

# RE: MS Query "Connections"

---

*Source:*

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2007-05/msg04016.html>

---

- *From:* Johnslg <[Johnslg@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:Johnslg@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Fri, 25 May 2007 06:36:02 -0700
- 

Thanks. I'll start playing around with this.....

I suppose after I get it all debugged I could use stored procedures. I may be hitting different databases for different companies but I could probably handle that with my connection string.

"Duke Carey" wrote:

Yes – you can pass a query string through ADO. HOWEVER, your query really ought to be converted to a stored procedure, allowing you to use code very much like what I posted, so that you simply pass in 2 parameters. The reason for using the stored procedure route is that it is faster.

If you truly want to pass in the query string, change

```
objCmd.CommandText = "dbo.vbMSA_Rankings2"  
objCmd.CommandType = adCmdStoredProc  
to  
objCmd.CommandText = your query string  
objCmd.CommandType = adCmdText
```

and drop all the stuff relating to parameters

The key is that you can open & close the ADODB connection at will, and if you are cycling through a number of account numbers on a sheet, open the connection once and close it when you are done AND in your error handler.

"Johnslg" wrote:

Can you pass a sql string with ADO? I'm using parameters & was trying to keep things simple.

```
Sub SQL_Query_Test()
```

RE: MS Query "Connections"

Dim Period As String, FirstAccountNumber As String,  
SecondAccountNumber  
As String, Result As String, MasterAccount As String, DataB As String  
Dim MyPos As Integer, MyLength As Integer

Worksheets("Sheet1").Range("B4").Activate 'sets active cell to  
first cell with account data

With Worksheets("Sheet1")  
Period = .Range("B2") 'cell B2 contains  
the period user enters  
DataB = .Range("B1")  
MasterAccount = ActiveCell 'Account Number of  
first account to be queried  
End With

For Counter = 1 To 100 '....pick a number,  
I could probably count the number of populated cells and get an exact #...

With Worksheets("Sheet1")  
Result = ActiveCell.Offset(0, 1).Address 'Place return data  
one cell to the right of current cell  
End With  
Worksheets("Sheet1").Range(Result).Clear 'Clear result cell  
If MasterAccount = "" Then GoTo NEXTFOR 'Skip empty cells  
If MasterAccount = "\*\*\*" Then Exit For 'End of file token

With Range(Result).Cells 'Format return data  
.NumberFormat = "\$###,###,###.00"  
.Font.Bold = True  
.Font.Size = 10  
.Font.Color = RGB(75, 0, 255)  
.Font.TintAndShade = -0.5  
End With

If InStr(MasterAccount, "~") = 0 Then 'check to see if two  
account#'s concantenated  
FirstAccountNumber = MasterAccount 'single account, set  
account number parameter for query  
With ActiveSheet.QueryTables.Add(Connection:= \_  
"ODBC;DRIVER=SQL  
Server;SERVER=LLCSERVER2;UID=john.church;Trusted\_Connection=Yes;APP=2007  
Microsoft Office system;" & \_  
"WSID=JC;DATABASE= & DataB & ",  
Destination:=Range(Result), Sql:= \_  
"Select Sum(AMTRANDTL.AMTRDTranAmt) from AMTRANDTL "  
& \_  
"left outer join GLACCTHDR on GLACCTHDR.GLACHNumber  
= AMTRANDTL.GLACHNumber " & \_  
"left outer join AMTRANHDR on AMTRANHDR.AMTRHNumber

RE: MS Query "Connections"

```
= AMTRANDTL.AMTRHNumber " & _  
"where GLACHCode= " & FirstAccountNumber & " and  
AMTRANHDR.AMTRHPeriod = " & Period & " Group By  
GLACCTHDR.GLACHCode")  
.FieldNames = False  
.RefreshStyle = xlOverwriteCells  
.PreserveColumnInfo = True  
.AdjustColumnWidth = False  
.Refresh BackgroundQuery:=False  
End With
```

"Duke Carey" wrote:

OK, you're not using ADODB. I'm not too familiar with what you are doing.

ADODB gives you lots of flexibility for creating your queries and returning results. And, you can close the connection after each query.

```
Option Explicit  
Sub RunStoredProcedures()  
Dim rng As Range  
Dim intRows As Integer  
Dim intParams As Integer  
  
Dim objConn As New ADODB.Connection  
Dim objCmd As New ADODB.Command  
Dim objRs As New ADODB.Recordset  
  
' On Error GoTo ErrHandler:  
  
Application.ScreenUpdating = False  
  
Range("A2").Resize(500, 10).ClearContents  
  
' Connect to the data source.  
Set objConn = GetNewConnection  
objCmd.ActiveConnection = objConn  
  
' Set CommandText equal to the stored procedure name.  
objCmd.CommandText = "dbo.vbMSA_Rankings2"  
objCmd.CommandType = adCmdStoredProc  
  
' Automatically fill in parameter info from stored procedure.  
objCmd.Parameters.Refresh  
  
intParams = objCmd.Parameters.Count - 1 'first one is  
RETURN value
```

RE: MS Query "Connections"

```
' call the stored procedure
objCmd(1) = Range("ticker")
objCmd(2) = 1 ' signifies we want PENDING

objCmd(0).Direction = adParamReturnValue
Set objRs = objCmd.Execute

If objRs.EOF = True Then
MsgBox "No Data", vbCritical
GoTo ErrHandler
End If

Set rng = Range("TopLeft")
rng.CopyFromRecordset objRs
Set rng = Range("Bank")
intRows = 1 + Range("B5000").End(xlUp).Row - rng.Row

ActiveSheet.Names.Add Name:="Print_Area", _
RefersTo:="" & rng.Resize(intRows, 10).Address

Range("ticker").Offset(2, 0).Value =
objRs.NextRecordset.Fields("Name").Value

Application.ScreenUpdating = True

rng.Select

ErrorHandler:
'clean up

If objConn.State = adStateOpen Then
objConn.Close
End If

Set objConn = Nothing
Set objCmd = Nothing

'Application.Calculation = xlCalculationAutomatic

If Err <> 0 Then
MsgBox Err.Source & "-->" & Err.Description, , "Error"
Stop
Resume 'Next
End If

End Sub
```

RE: MS Query "Connections"

```
Private Function GetNewConnection() As  
ADODB.Connection  
Dim oCn As New ADODB.Connection  
Dim sCnStr As String  
  
sCnStr = "Provider=sqloledb;User ID=duke; pwd=buckeyes;  
& _  
"Data Source=sandbox; Initial Catalog=BranchData"  
  
oCn.Open sCnStr  
  
If oCn.State = adStateOpen Then  
Set GetNewConnection = oCn  
End If  
  
End Function
```

"Johnslg" wrote:

hmmm....I'm calling QueryTable inside a  
For loop in a single VB macro. The  
loop will run for as many cells of data the  
user adds to a column in the  
spread sheet.

I'm doing a Activesheet.QueryTable.Add.....

I'm not sure how to "refresh".

Can I do ...QueryTable.Refresh.... instead of  
Add?

Is there no "drop connection" command or  
method?

THANKS!

"Tom Ogilvy" wrote:

<http://nickhodge.co.uk/gui/datamenu/dataexamples/externaldataexamples.htm>

<http://www.dicks-clicks.com/excel/ExternalData6.htm#Parameters>

once the query is built,  
change the parameter and  
refresh the query (don't

RE: MS Query "Connections"

rebuild it). Make the the  
Backgroundquery property  
is set to false.

--

Regards,  
Tom Ogilvy

"Johnslg" wrote:

I have a  
sheet that  
lets the user  
enter a chart  
of accounts  
in a column  
&  
then a  
macro runs  
against each  
account  
(using  
QueryTable),  
hits an SQL  
database,  
and returns  
data. My  
problem is  
that it adds  
a new  
connection  
each  
time the  
micro runs. I  
just deleted  
about 400  
connections.

Is there a  
way to  
reuse the  
same  
connection  
(it's all  
going  
against the  
same  
server/database)  
or is there a

RE: MS Query "Connections"

way to  
automatically  
delete  
connections?

Thanks.