\*\*\*Menu\*\*\*

unit uMenu;

interface

uses

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

Dialogs, ExtCtrls, acImage, sLabel, StdCtrls, sPanel, Buttons,

sSpeedButton, ComCtrls, sPageControl, ImgList, acAlphaImageList, jpeg;

type

TfMenu = class(TForm)

ImageList16: TsAlphaImageList;

ImageList24: TsAlphaImageList;

ImageList32: TsAlphaImageList;

ImgList\_MultiState: TsAlphaImageList;

ImgList\_Multi16: TsAlphaImageList;

sAlphaImageList1: TsAlphaImageList;

sPageControl1: TsPageControl;

sTabSheet2: TsTabSheet;

sSpeedButton5: TsSpeedButton;

sSpeedButton4: TsSpeedButton;

sSpeedButton2: TsSpeedButton;

sSpeedButton1: TsSpeedButton;

sSpeedButton6: TsSpeedButton;

sSpeedButton7: TsSpeedButton;

sSpeedButton3: TsSpeedButton;

sSpeedButton12: TsSpeedButton;

sPanel4: TsPanel;

Image1: TImage;

sSpeedButton8: TsSpeedButton;

procedure sSpeedButton6Click(Sender: TObject);

procedure sSpeedButton4Click(Sender: TObject);

procedure sSpeedButton5Click(Sender: TObject);

procedure sSpeedButton1Click(Sender: TObject);

procedure sSpeedButton2Click(Sender: TObject);

procedure sSpeedButton8Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

fMenu: TfMenu;

implementation

uses uKategori, uKriteria, uModul, uPenilaian, uPerusahaan, uVektor;

{$R \*.dfm}

procedure TfMenu.sSpeedButton6Click(Sender: TObject);

begin

fKategori.ShowModal;

end;

procedure TfMenu.sSpeedButton4Click(Sender: TObject);

begin

fKriteria.ShowModal;

end;

procedure TfMenu.sSpeedButton5Click(Sender: TObject);

begin

fPerusahaan.ShowModal;

end;

procedure TfMenu.sSpeedButton1Click(Sender: TObject);

begin

fPenilaian.ShowModal;

end;

procedure TfMenu.sSpeedButton2Click(Sender: TObject);

begin

fVektor.Show;

end;

procedure TfMenu.sSpeedButton8Click(Sender: TObject);

begin

Application.Terminate;

end;

end.

\*\*\*Vektor\*\*\*

unit uVektor;

interface

uses

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

Dialogs, StdCtrls, sEdit, sSpinEdit, ComCtrls, sTabControl, ImgList,

acAlphaImageList, sButton, Grids, DBGrids, acDBGrid, DB, DBTables,

Buttons, ExtCtrls, OleCtrls, Crystal\_TLB;

type

TfVektor = class(TForm)

GroupBox1: TGroupBox;

Label1: TLabel;

edit4: TsDecimalSpinEdit;

PG: TPageControl;

PG1: TTabSheet;

PG2: TTabSheet;

PG3: TTabSheet;

Q: TQuery;

dsQ: TDataSource;

sDBGrid1: TsDBGrid;

BitBtn1: TBitBtn;

tbkriteria: TTable;

dskriteria: TDataSource;

edJum: TEdit;

Label2: TLabel;

tbPenilaian: TTable;

dsPenilaian: TDataSource;

qPenilaian: TQuery;

dsQpenilaian: TDataSource;

sDBGrid2: TsDBGrid;

sDBGrid3: TsDBGrid;

Panel1: TPanel;

Panel2: TPanel;

BitBtn2: TBitBtn;

dsQvektorS: TDataSource;

qVektorS: TQuery;

Panel3: TPanel;

sDBGrid4: TsDBGrid;

BitBtn3: TBitBtn;

dsVektorH: TDataSource;

qVektorH: TQuery;

Edit1: TEdit;

q1: TQuery;

dsq1: TDataSource;

BitBtn4: TBitBtn;

BitBtn5: TBitBtn;

rpt: TCrystalReport;

bt: TBitBtn;

BitBtn6: TBitBtn;

procedure PGChange(Sender: TObject);

procedure FormActivate(Sender: TObject);

procedure BitBtn1Click(Sender: TObject);

procedure BitBtn2Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);

procedure BitBtn4Click(Sender: TObject);

procedure BitBtn5Click(Sender: TObject);

procedure btClick(Sender: TObject);

procedure BitBtn6Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

fVektor: TfVektor;

x : Integer;

implementation

uses uMenu, Math, uModul;

{$R \*.dfm}

procedure TfVektor.PGChange(Sender: TObject);

begin

if PG.ActivePageIndex = 0 then

begin

q.SQL.Clear;

Q.SQL.Add('SELECT SUM(bobot) AS jumlah FROM qkriteria');

q.Open;

x := Q['jumlah'];

edJum.Text := IntToStr(x);

q.SQL.Clear;

Q.SQL.Add('SELECT \* FROM qkriteria');

q.Open;

end else

if PG.ActivePageIndex = 1 then

Begin

tbPenilaian.Open;

// '--------------------------------------------------------------

qPenilaian.SQL.Clear;

qPenilaian.SQL.Add('SELECT \* from qPenilaian');

qPenilaian.Open;

qVektorS.SQL.Clear;

qVektorS.SQL.Add('SELECT \* from qvektors');

qVektorS.Open;

// '--------------------------------------------------------------

end;

if PG.ActivePageIndex = 2 then

Begin

//tbPenilaian.Open;

tbPenilaian.open;

// '--------------------------------------------------------------

Q.SQL.Clear;

Q.SQL.Add('SELECT SUM(vektors) AS jumlah FROM qvektors WHERE periode = '+ QuotedStr(edit4.Text));

Q.Open;

edit1.Text := FormatFloat('#.####',Q['jumlah']) ;

qVektorH.SQL.Clear;

qVektorH.SQL.Add('SELECT \* from q\_vektor\_h WHERE periode = '+ QuotedStr(edit4.Text) +'ORDER BY hasil DESC');

qVektorH.Open;

// '--------------------------------------------------------------

end;

end;

procedure TfVektor.FormActivate(Sender: TObject);

begin

tbkriteria.Open;

PG.ActivePageIndex := 0;

PGChange(Sender);

end;

procedure TfVektor.BitBtn1Click(Sender: TObject);

var i,j : Integer;

ada, kode : String;

nilbobot, perbaikan : Single;

begin

q.SQL.Clear;

Q.SQL.Add('SELECT \* FROM qkriteria');

q.Open;

kode := q['id\_kriteria'];

j := Q.RecordCount;

tbkriteria.First;

for i := 1 to j do

Begin

nilbobot := tbkriteria['bobot'];

if (tbkriteria['perbaikan\_bobot'] < 0) then

begin

perbaikan := -1 \* (nilbobot / StrToInt(edJum.text) );

end

else

begin

perbaikan := nilbobot / StrToInt(edJum.text);

end;

//perbaikan := FormatFloat('#.####',perbaikan);

tbkriteria.Edit;

tbkriteria['perbaikan\_bobot'] := FormatFloat('#.####',perbaikan);

tbkriteria.Post;

tbkriteria.Next;

end;

q.SQL.Clear;

Q.SQL.Add('SELECT \* FROM qkriteria');

q.Open;

ShowMessage('Perbaikan Bobot sukses...');

end;

procedure TfVektor.BitBtn2Click(Sender: TObject);

var a,b, i,j : Integer ;

nil\_pb, nil\_VektorS, nilaiKuadrat, THasil : Double;

Total : Variant;

nilaiS : array[1..6] of Double;

begin

q.SQL.Clear;

Q.SQL.Add('SELECT \* FROM qvektors WHERE periode = '+ QuotedStr(edit4.Text));

q.Open;

//kode := q['id\_kriteria'] ;

j := Q.RecordCount;

tbPenilaian.First;

for i := 1 to j do

Begin

q.SQL.Clear;

Q.SQL.Add('SELECT \* FROM qKriteria ');

q.Open;

b := Q.RecordCount;

THasil := 1;

tbkriteria.First; // ulang dari awal kriteria

for a := 1 to 6 do

begin

//Rumus vektor S

nil\_pb := tbkriteria['perbaikan\_bobot'];

nilaiS[a] := Power((tbpenilaian['nilai\_c'+inttostr(a)]),(nil\_pb));

nilaiKuadrat := nilaiS[a];

THasil := THasil \* nilaiS[a];

//--------------------------------------------------------------------

tbPenilaian.Edit;

tbPenilaian['vektors'] := THasil;

tbPenilaian['sc'+IntToStr(a)] := FormatFloat('#.####',nilaiKuadrat); //nilaiKuadrat

tbPenilaian.Post;

tbkriteria.Next;

end; //end for a

//tbPenilaian.Post;

tbPenilaian.Next;

if i = j then

BEGIN

qVektorS.SQL.Clear;

qVektorS.SQL.Add('SELECT \* FROM qvektors WHERE periode = '+ QuotedStr(edit4.Text));

qVektorS.Open;

ShowMessage('sukses...');

exit;

END;

end;//end for i

end;

procedure TfVektor.BitBtn3Click(Sender: TObject);

var a,b,c,d, jumRec : Integer;

Nvektor, Hasil, vektorV, NJumlah : Variant;

kode : String; ada : Boolean;

begin

tbPenilaian.Open;

NJumlah := StrToFloat(edit1.Text);

qVektorH.SQL.Clear;

qVektorH.SQL.Add('SELECT \* FROM q\_vektor\_h WHERE periode = '+ QuotedStr(edit4.Text));

qVektorH.Open;

jumRec := qVektorH.RecordCount;

qVektorH.First;

tbPenilaian.First;

for a := 1 to jumRec do

Begin

kode := qVektorH['id\_perusahaan'];

ada := tbPenilaian.FindKey([kode]);

if ada then

Begin

//Rumus Vektor H

Nvektor := tbPenilaian['vektors'];

vektorV := Nvektor/NJumlah;

//--------------------------------

tbPenilaian.Edit;

tbPenilaian['hasil'] := FormatFloat('#.####',vektorV);

tbPenilaian.Post;

qVektorH.Next;

tbPenilaian.Next;

end; //if

end;//for a

qVektorH.SQL.Clear;

qVektorH.SQL.Add('SELECT \* FROM q\_vektor\_h WHERE periode = '+ QuotedStr(edit4.Text) +'ORDER BY hasil DESC');

qVektorH.Open;

ShowMessage('...Sukses...');

;

end;

procedure TfVektor.BitBtn4Click(Sender: TObject);

begin

rpt.ReportFileName := 'c:\NURRAHMADANI\nilai\_vektor.rpt';

rpt.SelectionFormula := '';

rpt.SelectionFormula := '{q\_vektor\_h.periode}=' +QuotedStr(edit4.Text);

rpt.RetrieveDataFiles;

rpt.Destination := crptToWindow;

rpt.WindowState := crptMaximized;

rpt.Action := 1;

end;

procedure TfVektor.BitBtn5Click(Sender: TObject);

begin

rpt.ReportFileName := 'c:\NURRAHMADANI\nilai\_kriteria.rpt';

rpt.SelectionFormula := '';

rpt.SelectionFormula := '{qPenilaian.periode}=' +QuotedStr(edit4.Text);

rpt.RetrieveDataFiles;

rpt.Destination := crptToWindow;

rpt.WindowState := crptMaximized;

rpt.Action := 1;

end;

procedure TfVektor.btClick(Sender: TObject);

begin

rpt.ReportFileName := 'c:\NURRAHMADANI\kriteria.rpt';

rpt.SelectionFormula := '';

rpt.RetrieveDataFiles;

rpt.Destination := crptToWindow;

rpt.WindowState := crptMaximized;

rpt.Action := 1;

end;

procedure TfVektor.BitBtn6Click(Sender: TObject);

begin

Close;

end;

end.