---------------------------------------------------------------------------------------------------------------------------------------------------------------

-- access - sending mail from ms access with ms outlook.txt

The url is : http://www.jephens.com/howtoemail.asp for more details.

The Explanation

The code is pretty well commented, but this is what it does in English.

The code opens Outlook and opens your query. It asks you for the subject line to use and the text file to use for the body of the message. The code then creates a message, addresses it and sends it to each person in your list that made it through gauntlet known as "MyEMailAddresses."

Many people keep Outlook running all the time, so this routine will not shut it down when it's done running unless you uncomment the line (remove the ') that says MyOutlook.Quit.

Now, let's get this code into the module and call it a day.

The Code

-- Start Copying Here ---

Public Function SendEMail()

Dim db As DAO.Database

Dim MailList As DAO.Recordset

Dim MyOutlook As Outlook.Application

Dim MyMail As Outlook.MailItem

Dim Subjectline As String

Dim BodyFile As String

Dim fso As FileSystemObject

Dim MyBody As TextStream

Dim MyBodyText As

String

Set fso = New FileSystemObject

' First, we need to know the subject.

' We can't very well be sending around blank messages...

Subjectline$ = InputBox$("Please enter the subject line for this mailing.", \_

"We Need A Subject Line!")

' If there's no subject, call it a day.

If Subjectline$ = "" Then

MsgBox "No subject line, no message." & vbNewLine & vbNewLine & \_

"Quitting...", vbCritical, "E-Mail Merger"

Exit Function

End If

' Now we need to put something in our letter...

BodyFile$ = InputBox$("Please enter the filename of the body of the message.", \_

"We Need A Body!")

' If there's nothing to say, call it a day.

If BodyFile$ = "" Then

MsgBox "No body, no message." & vbNewLine & vbNewLine & \_

"Quitting...", vbCritical, "I Ain't Got No-Body!"

Exit Function

End If

' Check to make sure the file exists...

If fso.FileExists(BodyFile$) = False Then

MsgBox "The body file isn't where you say it is. " & vbNewLine & vbNewLine & \_

"Quitting...", vbCritical, "I Ain't Got No-Body!"

Exit Function

End If

' Since we got a file, we can open it up.

Set MyBody =

fso.OpenTextFile(BodyFile, ForReading, False, TristateUseDefault)

' and read it into a variable.

MyBodyText = MyBody.ReadAll

' and close the file.

MyBody.Close

' Now, we open Outlook for our own device..

Set MyOutlook = New Outlook.Application

' Set up the database and query connections

Set db = CurrentDb()

Set MailList = db.OpenRecordset("MyEmailAddresses")

' now, this is the meat and potatoes.

' this is where we loop through our list of addresses,

' adding them to e-mails and sending them.

Do Until MailList.EOF

' This creates the e-mail

Set MyMail = MyOutlook.CreateItem(olMailItem)

' This addresses it

MyMail.To = MailList("email")

'This gives it a subject

MyMail.Subject = Subjectline$

'This gives it the body

MyMail.Body = MyBodyText

'If you want to send an attachment

'uncomment the following line

'MyMail.Attachments.Add "c:\myfile.txt", olByValue, 1, "My Displayname"

' To briefly describe:

' "c:\myfile.txt" = the file you want to attach

'

' olByVaue = how to pass the file. olByValue attaches it, olByReference creates a shortcut.

' the shortcut only works if the file is available locally (via mapped or local drive)

'

' 1 = the position in the outlook message where to attachment goes. This is ignored by most

' other mailers, so you might want to ignore it too. Using 1 puts the attachment

' first in line.

'

' "My Displayname" = If you don't want the attachment's icon string to be "c:\myfile.txt" you

' can use this property to change it to something useful, i.e. "4th Qtr Report"

'This sends it!

MyMail.Send

'Some people have asked how to see the

e-mail

'instead of automaticially sending it.

'Uncomment the next line

'And comment the "MyMail.Send" line above this.

'MyMail.Display

'And on to the next one...

MailList.MoveNext

Loop

'Cleanup after ourselves

Set MyMail = Nothing

'Uncomment the next line if you want Outlook to shut down when its done.

'Otherwise, it will stay running.

'MyOutlook.Quit

Set MyOutlook = Nothing

MailList.Close

Set MailList = Nothing

db.Close

Set db = Nothing

End

Function

-- End Copying Here ---

---------------------------------------------------------------------------------------------------------------------------------------------------------------

-- MailCode\_Delete duplicate mails from Inbox Outlook code.txt

'Delete duplicate mails from Inbox

'---------------------------------------------------

'delete duplicate mails from Inbox

Public Sub DelDupMailFromInbox()

Dim objOutlook As outlook.Application

Dim objNamespace As outlook.NameSpace

Dim objInbox As outlook.MAPIFolder

Dim objMailbox As outlook.MAPIFolder

Dim objFolder As outlook.MAPIFolder

Dim objItems As outlook.Items

Dim objItms As outlook.Items

Dim strFolderName As String

Dim i As Integer, j As Integer, cnt As Integer

' Instantiate an Outlook Application object.

Set objOutlook = CreateObject("Outlook.Application")

Set objNamespace = objOutlook.GetNamespace("MAPI")

Set objInbox = objNamespace.GetDefaultFolder (6)

'strFolderName = objInbox.Parent

'Set objMailbox = objNamespace.Folders(strFolderName)

'Set objFolder = objMailbox.Folders("ALL")

'Set objItems = objFolder.Items

Set objItems = objInbox.Items

objItems.Sort "[Subject]"

For i = 1 To objItems.Count

DoEvents

DoEvents

DoEvents

If i > objItems.Count Then Exit For

For j = i To objItems.Count

DoEvents

DoEvents

DoEvents

If j > objItems.Count Then Exit For

If StrComp(LCase(objItems.Item(i).Body), LCase(objItems.Item(j).Body)) = 0 And i <> j Then

objItems.Item (j).Delete

cnt = 0

End If

If StrComp(LCase(objItems.Item(i).Body), LCase(objItems.Item(j).Body)) <> 0 Then

cnt = cnt + 1

If cnt > 2 Then Exit For

End If

Next

cnt = 0

Set objItems = objInbox.Items

objItems.Sort "[Subject]"

Next

MsgBox "completed"

End Sub

'end of the DelDupMailFromInbox

'delete duplicate mails from Inbox

'---------------------------------------------------

'---------------------------------------------------

'delete duplicate mails from subfolders

Public Sub DelDupMailFromFolders()

Dim objOutlook As outlook.Application

Dim objNamespace As outlook.NameSpace

Dim objInbox As outlook.MAPIFolder

Dim objMailbox As outlook.MAPIFolder

Dim objFolder As outlook.MAPIFolder

Dim objItems As outlook.Items

Dim objItms As outlook.Items

Dim strFolderName As String

Dim i As Integer, j As Integer, cnt As Integer

' Instantiate an Outlook Application object.

Set objOutlook = CreateObject("Outlook.Application")

Set objNamespace = objOutlook.GetNamespace("MAPI")

Set objInbox = objNamespace.GetDefaultFolder (6)

strFolderName = objInbox.Parent

Set objMailbox = objNamespace.Folders(strFolderName)

Set objFolder = objMailbox.Folders("ALL")

Set objItems = objFolder.Items

objItems.Sort "[Subject]"

For i = 1 To objItems.Count

DoEvents

DoEvents

DoEvents

If i > objItems.Count Then Exit For

For j = i To objItems.Count

DoEvents

DoEvents

DoEvents

If j > objItems.Count Then Exit For

If StrComp(LCase(objItems.Item(i).Body), LCase(objItems.Item(j).Body)) = 0 And i <> j Then

objItems.Item (j).Delete

cnt = 0

End If

If StrComp(LCase(objItems.Item(i).Body), LCase(objItems.Item(j).Body)) <> 0 Then

cnt = cnt + 1

If cnt > 2 Then Exit For

End If

Next

cnt = 0

Set objItems = objFolder.Items

objItems.Sort "[Subject]"

Next

MsgBox "completed"

End Sub

'end of the DelDupMailFromInbox

'delete duplicate mails from subfolders

'---------------------------------------------------

---------------------------------------------------------------------------------------------------------------------------------------------------------------