

Re: Help with Microsoft help on linking Access to Outlook

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.formscoding/2006-09/msg01091.html>

- *From:* JohnB <JohnB@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Sun, 17 Sep 2006 10:27:01 -0700
-

Hi Doug.

I read your reply earlier but gave myself some time to digest all you'd said before I got back.

You said: "If you had a parameter-driven query, that's what you'd use when opening your recordset. The SQL string I gave was a sample of how you can dynamically create SQL statements. The assumption was that you'd pass some additional parameters to the function, indicating how to build the WHERE clause."

meaning your eariler code:

Dim strSQL As String

```
strSQL = "SELECT EmailAddress FROM tblSchool WHERE Town = 'xxxx'"  
Set MyRS = MyDB.OpenRecordset(strSQL)
```

I didn't realise that this was meant to be dynamic. I thought you meant I should build the SQL, using a query, then paste it into your code. I thought that it would not suit my needs because it would be fixed and would select the same filtered recordset every time. Sorry – I should have realised you didn't mean that. So, how would the strSQL be generated dynamically by the user filtering down a recordset?

BTW, I'm fairly familiar with parameters in queries but I've only ever used them in a simplistic way when producing reports. I've found them fairly crude, as I can only present the user with the ability to specify one criteria at a time and they have to remember to enter the exact field contents every time. I guess there is a way to present the user with a combo box to select entries from but I don't know how to do that. Even then, the user wouldn't get to see the filtered recordset before the routine would run, hence my idea of using a normal form to allow the normal filtering process and viewing of the results before clicking on a command button to pass the filtered set to the routine.

So at the moment, I have two options:

Re: Help with Microsoft help on linking Access to Outlook

1. build a simple query with a parameter for the field Town, then use that query as the record source for the routine. That would present a fixed question (e.g. "What Town Name?") to the user every time. Actually, that would suit my needs at the present, although they wouldn't be able to view the resultant filtered set before Outlook opens. Still, it would do.
2. try to fully understand what you mean by dynamically producing the SQL to act as the record source, because presumably that would allow the users to choose themselves what fields to filter on. Much more useful. And maybe this option would allow them to view the filtered recordset and perhaps correct/redo it before it is passed to the routine?

So, to repeat the question – how would the stSQL be generated dynamically by the user filtering down a recordset?

Hopefully I'm understanding you correctly now. Many thanks, again, for the help.

"Douglas J. Steele" wrote:

Sounds as though you may need to do a bit of homework into how to create queries that accept parameters from the user.

If you had a parameter-driven query, that's what you'd use when opening your recordset. The SQL string I gave was a sample of how you can dynamically create SQL statements. The assumption was that you'd pass some additional parameters to the function, indicating how to build the WHERE clause.

--
Doug Steele, Microsoft Access MVP
<http://I.Am/DougSteele>
(no private e-mails, please)

"JohnB" <JohnB@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
<news:B3A8C562-DAC6-479E-8348-5321637DEA36@xxxxxxxxxxxxxxxxxxxx>

Thank you, Doug, for providing such a comprehensive reply and for the additional crucial info.

I just want to focus on your last point, where you said:

"Now, this code is going to send email to every entry in the table. To limit to whom the e-mail gets send, you'll need to create a query that returns only those schools to whom you wish to e-mail, use that query name rather than tblSchool in the OpenRecordset command. Alternatively, you could create

Re: Help with Microsoft help on linking Access to Outlook
an SQL Statement in the routine, and use that to open the recordset:

```
Dim strSQL As String
```

```
strSQL = "SELECT EmailAddress FROM tblSchool WHERE Town =  
'xxxx'"  
Set MyRS = MyDB.OpenRecordset(strSQL)"
```

The thing is, I want to give my users the ability to choose the group of Schools to whom an Email is sent, dynamically. By that I mean they open a form which initially shows all School records, then they filter the recordset to, say, only those Schools in a particular town, then they would click on a command button that calls up the routine to run using the filtered set of records. The next time, they might choose to filter down to those in a different town, or using a different selection criteria all together. Your suggestions, if I understand them correctly, would assume that the set of Schools was fixed and predetermined.

To look at it another way, where a particular line of code in the routine reads:

```
Set MyRS = MyDB.OpenRecordset("tblMailingList")
```

it would instead say something like:

```
Set MyRS = MyDB.OpenRecordset("Something here that points to the forms  
currently filtered recordset")
```

I hope you get the idea.

Again, thanks for taking the time to produce such useful answers.

"Douglas J. Steele" wrote:

"JohnB" <JohnB@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in
message
news:B7AF82EF-BE21-4361-A43C-5E91878BB389@xxxxxxxxxxxxxxxxxxxx

The article provides code for a Module but I don't know how this is used. i.e. how do I call up this code – from the 'on click' event of a command button on the form described? How exactly

Re: Help with Microsoft help on linking Access to Outlook

– what is the Module name? As you can tell, I'm not too hot on the use of modules/macros.

Create a new module (not a Class Module nor a module associated with a form or report) and copy all of the code from step 7 into that new module.

What you name the module when you save it is irrelevant (as long as you don't name it SendMessages, since modules cannot have the same name as functions or subs contained within them)

In your "on click" event (or however you want to invoke it), you'd put a line of code:

Call SendMessages

or

Call SendMessages("C:\TestAttachment.txt")

I disagree stylistically with what they're saying in step 11: I think it's better to use the Call keyword, but the two lines above are equivalent to just using

SendMessages

or

SendMessages "C:\TestAttachment.txt"

(Note that you must enclose the parameter in parentheses when you use the Call keyword, and you don't when not using Call)

Next, I don't understand what is meant by step 11 in the instructions.

I don't want to send attachments so I would

Re: Help with Microsoft help on linking Access to Outlook

enter just 'SendMessages' in
the
debug window, but why? – what happens
when I press enter?

Pressing Enter in the debug window means that what's been
typed preceding
the Enter gets executed.

Is this meant to test the procedure before
building it into a command
button's code?

Yes.

Once I get the code to work, I then need to
know how to apply it to my
own
particular circumstances. I will be providing
my users with a form
bound
to
table tblSchools. This table lists School
details, including Email
address.
They will use the form to filter down to a set
of records, e.g all
schools
in
a certain Town and I want this code (say, on
a command button on the
form)
to
transfer only those email addresses in the
filtered set to Outlook,
rather
than all records in tblSchools. How could
this be done?

First, let me tell you about a fairly serious omission in the
sample if
you're using Access 2000 or 2002. The code is using DAO as
the method of
getting the data from the table, but by default Access 2000
and 2002 do
not

Re: Help with Microsoft help on linking Access to Outlook

contain a reference to the DAO library. In section 9, where it's talking about setting references, you need to ensure that you also have a reference set to the Microsoft DAO 3.6 Object Library. As well, you need to change the line of code

```
Dim MyRS As Recordset
```

to

```
Dim MyRS As DAO.Recordset
```

(You could also change Dim MyDB As Database to Dim MyDB As DAO.Database if you want, but that's not absolutely necessary)

To customize the code to your specific situation, you need to change

```
Set MyRS = MyDB.OpenRecordset("tblMailingList")
```

to

```
Set MyRS = MyDB.OpenRecordset("tblSchool")
```

If the e-mail address in tblSchool isn't named EmailAddress, you also need to change EmailAddress to the actual field name in this line of code:

```
TheAddress = MyRS![EmailAddress]
```

Now, this code is going to send email to every entry in the table. To limit to whom the e-mail gets sent, you'll need to create a query that returns only those schools to whom you wish to e-mail, use that query name rather than tblSchool in the OpenRecordset command. Alternatively, you could create an SQL Statement in the routine, and use that to open the recordset:

Re: Help with Microsoft help on linking Access to Outlook

Dim strSQL As String

```
strSQL = "SELECT EmailAddress FROM tblSchool  
WHERE Town = 'xxxx'"  
Set MyRS = MyDB.OpenRecordset(strSQL)
```

--

Doug Steele, Microsoft Access MVP

<http://I.Am/DougSteele>

(no private e-mails, please)