

الجمهورية الجزائرية الديمقراطية الشعبية  
وزارة الشبيبة والشباب الضائع تقدم

# برمجة برنامج لقراءة الرسائل

## Mail Box

صاحب الكتاب Minou DZ

بتاريخ 08/06/2012



**GMX** CaraMail

حكمة : يجب أن نثق أننا ما خلقنا أبداً لنفشل أو نحرز أو لنكن أناس بلا هدف يجب أن نثق أن وجودنا ليس صدفة وليس رقماً فحسب ، وجودنا لحاجة أنا موجود : لأن الكون يحتاجني .

السلام عليكم بداية وختاما وتحية طيبة للجميع

الحمد لله نحمده ونشكره ونستعين به ونعوذ بالله من شرور أنفسنا

من أضل فلا هادي له ومن هداه فلا مضل له

وأما بعد...

أهلا بكم في هذا الكتيب الصغير

حيث يتناول هذا الكتيب كيفية برنامج قراءة الرسائل باستخدام

برتوكول pop3

لنبدأ بتسجيل الدخول واستيراد جميع الرسائل



## تسجيل الدخول واستيراد الرسائل

```
unit Unit1;  
  
interface  
  
uses  
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
  Dialogs, IdBaseComponent, IdComponent, IdTCPConnection, IdTCPClient,  
  IdExplicitTLSClientServerBase, IdMessageClient, IdPOP3, ComCtrls,  
  StdCtrls, IdMessage, ExtCtrls, jpeg, XPMan, IdAntiFreezeBase,  
  IdAntiFreeze;  
  
type  
  TForm1 = class(TForm)  
    btn1: TButton;  
    GroupBox1: TGroupBox;  
    edt1: TEdit;  
    edt2: TEdit;  
    lbl1: TLabel;  
    lbl2: TLabel;  
    btn2: TButton;  
    stat1: TStatusBar;  
    IdPOP31: TIdPOP3;  
    IdMessage1: TIdMessage;  
    img1: TImage;
```

```

XPManifest1: TXPManifest;
IdAntiFreeze1: TIdAntiFreeze;
procedure btn1Click(Sender: TObject);
procedure IdPOP31Status(ASender: TObject; const AStatus: TIdStatus;
  const AStatusText: String);
procedure btn2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  form1: TForm1;

implementation

uses Unit2;

{$R *.dfm}

procedure TForm1.btn1Click(Sender: TObject); //زر دخول
var i:Integer;
begin
  if IdPOP31.Connected then IdPOP31.Disconnect ;

  try
    IdPOP31.Host := 'pop.gmx.com';
    IdPOP31.Port:=110;
    IdPOP31.Username := edt1.Text;
    IdPOP31.Password := edt2.Text;
    try
      form2.lv1.Clear;
      IdMessage1.Clear;
      form2.Mmo1.Clear;
      IdPOP31.Connect;
      //for i:=IdPOP31.CheckMessages downto 1 do begin

    for i:=1 to IdPOP31.CheckMessages do begin
      IdPOP31.Retrieve(i,IdMessage1) ;
      stat1.Panels[0].Text:='Load : '+inttostr(i)+' / '+inttostr(IdPOP31.CheckMessages) +' messages ..';
      Application.ProcessMessages;
      with form2.lv1.Items.Add do
        begin
          Caption:=IdMessage1.From.Address;
          SubItems.Add(IdMessage1.Recipients.EmailAddresses) ;
          SubItems.Add(IdMessage1.Subject) ;
          SubItems.Add(IdMessage1.Sender.Address) ;
          SubItems.Add(IdMessage1.Body.Text) ;
          SubItems.Add(FormatDateTime( 'yyy-mm-dd hh:mm:ss ',IdMessage1.Date)) ;
          SubItems.Add(IntToStr(IdPOP31.RetrieveMsgSize(i) div 1024) + ' K' ) ;

        end;

      end;
    except on err:Exception do
      stat1.Panels[0].Text:=err.Message;
    end;
  finally
    if IdPOP31.Connected then begin
      form2.Show ;
      form1.Hide;
      Form2.Caption:='Inbox [ '+edt1.Text+' ]';
    end;
  //IdPOP31.Disconnect

```

```

end;
end;

procedure TForm1.IdPOP31Status(ASender: TObject; const AStatus: TIdStatus;
const AStatusText: String);
begin
stat1.Panels[0].Text:= AStatusText;
Application.ProcessMessages;
end;

procedure TForm1.btn2Click(Sender: TObject); // زر خروج
begin
IdPOP31.Disconnect;
end;

end.

```

## الواجهة النهائية للتسجيل الدخول واستيراد الرسائل



## ننتقل إلى واجهة البرنامج الرئيسية والكود سورس بداية

### قراءة وحذف الرسائل

```

unit Unit2;

interface

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

type
TForm2 = class(TForm)

```

```

lbl4: TLabel;
lbl5: TLabel;
mmo1: TMemo;
lv1: TListView;
stat1: TStatusBar;
XPManifest1: TXPManifest;
btn1: TBitBtn;
btn2: TBitBtn;
btn3: TBitBtn;
procedure lv1Click(Sender: TObject);
procedure btn1Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure btn2Click(Sender: TObject);
procedure btn3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses Unit1, Unit3;

{$R *.dfm}

procedure TForm2.lv1Click(Sender: TObject); // لقراءة محتوى الرسالة
begin
  if lv1.ItemIndex > -1 then
  begin
    Mmo1.Text := lv1.Items.Item[lv1.ItemIndex].SubItems.Strings[3] ;
    lbl4.Caption := 'Subject : ' + lv1.Items.Item[lv1.ItemIndex].SubItems.Strings[1] ;
    lbl5.Caption := 'From : ' + lv1.Items.Item[lv1.ItemIndex].Caption
  end;
end;

procedure TForm2.btn1Click(Sender: TObject); // زر حذف
begin
  if lv1.ItemIndex > -1 then begin

  if (Form1.IdPOP31.Retrieve(lv1.ItemIndex+1,Form1.IdMessage1))then
  if Form1.IdPOP31.Connected then
  if form1.IdPOP31.Delete(lv1.ItemIndex+1)then
    lv1.DeleteSelected
  end;
end;

procedure TForm2.FormClose(Sender: TObject; var Action: TCloseAction); // في حدث غلق البرنامج
begin
  form1.IdPOP31.Disconnect;
  Application.Terminate;
end;

procedure TForm2.btn2Click(Sender: TObject); // زر ارسال
begin
  if lv1.ItemIndex > -1 then
  begin
    form3.edt2.text := lv1.Items.Item[lv1.ItemIndex].SubItems.Strings[1] ;
    form3.edt1.text := lv1.Items.Item[lv1.ItemIndex].Caption ;
    Form3.Show;
    Form3.Caption := 'Send [ ' + lv1.Items.Item[lv1.ItemIndex].Caption + ' ]'
  end;
end;

procedure TForm2.btn3Click(Sender: TObject); // زر تحديث check
begin
  lv1.Clear;
  form1.btn1.Click;

```

```

end;

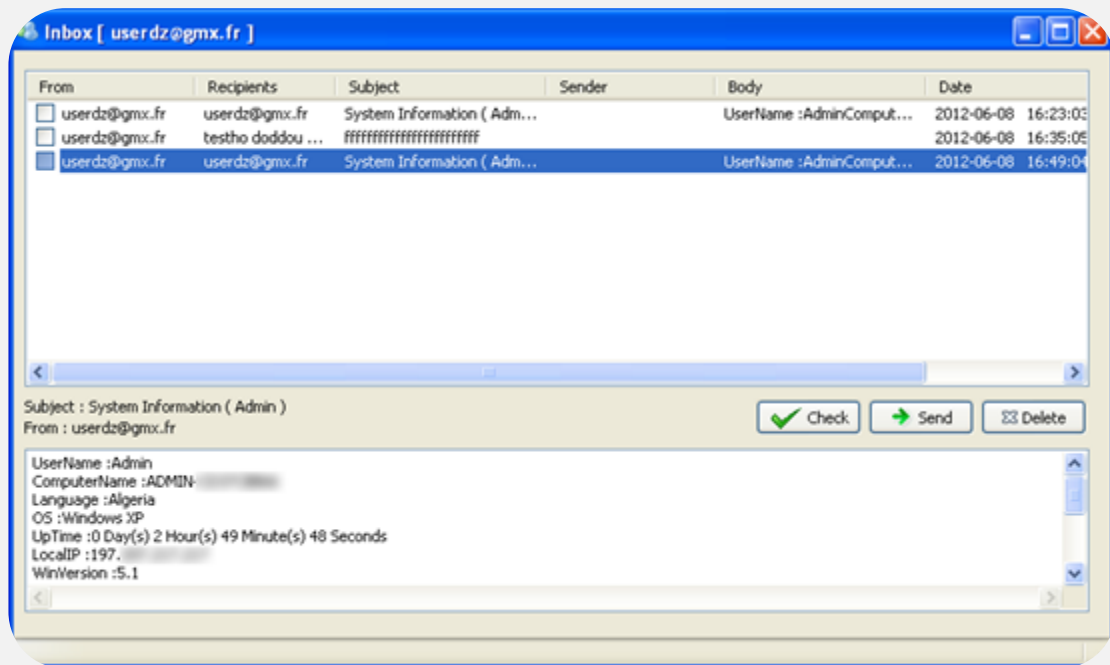
end.

{Memo2.Clear;
Label4.Caption := 'Attachments =>';
for i := 0 to Pred(IdMessage1.MessageParts.Count) do
begin
if(IdMessage1.MessageParts.Items[i] is TIdAttachment) then
begin
fn := p+TIdAttachment(IdMessage1.MessageParts.Items[i]).FileName;
Label4.Caption := Label4.Caption + fn;
try
TIdAttachment(IdMessage1.MessageParts.Items[i]).SaveToFile(fn);

except on E:Exception do}

```

## ونحصل على



## ننتقل إلى برنامج الإرسال



```
unit Unit3;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, jpeg, ExtCtrls, StdCtrls, XPMan, ComCtrls, IdBaseComponent,
  IdComponent, IdTCPConnection, IdTCPClient, IdExplicitTLSClientServerBase,
  IdMessageClient, IdSMTPBase, IdSMTP;

type
  TForm3 = class(TForm)
    img1: TImage;
    stat1: TStatusBar;
    edt1: TEdit;
    edt2: TEdit;
    mmo1: TMemo;
    XPManifest1: TXPManifest;
    btn1: TButton;
    IdSMTP1: TIdSMTP;
    lbl1: TLabel;
    lbl2: TLabel;
    lbl3: TLabel;
    procedure btn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form3: TForm3;

implementation

uses Unit1;

{$R *.dfm}

procedure TForm3.btn1Click(Sender: TObject);
begin
  form1.idmessage1.Clear;
  IdSMTP1.Host:='smtp.gmx.com';
  IdSMTP1.Port:=587;
  IdSMTP1.Username:=form1.edt1.Text;
  IdSMTP1.Password:=form1.edt2.Text;
  form1.IdMessage1.Body.Text:=Mmo1.Text;
  form1.IdMessage1.From.Text:=form1.edt1.text;

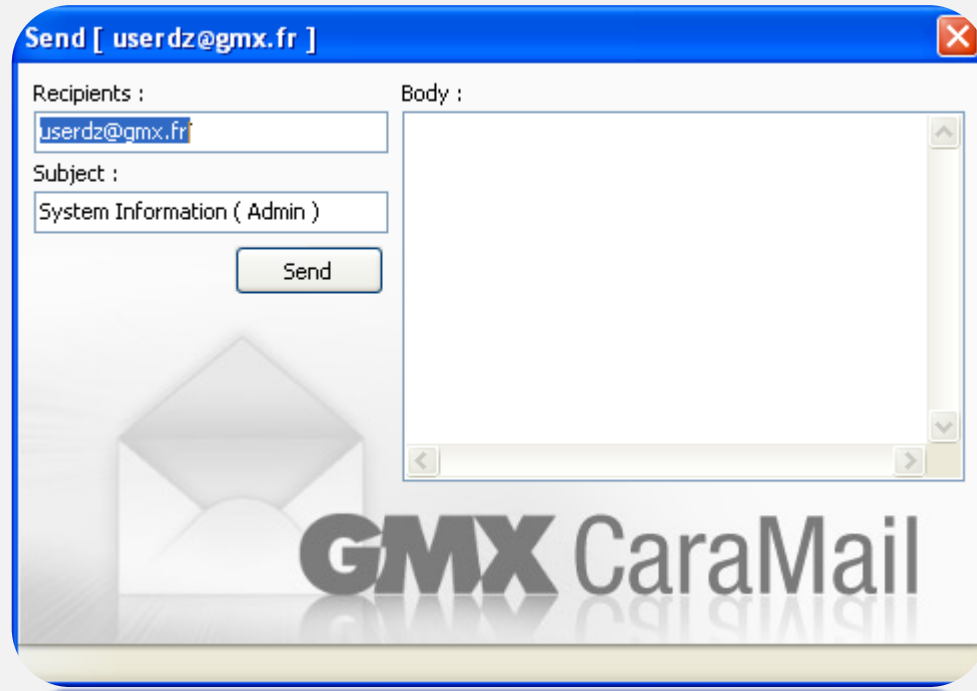
  form1.IdMessage1.Recipients.EmailAddresses:=edt1.text;
  form1.IdMessage1.Subject:=edt2.Text;
  form1.IdMessage1.Date:=Date;

  try
    IdSMTP1.Connect;
  try
    if IdSMTP1.Connected then begin
      stat1.Panels[0].Text:=(TimeToStr(Time)+ ' : '+ 'Please wait while the email is sending...');

      IdSMTP1.Send(form1.IdMessage1);
      stat1.Panels[0].Text:=(TimeToStr(Time)+ ' : '+ 'Mail sent.')
    end;
  except
    on e:Exception do begin
      stat1.Panels[0].Text:=e.Message;
      stat1.Panels[0].Text:=(TimeToStr(Time)+ ' : '+ e.Message)
    end;
  end;
end;
```

```
finally  
IdSMTP1.Disconnect();  
  
end;  
end;  
  
end.
```

حيث واجهته من هذا الشكل



تم إرفاق المشروع كامل

بالتوفيق للجميع



# تم بحمد الله



للتواصل

**minouzd@gmail.com**