unit Unit1;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.ExtCtrls,

Vcl.Imaging.jpeg, System.IOUtils;

type

TForm1 = class(TForm)

Button1: TButton;

Image1: TImage;

OpenDialog1: TOpenDialog;

procedure Button1Click(Sender: TObject);

procedure FormActivate(Sender: TObject);

private

{ Private declarations }

procedure CopyFileExample(const SourceFile, DestFile: string);

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{$R \*.dfm}

procedure TForm1.Button1Click(Sender: TObject);

var

DestFileName: string;

begin

// Show the file open dialog

if OpenDialog1.Execute then

begin

// If a file is selected, load the image into Image1

Image1.Picture.LoadFromFile(OpenDialog1.FileName);

// ShowMessage('awdw ada'+OpenDialog1.FileName) ;

DestFileName := TPath.Combine(ExtractFilePath(Application.ExeName), 'Images');

// CopyFileExample(Image1.geti, 'Images');

ShowMessage(DestFileName ) ;

Image1.Picture.SaveToFile(DestFileName )

end;

end;

procedure TForm1.CopyFileExample(const SourceFile, DestFile: string);

begin

try

TFile.Copy(SourceFile, DestFile, True); // True to overwrite the destination file if it exists.

ShowMessage('File copied successfully.');

except

on E: EInOutError do

ShowMessage(Format('I/O Error: %s', [E.Message]));

on E: EAccessViolation do

ShowMessage(Format('Access Violation: %s', [E.Message]));

on E: Exception do

ShowMessage(Format('Error copying file from "%s" to "%s". Error: %s',

[SourceFile, DestFile, E.Message]));

end;

end;

procedure TForm1.FormActivate(Sender: TObject);

begin

// Apply the filter to allow only certain file types to be selected

OpenDialog1.Filter := 'Image Files (\*.bmp;\*.jpg;\*.jpeg;)|\*.BMP;\*.JPG;\*.JPEG;|All Files (\*.\*)|\*.\*';

// Make the image fit in the frame

Image1.Stretch := true;

end;

end.