
    End; }

  //Tampilkan tanggal di edTgl
  edTgl.Text := DateToStr(Date);

  //Tarik data Customer(kode) ke cmbKodeCust
  cmbKodeCust.Clear;
  With dm.QProses Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select Distinct KodeCust From Customer');
      Sql.Add('Order by KodeCust');
      Open;
      while not Eof do
        begin
          cmbKodeCust.Items.Add(Trim(Fields[0].AsString));
          Next;
        end;
      end;

  //Tarik Kode Barang ke cmbKodeBrg
  cmbKodeBrg.Clear;
  With dm.QProses Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select Distinct KODEBrg From Barang');
      //Sql.Add('where stok is null');
      Sql.Add('Order by KodeBrg');
      Open;
      while not Eof do
        begin
          cmbKODEBrg.Items.Add(Trim(Fields[0].AsString));
          Next;
        end;
    End;
  End;

```

end;

TampilkanSemua;

end;

```
procedure TfrmJual.cmbKodeCustChange(Sender: TObject);
begin
  With dm.qData Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select NamaCust,AlamatCust From Customer');
      Sql.Add('Where KodeCust =:kc');
      Prepare;
      ParamByName('kc').AsString :=cmbKodeCust.Text;
      Open;
      edNamaCust.Text := Trim(Fields[0].AsString);
      edAlamatCust.Text := Trim(Fields[1].AsString);
      End;

    If cmbKodeCust.Text <> " Then
      Begin
        With dm.qProses DO
          Begin
            Close;
            Sql.Clear;
            Sql.Add('Update Jual Set KodeCust =:kc');
            Sql.Add('Where BonJual =:bj');
            Prepare;
            ParamByName('bj').AsString := Trim(ednoBon.Text);
            ParamByName('kc').AsString := Trim(cmbKodeCust.Text);
            ExecSql;
            End;

          dm.trToko.Commit
        End
      Else
        Begin
          ShowMessage('Maaf Customer Harus Di Isi ');
          cmbKodeCust.SetFocus;
          end;
        end;
```

```

    TampilkanSemua;

end

procedure TfrmJual.TampilkanItem;
Begin
  With dm.qProses Do
    Begin
      Close;
      Sql.Clear;
      sql.Add('Select JualItem.KodeBrg,Barang.MerkBrg,Barang>NamaBrg,JualItem.Qty,
'+
      'Jualitem.HrgJual, (JualItem.HrgJual*JualItem.Qty) Jumlah From JualItem');
      sql.Add('Inner Join Barang On Barang.KodeBrg=Jualitem.KodeBrg');
      Sql.Add('Where JualItem.BonJual=:BJ');
      Prepare;
      ParamByName('BJ').AsString := Trim(ednoBon.Text);
      Open;
      End;
    end;

  Procedure TfrmJual.TampilkanSemua;
  Begin

  TampilkanItem;
  JumlahTotal;
  end;

  procedure TfrmJual.JumlahTotal;
  Begin
  With dm.qData Do
  Begin
    Close;
    Sql.Clear;
    Sql.Add('Select Sum(Qty*HrgJual) As JumlahTotal From JualItem');
    Sql.Add('Where JualItem.BonJual =:BJ');
    Prepare;
    ParamByName('BJ').AsString := Trim(edNoBon.Text);
    Open;
    edJumlahTotal.Text := FloatToStr(Fields[0].AsFloat);
    End;
  end;

  procedure TfrmJual.btkeluarClick(Sender: TObject);
  begin

```

```

close;
end;

procedure TfrmJual.btSimpanClick(Sender: TObject);
var stok : double;
begin
//Masukkan data kedalam tabel jualitem dengan data yang tercantum
//di masing editbox
  With dm.qData Do
    Begin
      Close;
      sql.Clear;
      Sql.Add('Insert Into JualItem(BonJual,KodeBrg,HrgJual,Qty) '+
        'Values (:bj, :kb, :Hj, :QY)');
      Prepare;
      ParamByName('bj').AsString := Trim(ednoBon.Text);
      ParamByName('kb').AsString := Trim(cmbKodeBrg.text);
      ParamByName('Hj').AsFloat := StrToFloat(edHarga.Text);
      ParamByName('QY').AsFloat := StrToFloat(edQty.Text);
      ExecSql;
    End;

//Untuk mengurangi jumlah stok barang
  With dm.qData Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select Stok from Barang');
      Sql.Add('Where KodeBrg =:kb');
      Prepare;
      Parambyname('kb').AsString := cmbKodeBrg.Text;

      Open;

      dm.qData.Close;
      dm.qData.SQL.Clear;
      dm.qData.SQL.Add('Update Barang Set Stok=:st');
      dm.qData.SQL.Add('Where KodeBrg =:kb');
      Prepare;

      Parambyname('kb').AsString := cmbKodeBrg.Text;
      ParamByName('st').AsString := '0';
      ExecSql;
    end;
end;

```



```

    { cmbKodeBrg.Clear;
With dm.QProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Distinct KodeBrg From Barang');
Sql.Add('where stok = 1');
Sql.Add('Order by KodeBrg');
Open;
while not Eof do
begin
    cmbKodeBrg.Items.Add(Trim(Fields[0].AsString));
    Next;
end; }

```

```

    dm.trToko.Commit;
    edNamaBrg.Text :=Trim("");
    edMerkBrg.Text := Trim("");
    edHarga.clear;
    cmbKodeBrg.Text := Trim("");
    edQty.Clear;
    TampilkanSemua;
end;

```

```

procedure TfrmJual.btTambahClick(Sender: TObject);
begin
//Tampilkan BonJual (No Bon Jual)
With dm.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Select BONJUAL From JUAL');
Open;

//Jika Ternyata tidak ada (kosong/null) maka
If Fields[0].IsNull Then
Begin
    EdnoBon.Text := '00001';
End
Else
    //Jika ada nilainya maka, tentukan nobon paling akhir
    Begin
        With dm.qData Do
            Begin

```

```

    Close;
    Sql.Clear;
    Sql.Add('Select Max(BONJual) From JUAL');
    Open;
    //Ini adalah format agar diawali dengan 0000
    EdnoBon.Text := FormatFloat('00000',StrToFloat(Fields[0].Value+1));

    End;
    End;
End;

```

```

//Simpan nobon tersebut kedalam tabel jual ke field Bonjual
With dm.qProses Do
Begin
    Close;
    Sql.Clear;
    Sql.Add('Insert Into JUAL(BONJual,TGL) Values(:bj,:tg)');
    Prepare;
    ParamByname('bj').AsString := EdnoBon.Text;
    ParamByName('tg').AsDate := StrToDate(EdTgl.Text);
    ExecSql;
end;
dm.TrToko.Commit;

```

```

    CmbKodeCust.Text := Trim("");
    EdNamaCust.Clear;
    EdAlamatCust.Clear;

```

```

    TampilkanSemua;
end;

```

```

procedure TfrmJual.dbgJualDbClick(Sender: TObject);
var stok : double;
begin
If      MessageDlg('Benar      Mau      Hapus      Data      Ini
!',mtConfirmation,[mbYes,mbNo],0)=mrYes Then
    Begin
    //Untuk menambah jumlah stok barang
    With dm.qData Do
    Begin
    Close;
    Sql.Clear;
    Sql.Add('Select Stok from Barang');
    Sql.Add('Where KodeBrg =:kb');
    Prepare;

```

```

Parambyname('kb').AsString := Trim(dbgJual.Fields[0].AsString);
Open;
dm.qData.Close;
dm.qData.SQL.Clear;
dm.qData.SQL.Add('Update Barang Set Stok=:st');
// dm.qData.SQL.Add('Where KodeBrg =:kb);
Prepare;
Parambyname('kb').AsString := Trim(dbgJual.Fields[0].AsString);
ParamByName('st').value := null;
ExecSql;
end;

//hapus record dari table beliitem dengan no bon jual dan kode barang
//yang sudah ditentukan di dbgJual
With dm.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Delete From JaulItem');
Sql.Add('Where BonJual =:Bj And KodeBrg=:kb');
Prepare;
ParamByName('Bj').AsString := Trim(ednoBon.Text);
ParamByName('kb').AsString := dbgJual.Fields[0].AsString;
ExecSql;
End;

cmbKodeBrg.Clear;
With dm.QProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Distinct KodeBrg From Barang');
Sql.Add('where stok is null');
Sql.Add('Order by KodeBrg');
Open;
while not Eof do
begin
cmbKodeBrg.Items.Add(Trim(Fields[0].AsString));
Next;
end;
end;
dm.trToko.Commit;
TampilkanSemua;
End;

```

end;

```
procedure TfrmJual.edNamabrgExit(Sender: TObject);
begin
//Jika kode barang tidak kosong maka tampilan nama barang dan
//harga jual dari table barang sesuai dengan kodde barang yang
//diminta yang berada di edKodeBrg.Text
if edNamaBrg.Text <>Trim('') Then
Begin
  With dm.qData Do
  Begin
    Close;
    Sql.Clear;
    Sql.Add('Select NamaBrg,HrgJual from Barang');
    Sql.Add('Where KodeBrg =:kb');
    Prepare;
    ParamByName('kb').AsString :=edNamaBrg.Text;
    Open;
    cmbKodeBrg.Text :=Trim(Fields[0].Value);
    edHarga.Text :=Trim(FloatToStr(Fields[1].Value));
    edQty.Text :=Trim('1');
  end;
end ;
end;
```

```
procedure TfrmJual.cmbKodeBrgChange(Sender: TObject);
begin

  With dm.qData Do
  Begin
    Close;
    Sql.Clear;
    Sql.Add('Select NamaBrg, MerkBrg, HrgJual from Barang');
    Sql.Add('Where KodeBrg =:kb');
    Prepare;
    ParamByName('kb').AsString := Trim(cmbKodeBrg.Text);
    Open;
    edNamaBrg.Text := Trim(Fields[0].Value);
    edmerkBrg.Text := Trim(Fields[1].Value);
    edHarga.Text := Trim(FloatToStr(Fields[2].value));
    edQty.Text := Trim('1');
  End;
  TampilkanSemua;
```

end;

```
procedure TfrmJual.btnCloseClick(Sender: TObject);  
begin  
close;  
end;
```

end.

g. Pembuatan program pembelian:

```
unit UJual;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmJual = class(TForm)  
Label1: TLabel;  
edTgl: TEdit;  
Label2: TLabel;  
edNoBon: TEdit;  
Label3: TLabel;  
Label4: TLabel;  
Label5: TLabel;  
edNamaCust: TEdit;  
cmbKodeCust: TComboBox;  
edAlamatCust: TEdit;  
Label6: TLabel;  
edNamabrg: TEdit;  
cmbKodeBrg: TComboBox;  
Label7: TLabel;  
edMerkBrg: TEdit;  
Label8: TLabel;  
edQty: TEdit;  
Label9: TLabel;  
edHarga: TEdit;  
Label10: TLabel;  
dbgJual: TDBGrid;  
btTambah: TButton;
```

```

btSimpan: TButton;
Label11: TLabel;
edJumlahTotal: TEdit;
btnClose: TBitBtn;
btHapus: TButton;
btCari: TButton;
edDiskon: TEdit;
Diskon: TLabel;
Label12: TLabel;
Label13: TLabel;
edBayar: TEdit;
    edKembali: TEdit;
Hitung: TButton;
RadioGroup1: TRadioGroup;
edCari: TEdit;
Label14: TLabel;
procedure FormActivate(Sender: TObject);
    procedure FormShow(Sender: TObject);
procedure cmbKodeCustChange(Sender: TObject);
Procedure JumlahTotal;
procedure TampilkanItem;
procedure TampilkanSemua;

procedure btkeluarClick(Sender: TObject);
procedure btSimpanClick(Sender: TObject);
procedure btTambahClick(Sender: TObject);
procedure dbgJualDbClick(Sender: TObject);
    procedure edNamabrgExit(Sender: TObject);
procedure cmbKodeBrgChange(Sender: TObject);
procedure btnCloseClick(Sender: TObject);
procedure HitungClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
// procedure cmbKodeBrgChange(Sender: TObject);
// procedure edMerkBrgChange(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmJual: TfrmJual;

implementation

uses UModule;

```

```
{SR *.dfm}
```

```
procedure TfrmJual.FormActivate(Sender: TObject);  
begin  
  With dm.qData Do  
    Begin  
      Close;  
      Sql.Clear;  
      Sql.Add('Select KodeBrg From jualItem');  
      Sql.Add('Where BonJual =:Bj');  
      Prepare;  
      ParamByName('Bj').AsString := Trim(ednoBon.Text);  
      Open;  
      edNamaCust.Text := Fields[0].AsString;  
    End;  
    TampilkanSemua;  
  end;  
end;
```

```
procedure TfrmJual.FormShow(Sender: TObject);  
begin  
  
  //Tampilkan tanggal di edTgl  
  edTgl.Text := DateToStr(Date);  
  
  //Tarik data Customer(kode) ke cmbKodeCust  
  cmbKodeCust.Clear;  
  With dm.QProses Do  
    Begin  
      Close;  
      Sql.Clear;  
      Sql.Add('Select Distinct KodeCust From Customer');  
      Sql.Add('Order by KodeCust');  
      Open;  
      while not Eof do  
        begin  
          cmbKodeCust.Items.Add(Trim(Fields[0].AsString));  
        Next;  
        end;  
      end;  
  
      //Tarik Kode Barang ke cmbKodeBrg  
      cmbKodeBrg.Clear;  
      With dm.QProses Do  
        Begin  
          Close;
```

```

Sql.Clear;
Sql.Add('Select Distinct KodeBrg From Barang');
//Sql.Add('where stok is null');
Sql.Add('Order by KodeBrg');
Open;
while not Eof do
  begin
    cmbKodeBrg.Items.Add(Trim(Fields[0].AsString));
    Next;
  end;
end;

```

TampilkanSemua;

```

End;
procedure TfrmJual.cmbKodeCustChange(Sender: TObject);
begin
  With dm.qData Do
    Begin
      Close;
      Sql.Clear;
      Sql.Add('Select NamaCust,AlamatCust From Customer');
      Sql.Add('Where KodeCust =:kc');
      Prepare;
      ParamByName('kc').AsString :=cmbKodeCust.Text;
      Open;
      edNamaCust.Text := Trim(Fields[0].AsString);
      edAlamatCust.Text := Trim(Fields[1].AsString);
      End;

    If cmbKodeCust.Text <> " Then
      Begin
        With dm.qProses DO
          Begin
            Close;
            Sql.Clear;
            Sql.Add('Update Jual Set KodeCust =:kc');
            Sql.Add('Where BonJual =:bj');
            Prepare;
            ParamByName('bj').AsString := Trim(ednoBon.Text);
            ParamByName('kc').AsString := Trim(cmbKodeCust.Text);
            ExecSql;
            End;

            dm.trToko.Commit
          End
        End;
      End;
    End;
end;

```



```

Else
  Begin
    ShowMessage('Maaf Customer Harus Di Isi ');
    cmbKodeCust.SetFocus;
  end;

  TampilkanSemua;

end;
procedure TfrmJual.TampilkanItem;
Begin
  With dm.qProses Do
  Begin
    Close;
    Sql.Clear;
    sql.Add('Select
jualItem.KodeBrg,Barang.MerkBrg,Barang>NamaBrg,jualItem.Qty, '+
'Jualitem.HrgJual, ((JualItem.HrgJual*JualItem.Qty)* 0.9) Jumlah From
JualItem');
    sql.Add('Inner Join Barang On Barang.KodeBrg=Jualitem.KodeBrg');
    Sql.Add('Where JualItem.BonJual=:BJ');
    Prepare;
    ParamByName('BJ').AsString := Trim(edNoBon.Text);
    Open;
  End;
end;

Procedure TfrmJual.TampilkanSemua;
Begin

TampilkanItem;
JumlahTotal;
end;

procedure TfrmJual.JumlahTotal;
Begin
  With dm.qData Do
  Begin
    Close;
    Sql.Clear;
    Sql.Add('Select Sum(Qty*HrgJual)*0.9 As JumlahTotal From JualItem');
    Sql.Add('Where JualItem.BonJual =:BJ');
    Prepare;
    ParamByName('BJ').AsString := Trim(edNoBon.Text);
    Open;
    edJumlahTotal.Text := FloatToStr(Fields[0].AsFloat);
  End;
end;

```

```
End;  
end;
```

```
procedure TfrmJual.btkeluarClick(Sender: TObject);  
begin  
close;  
end;
```

```
procedure TfrmJual.btSimpanClick(Sender: TObject);  
var stok : double;  
begin  
//Masukkan data kedalam tabel jualitem dengan data yang tercantum  
//di masing editbox  
With dm.qData Do  
Begin  
Close;  
sql.Clear;  
Sql.Add('Insert Into JualItem(BonJual,KodeBrg,HrgJual,Qty,diskon) '+  
'Values (:bj, :kb, :Hj, :QY ,:ds)');  
Prepare;  
ParamByName('bj').AsString := Trim(ednoBon.Text);  
ParamByName('kb').AsString := Trim(cmbKodeBrg.text);  
ParamByName('Hj').AsFloat := StrToFloat(edHarga.Text);  
ParamByName('QY').AsFloat := StrToFloat(edQty.Text);
```

```
ExecSql;  
End;
```

```
//Untuk mengurangi jumlah stok barang  
With dm.qData Do  
Begin  
Close;  
Sql.Clear;  
Sql.Add('Select Stok from Barang');  
Sql.Add('Where KodeBrg =:kb');  
Prepare;  
Parambyname('kb').AsString := cmbKodeBrg.Text;
```

```
Open;
```

```
dm.qData.Close;  
dm.qData.SQL.Clear;  
dm.qData.SQL.Add('Update Barang Set Stok=:st');
```

```

        dm.qData.SQL.Add('Where KodeBrg =:kb');
        Prepare;

        ParamByname('kb').AsString := cmbKodeBrg.Text;
        ParamByName('st').AsString := '0';
        ExecSql;
        End;

dm.trToko.Commit;
edNamaBrg.Text :=Trim("");
edMerkBrg.Text := Trim("");
edHarga.clear;
cmbKodeBrg.Text := Trim("");
edQty.Clear;
eddiskon.Clear;

        TampilkanSemua;
    end;

procedure TfrmJual.btTambahClick(Sender: TObject);
begin
//Tampilkan BonJual (No Bon Jual)
With dm.qData Do
    Begin
        Close;
        Sql.Clear;
        Sql.Add('Select BONJUAL From JUAL');
        Open;

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

```

```

        End;
    End;
End;

//Simpan nobon tersebut kedalam tabel jual ke field Bonjual
With dm.qProses Do
    Begin
    Close;
Sql.Clear;
        Sql.Add('Insert Into JUAL(BONJual,TGL) Values(:bj,:tg)');
        Prepare;
        ParamByname('bj').AsString := EdnoBon.Text;
        ParamByName('tg').AsDate := StrToDate(EdTgl.Text);
        ExecSql;
        end;
        dm.TrToko.Commit;

        CmbKodeCust.Text := Trim("");
        EdNamaCust.Clear;
        EdAlamatCust.Clear;

        TampilkanSemua;
    end;

procedure TfrmJual.dbgJualDbfClick(Sender: TObject);
var stok : double;
begin
    If      MessageDlg('Benar      Mau      Hapus      Data      Ini
!',mtConfirmation,[mbYes,mbNo],0)=mrYes Then
    Begin
//Untuk menambah jumlah stok barang
        With dm.qData Do
        Begin
        Close;
        Sql.Clear;
        Sql.Add('Select Stok from Barang');
        Sql.Add('Where KodeBrg =:kb');
        Prepare;
        Parambyname('kb').AsString := Trim(dbgJual.Fields[0].AsString);
        Open;
        dm.qData.Close;
        dm.qData.SQL.Clear;
        dm.qData.SQL.Add('Update Barang Set Stok=:st');
        // dm.qData.SQL.Add('Where KodeBrg =:kb');
        Prepare;

```

```

ParamByname('kb').AsString := Trim(dbgJual.Fields[0].AsString);
ParamByName('st').value := null;
ExecSql;
end;

```

```

//hapus record dari table beliitem dengan no bon jual dan kode barang
//yang sudah ditentukan di dbgJual

```

```

With dm.qData Do
Begin
Close;
Sql.Clear;
Sql.Add('Delete From JaulItem');
Sql.Add('Where BonJual =:Bj And KodeBrg=:kb');
Prepare;
ParamByName('Bj').AsString := Trim(ednoBon.Text);
ParamByName('kb').AsString := dbgJual.Fields[0].AsString;
ExecSql;
End;

```

```

cmbKodeBrg.Clear;
With dm.QProses Do
Begin
Close;
Sql.Clear;
Sql.Add('Select Distinct KodeBrg From Barang');
Sql.Add('where stok is null');
Sql.Add('Order by KodeBrg');
Open;
while not Eof do
begin
cmbKodeBrg.Items.Add(Trim(Fields[0].AsString));
Next;
end;
end;
dm.trToko.Commit;
TampilkanSemua;
End;

```

```

end;

```

```

procedure TfrmJual.edNamabrgExit(Sender: TObject);
begin
//Jika kode barang tidak kosong maka tampilan nama barang dan

```

```

        //harga jual dari table barang sesuai dengan kodde barang yang
        //diminta yang berada di edKodeBrg.Text
        if edNamaBrg.Text <>Trim("") Then
Begin
    With dm.qData Do
        Begin
            Close;
            Sql.Clear;
            Sql.Add('Select NamaBrg,HrgJual from Barang');
Sql.Add('Where KodeBrg =:kb');
Prepare;
ParamByName('kb').AsString :=edNamaBrg.Text;
Open;
    cmbKodeBrg.Text :=Trim(Fields[0].Value);
edHarga.Text :=Trim(FloatToStr(Fields[1].Value));
    edQty.Text :=Trim('1');
edDiskon.Text :=Trim('10');
end;
end ;
end;

```

```

procedure TfrmJual.cmbKodeBrgChange(Sender: TObject);
begin

    With dm.qData Do
        Begin
            Close;
            Sql.Clear;
            Sql.Add('Select NamaBrg, MerkBrg, HrgJual from Barang');
            Sql.Add('Where KodeBrg =:kb');
            Prepare;
            ParamByName('kb').AsString := Trim(cmbKodeBrg.Text);
            Open;
            edNamaBrg.Text := Trim(Fields[0].Value);
            edmerkBrg.Text := Trim(Fields[1].Value);
            edHarga.Text := Trim(FloatToStr(Fields[2].value));
            edQty.Text := Trim('1');
            edDiskon.Text :=trim('10');
        End;
        TampilkanSemua;
    end;

```

```
procedure TfrmJual.btnCloseClick(Sender: TObject);
begin
close;
end;
```

```
procedure TfrmJual.HitungClick(Sender: TObject);
var edbayar,edjumlahtotal,edkembali:real;
begin
```

```
edkembali:=edbayar-edjumlahtotal
end;
```

```
procedure TfrmJual.FormCreate(Sender: TObject);
begin
```

```
end;
```

```
end.
```

H.Pembuatan program laporan barang:

```
unit UlaporanBarang;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmLaporanBarang = class(TForm)
QLBarang: TIBQuery;
RvLBarang: TRvDataSetConnection;
RvLapBarang: TRvProject;
btnClose: TBitBtn;
btnPrint: TBitBtn;
procedure btKeluarClick(Sender: TObject);
procedure btCetakClick(Sender: TObject);
procedure btnPrintClick(Sender: TObject);
procedure btnCloseClick(Sender: TObject);
```

```
private
```

```
{ Private declarations }
```

```
public
```

```
{ Public declarations }
```

```
end;
```

```
var
```

```

    frmLaporanBarang: TfrmLaporanBarang;

implementation

uses UModule;

{$R *.dfm}

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

procedure TfrmLaporanBarang.btCetakClick(Sender: TObject);
begin
{ with QLBarang do
begin
close;
open;
end;

RvLapBarang.execute; }
end;

procedure TfrmLaporanBarang.btnPrintClick(Sender: TObject);
begin
with QLBarang do
begin
close;
open;
end;

RvLapBarang.execute;
end;

procedure TfrmLaporanBarang.btnCloseClick(Sender: TObject);
begin
close;
end;

end.

```

h. Pembuatan Program laporan Customer:

```

unit ULaporanCustomer;

interface

```



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

```
type
  TfrmLaporanCustomer = class(TForm)
    RvLapCust: TRvProject;
    RvLCust: TRvDataSetConnection;
    btnPrint: TBitBtn;
    btnClose: TBitBtn;
    QLCust: TIBQuery;
    procedure btKeluarClick(Sender: TObject);
    procedure btCetakClick(Sender: TObject);
    procedure btnCloseClick(Sender: TObject);
    procedure btnPrintClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
```

```
var
  frmLaporanCustomer: TfrmLaporanCustomer;
```

```
implementation
```

```
uses UModule;
```

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

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

```
procedure TfrmLaporanCustomer.btCetakClick(Sender: TObject);
begin
  begin
    with QLCust do
      begin
        close;
        open;
      end;
  end;
```

```

    RvLapCust.execute;

end;
end;

procedure TfrmLaporanCustomer.btnCloseClick(Sender: TObject);
begin
close;
end;

procedure TfrmLaporanCustomer.btnPrintClick(Sender: TObject);
begin
    with QLCust do
    begin
    close;
    open;
    end;

    RvLapCust.execute;

end;
end.

```

i. Pembuatan Program Laporan Supplier:

```

unit ULaporanSupplier;

interface

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

type
    TfrmLaporanSupplier = class(TForm)
        QLSupp: TIBQuery;
        RvLapSupp: TRvProject;
        RvLCust: TRvDataSetConnection;
        btnClose: TBitBtn;
        btnPrint: TBitBtn;
        procedure btKeluarClick(Sender: TObject);
        procedure btCetakClick(Sender: TObject);
        procedure btnCloseClick(Sender: TObject);
        procedure btnPrintClick(Sender: TObject);
    private
        { Private declarations }
    end;

```

```

public
  { Public declarations }
end;

var
  frmLaporanSupplier: TfrmLaporanSupplier;

implementation

uses UModule;

{$R *.dfm}

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

procedure TfrmLaporanSupplier.btCetakClick(Sender: TObject);
begin
  {begin
  with QLSupp do
  begin
  close;
  open;
  end;

  RvLapSupp.execute;

end; }
end;

procedure TfrmLaporanSupplier.btnCloseClick(Sender: TObject);
begin
  close;
end;

procedure TfrmLaporanSupplier.btnPrintClick(Sender: TObject);
begin
  begin
  with QLSupp do
  begin
  close;
  open;
  end;
end;

```

```
RvLapSupp.execute;
```

```
end;
```

```
end;
```

```
end.
```

K. Pembuatan Program Laporan pembelian:

```
unit ULaporanJualBeli;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmLaporanJualBeli = class(TForm)  
  rgLaporan: TRadioGroup;  
  GroupBox1: TGroupBox;  
  DateTimePicker1: TDateTimePicker;  
  DateTimePicker2: TDateTimePicker;  
  Label1: TLabel;  
  Label2: TLabel;  
  cmbNamaCust: TComboBox;  
  cmbNamaSupp: TComboBox;  
  Label3: TLabel;  
  Label4: TLabel;  
  QLBonJual: TIBQuery;  
  RvLapBonJual: TRvDataSetConnection;  
  RvLapJualBeli: TRvProject;  
  btnClose: TBitBtn;  
  btnPrint: TBitBtn;  
  procedure FormShow(Sender: TObject);  
  procedure btKeluarClick(Sender: TObject);  
  procedure btnCloseClick(Sender: TObject);  
  procedure FormCreate(Sender: TObject);  
private  
  { Private declarations }  
public  
  { Public declarations }  
end;
```

```
var
```

```

frmLaporanJualBeli: TfrmLaporanJualBeli;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmLaporanJualBeli.FormShow(Sender: TObject);
begin
//Tarik Nama customer ke cmbNamaCust
cmbNamaCust.Clear;
With dm.QProses Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select Distinct NamaCust From Customer');
  Sql.Add('Order by NamaCust');
  Open;
  while not Eof do
    begin
      cmbNamaCust.Items.Add(Trim(Fields[0].AsString));
      Next;
    end;
  end;

//Tarik Nama supplier ke cmbNamaSupp
cmbNamaSupp.Clear;
With dm.QProses Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select Distinct NamaSupp From Supplier');
  Sql.Add('Order by NamaSupp');
  Open;
  while not Eof do
    begin
      cmbNamaSupp.Items.Add(Trim(Fields[0].AsString));
      Next;
    end;
  end;
end;

procedure TfrmLaporanJualBeli.btKeluarClick(Sender: TObject);
begin
close;

```

```

end;

procedure TfrmLaporanJualBeli.btnCloseClick(Sender: TObject);
begin
close;
end;

```

K. Pembuatan Program Laporan pembelian:
unit ULaporanJualBeli;

interface

uses

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

type

```

TfrmLaporanJualBeli = class(TForm)
  GroupBox1: TGroupBox;
  DateTimePicker1: TDateTimePicker;
  DateTimePicker2: TDateTimePicker;
  Label1: TLabel;
  Label2: TLabel;
  cmbNamaSupp: TComboBox;
  Label3: TLabel;
  QLBel: TIBQuery;
  RvLapBonBeli: TRvDataSetConnection;
  RvLapJualBeli: TRvProject;
  btnClose: TBitBtn;
  btnPrint: TBitBtn;
  QLBelBONBELI: TIBStringField;
  QLBelTGL: TDateField;
  QLBelKODESUPP: TIBStringField;
  QLBelNAMASUPP: TIBStringField;
  QLBelALAMATSUPP: TIBStringField;
  QLBelKODEBRG: TIBStringField;
  QLBelNAMABRG: TIBStringField;
  QLBelQTY: TIBBCDField;
  QLBelHRGBELI: TIBBCDField;
  QLBelJUMLAH: TIBBCDField;
  procedure FormShow(Sender: TObject);
  procedure btKeluarClick(Sender: TObject);
  procedure btnCloseClick(Sender: TObject);
  procedure btnPrintClick(Sender: TObject);
  procedure FormCreate(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmLaporanJualBeli: TfrmLaporanJualBeli;

implementation

uses UModule;

{$R *.dfm}

procedure TfrmLaporanJualBeli.FormShow(Sender: TObject);
begin
//Tarik Nama customer ke cmbNamaCust
cmbNamaSupp.Clear;
With dm.QProses Do
  Begin
  Close;
  Sql.Clear;
  Sql.Add('Select Distinct NamaSupp From Supplier');
  Sql.Add('Order by NamaSupp');
  Open;
  while not Eof do
    begin
      cmbNamaSupp.Items.Add(Trim(Fields[0].AsString));
      Next;
    end;
  end;
end;

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

procedure TfrmLaporanJualBeli.btnCloseClick(Sender: TObject);
begin
close;
end;

procedure TfrmLaporanJualBeli.btnPrintClick(Sender: TObject);

```

```
begin
with QLBeli do
  begin
  close;
  open;
  end;

  RvLapJualBeli.execute;
end;
```