

# Using MS-Query

1	Introduction.....	2
2	Installation .....	2
3	The Database.....	2
4	Setting Up Data Sources .....	2
4.1.1	Selecting the Database Source Type.....	3
4.2	Selecting the Database.....	4
4.2.1	Selecting the Table.....	4
4.2.2	Selecting the Columns .....	5
4.2.3	Filtering the Columns .....	5
4.2.4	Sort the Data .....	6
4.2.5	Setting up to Read a Parameter at Run Time .....	7
4.2.6	Import the Data .....	8
4.2.7	Redefining Runtime Parameters .....	9
4.2.8	Finishing Up .....	10
5	Setting up a New Data Source.....	11
6	Importing Text Files .....	13
7	Query Behaviors.....	15
7.1	Push Aside.....	15
7.2	Inserting Columns .....	16
7.3	Changing Source Location (Excel Spreadsheets).....	16
7.4	Changing Source Code Location (MS-Access).....	17
8	Bugs.....	20
8.1	This Data Source Contains No Visible Tables .....	20
8.2	No Header When Linking to a Text File .....	22

# Using MS-Query

## 1 Introduction

Do you find yourself frequently needing information contained in a database, another spreadsheet, or a text file in your spreadsheet? Do you have to apply formulas to this data? Do you need to filter it and sort it at run time? MS-Query can do all of this for you.

This tutorial explains how to set up a data source, generate a query, and set up parameters to run your reports.

## 2 Installation

MS-Query is part of Excel, but is normally not activated until used. The first time you attempt to use it, you will be prompted to "install" it. Depending on how Office was installed, you may or may not need to use the installation CDs. Follow the prompts to complete the installation. Once installed, MS-Query will always be available.

## 3 The Database

In this document you will learn how to set up queries to attach to MS-Access Databases, Excel Spreadsheets and text files. You will also learn how to import from a text file.

The data source we will be working with contains the following information.

*The Database*

Name	Gender	Occupation
Bailey, Beetle	Male	Perpetual Private
Bumstead, Blondie	Female	Caterer
Clampett, Jed	Male	Poor Mountaineer
Drew, Nacy	Female	Amateur Detective
Flintstone, Fred	Male	Dino Operator
Kent, Clarke	Male	Mild Mannered Reporter
Possible, Kim	Female	Student
Woman, Wonder	Female	Crime Fighter
Ziffle, Arnold	Male	Barnyard Animal

These data sources, for the purposes of this tutorial are in C:\Temp and are named:

- Access Database: People.mdb (tbl\_people)
- Excel Spreadsheet: People.XLS
- Text File: People.CSV

These files are contained in the zip file. Download them to C:\Temp if you want to follow along with the exercises.

## 4 Setting Up Data Sources

MS-Query has the ability to read data from various sources. Please note: the instructions presented here are for Office 2003. Microsoft likes to change the user

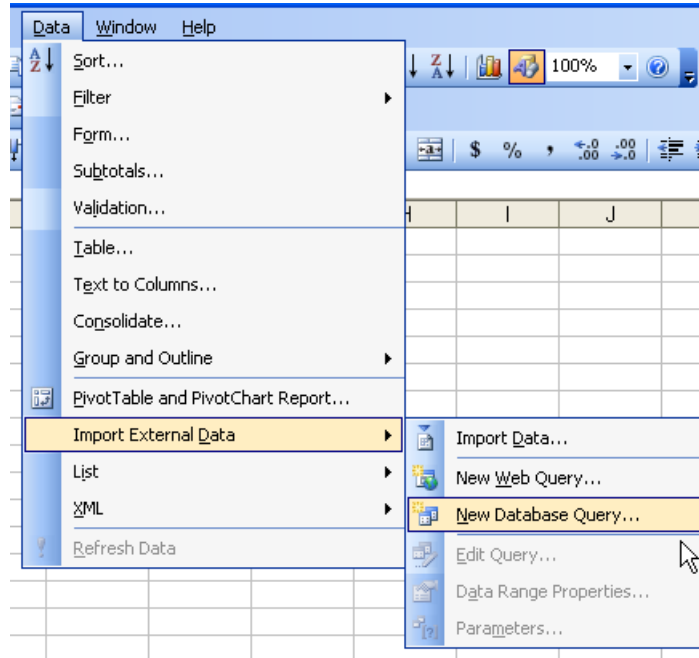
## USING MS-QUERY

interface with each release, if you are using another version of office, your displays may look different.

For all applications, the process starts with invoking the “External Data” menu.

Select Data → Import (or Get) External Data → New Database Query. This action will bring up a menu similar to the following:

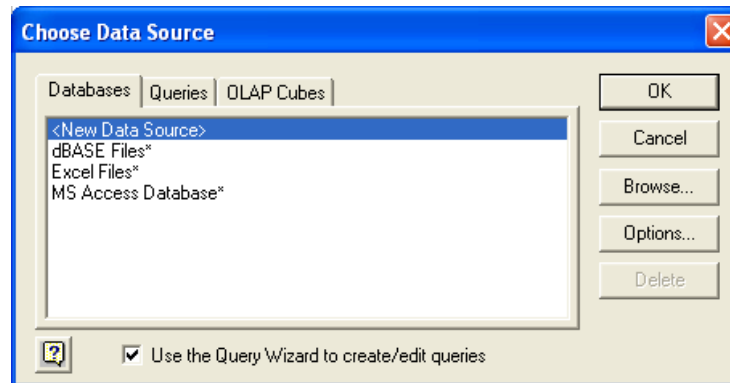
*External Data Menu*



### 4.1.1 Selecting the Database Source Type

Selecting “New Database Query” brings up the following dialog box.

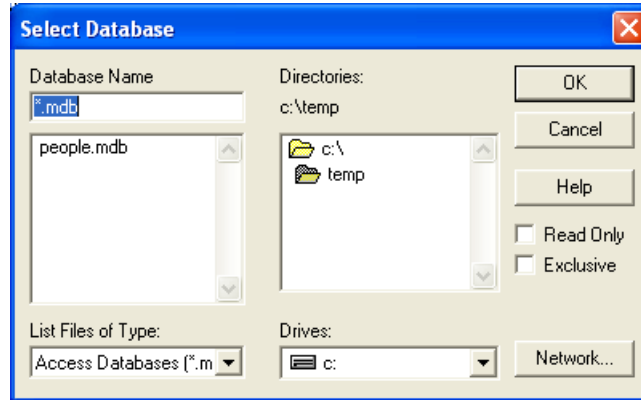
*Data Source(Type) Dialog Box*



## 4.2 Selecting the Database

Setting up the basic query is the same for any of the data sources with minor variations. For example, selecting the MS Access option brings up the following dialog box.

*Database Selection Dialog Box*

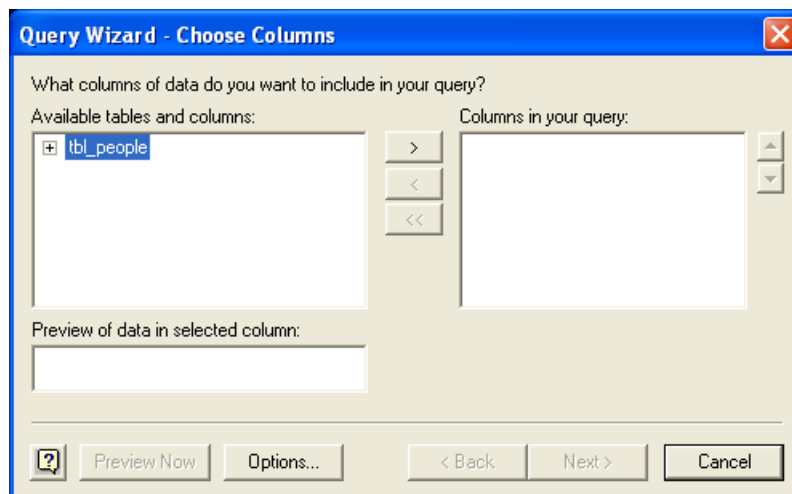


Microsoft is very fond of mapped drives. If you are planning to share the spreadsheet and query with others who may not map their drives the same way as you, specify the full path name rather than using the dialog box to navigate to the data source. For example \\Server Name\Data Sources\Data Source Name.

- In the case of an Access database, you will be given a list of available tables and queries. **Note:** Do not select queries that prompt for a parameter. Doing so will cause an error message later on in the process. See section 4.2.5 for setting up parameters.
- In the case of an Excel spreadsheet, you will be given a list of page names and named ranges if the named ranges are static. Named ranges that are dynamic will not be displayed.

### 4.2.1 Selecting the Table

*Table Selector Dialog Box*

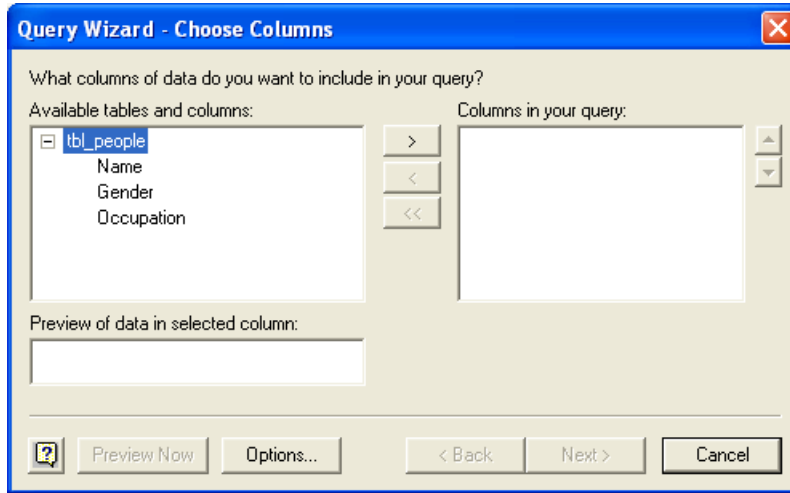


From this point on, all the query methods continue in the same fashion.

### 4.2.2 Selecting the Columns

Once you select the table you can select which columns you want to display.

*Column Selection*

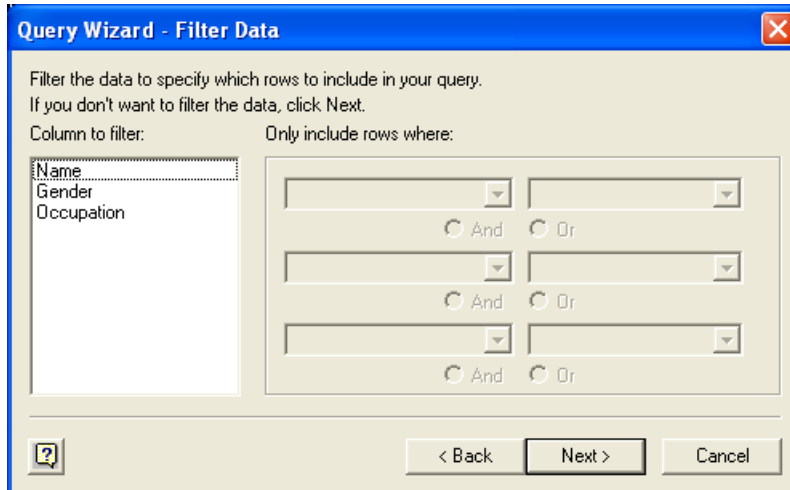


You can select individual columns or if you click the ">" while the table is selected, all columns will be moved over.

### 4.2.3 Filtering the Columns

You can choose which records are passed on by the query.

*Filter Dialog Box*



**Note:** You can only set up "fixed" parameters with this part of the wizard. You cannot set up a prompt or formula. To set up parameters that can be set at run time, see section 4.2.5.

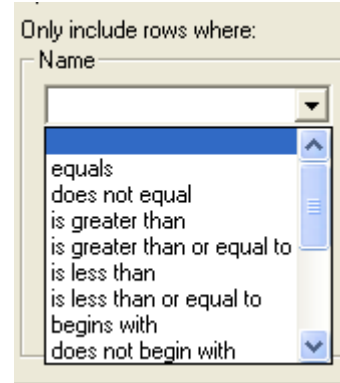
## USING MS-QUERY

Highlight the column on which you want to filter. You can select multiple and/or criteria for this parameter.

You can also select additional columns for filtering.

Selecting an item will activate the dropdown list of the operators.

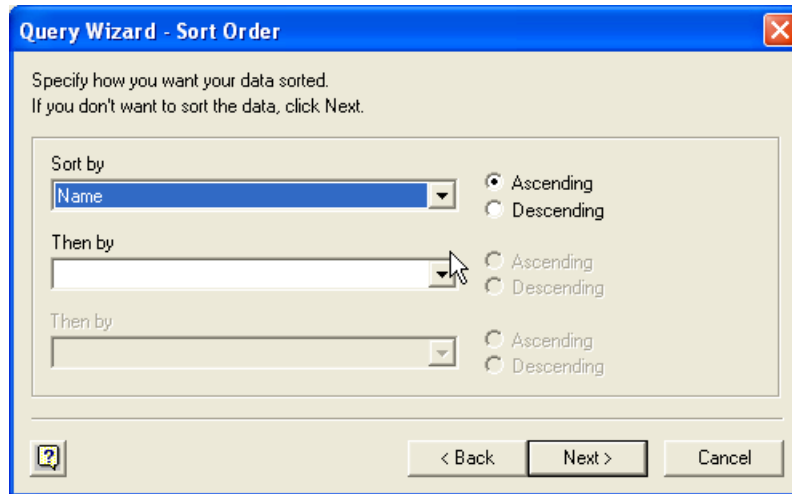
Some of the options are



### 4.2.4 Sort the Data

You can also perform three levels of sort using the next dialog box.

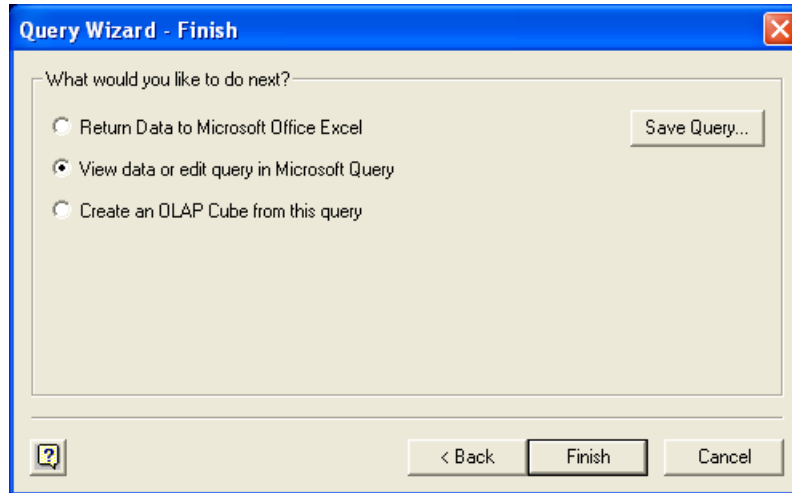
*Sort Order Dialog Box*



### 4.2.5 Setting up to Read a Parameter at Run Time

You can skip this section if you do not need to read a parameter at run time. You can always edit the query and get back to this screen later.

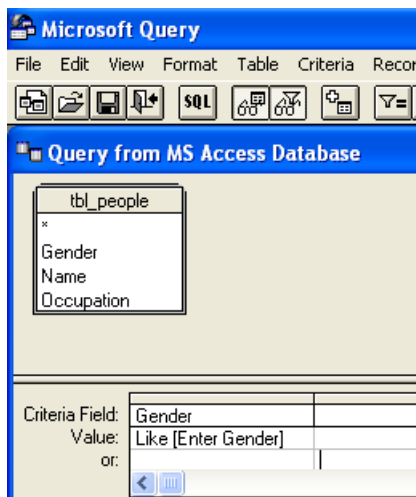
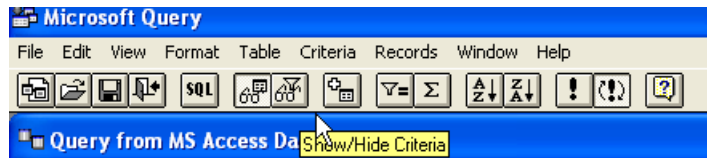
*Finish Dialog Box*



- To exit back to Excel, select "Return Data to Microsoft Office Excel."
- To set a parameter for run time, select "View Data or Edit Query in Microsoft Query."

Selecting the view data option will bring up a dialog box showing the results of the query. To add a criteria, click on the "Show/Hide Criteria" icon (eye glasses and funnel).

*Microsoft Query Tool Bar*



This action will open up the criteria section of the dialog box.

You can drag the column name into the criteria field, or use the drop-down list to select the column

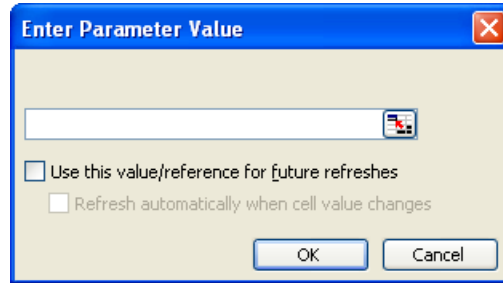
Type in the value for the criteria. In this step, make the parameter interactive by using square brackets []. This choice can be changed later.

In this case, we are setting up the parameter to look at a string (that can accept wildcards) to match to the gender.

## USING MS-QUERY

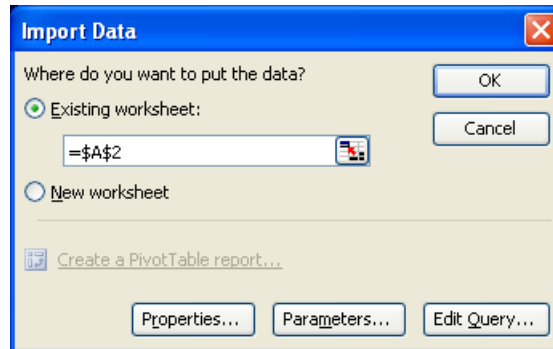
Once you set up the parameter, you will be prompted for it.

### *Interactive Parameter Prompt*



### 4.2.6 Import the Data

Once you have the query set up the way you want it, you will be prompted where to put it.



The cell definition is where the top, left of the returned data will be placed. You can put the query anywhere on the page. See the discussion on query behavior in Section 4.2.5 for additional information.

Normally the query is written to cell \$A1 or cell \$A2. If the query has nice headers and you want to keep them, return the data to cell \$A1. If query has oddly-named headers and you want to suppress them, return the data to cell \$A2.

### 4.2.7 Redefining Runtime Parameters

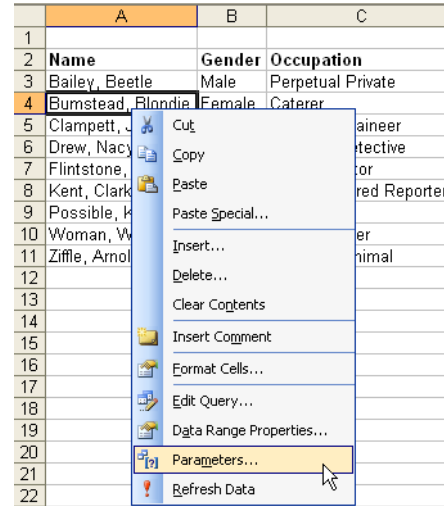
If you set up a query to read parameters at runtime, you can set them to be read by right clicking on the returned data set.

This action brings up a menu as shown in the illustration at right.

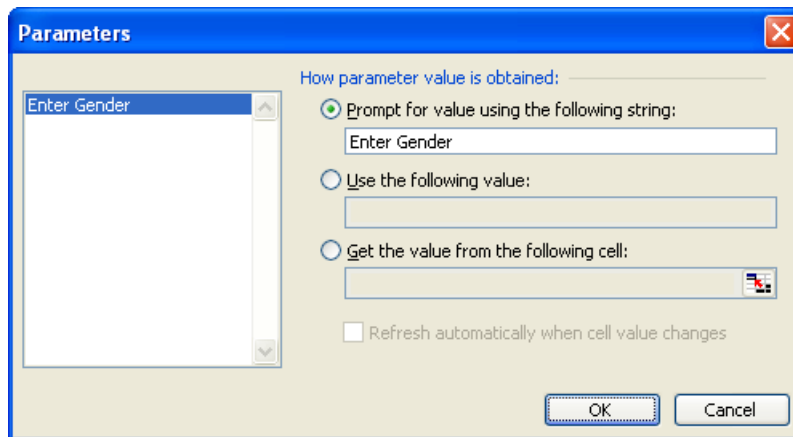
Select "Parameters" from this menu.

This action will bring up a dialog box as shown below.

This option will not be available on the menu unless an interactive query is set up as in Section 4.2.5.



Parameters Dialog Box



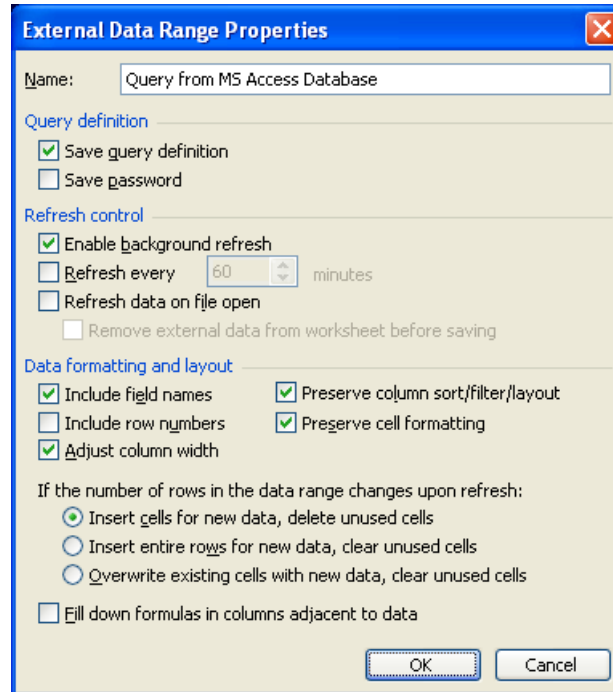
Use this box to:

- Prompt the user for a value
- Use a value. The value can be a constant like 'Female' or a formula like =today().
- Get the value from a cell.

#### 4.2.8 Finishing Up

Now that you have the data returned, it is time to define some of the behavior. Right click on the data set and get the menu shown in section 4.2.7. This time select “Data Range Properties.” This action will give you the dialog box shown below.

*Data Range Properties Dialog Box*



Select the options that make the most sense for you. Shown here are the defaults.

To suppress headers, uncheck “Include Field Names.” You might also wish to uncheck “Adjust Column Width” with this options.

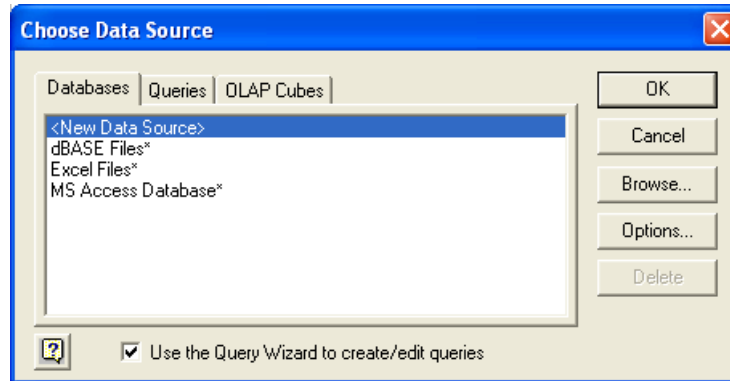
A word or two about “Fill Down Formulas in Columns Adjacent to Data:” This option works on all columns immediately adjacent to the right of the data set. The formulas on the top row are replicated for each row of data returned. MS-Query also “cleans up” after itself and “removes” excess formulas if subsequent refreshes return fewer rows.

A break in the continuity of the formulas (that is, a blank column) stops the replication, so you can set up summary calculations in this area.

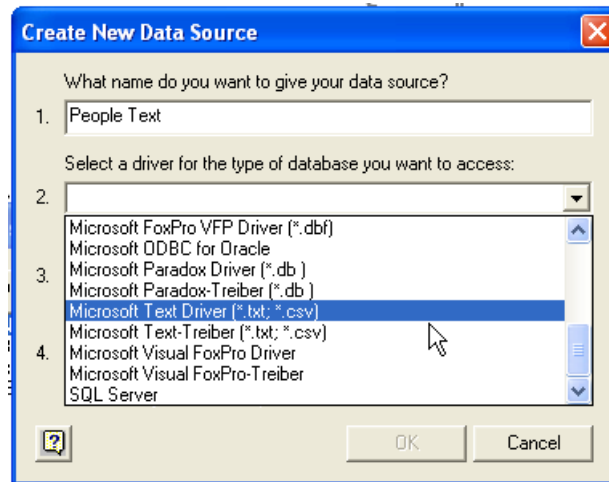
## 5 Setting up a New Data Source

There are two ways of getting text information into Excel. The first method is to define the directory in which text files reside as a data source. Then you can use any text file within the directory as a data source

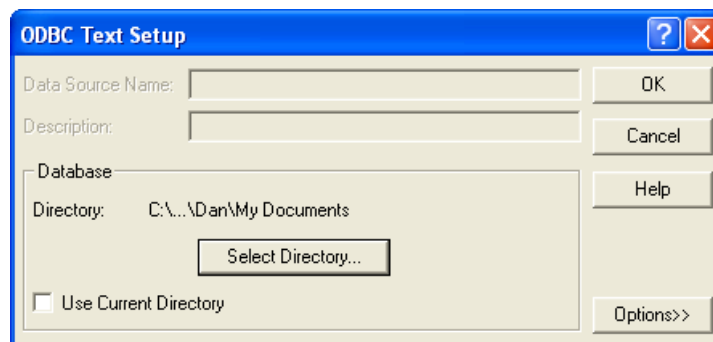
To start the process select Data → Import (or Get) External Data → New Database Query → New Data Source.



Select the Microsoft Text Driver

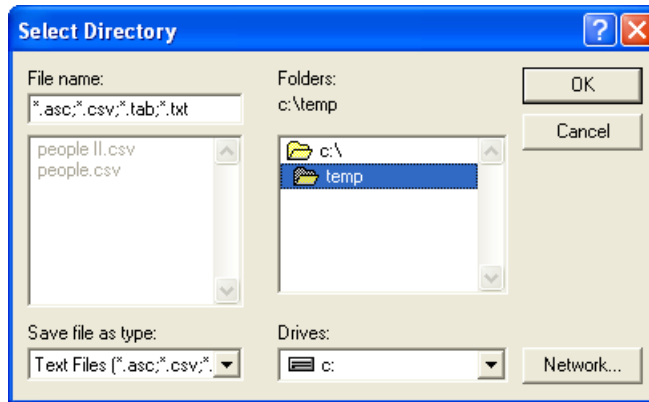


In the resulting dialog box, uncheck the "Use Current Directory" box.

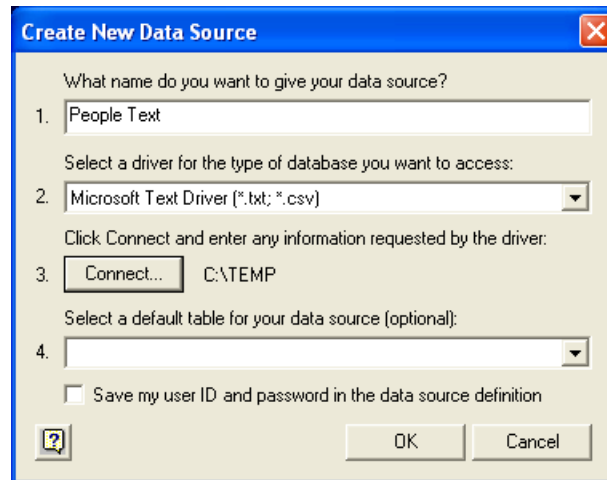


## USING MS-QUERY

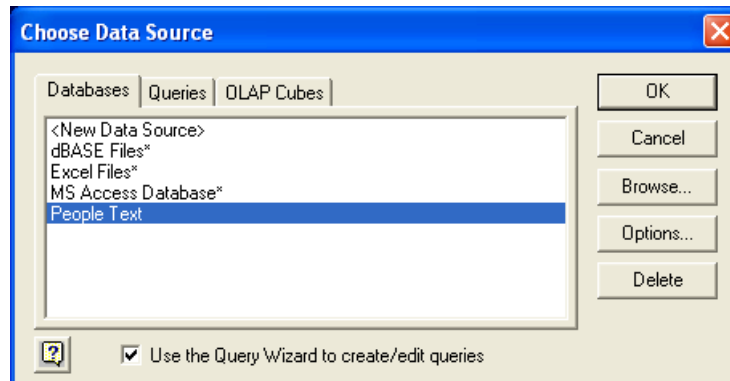
Select the folder containing your data sources.



When you are done, the original dialog box should look something like:



Click OK and you are brought back to a familiar screen.

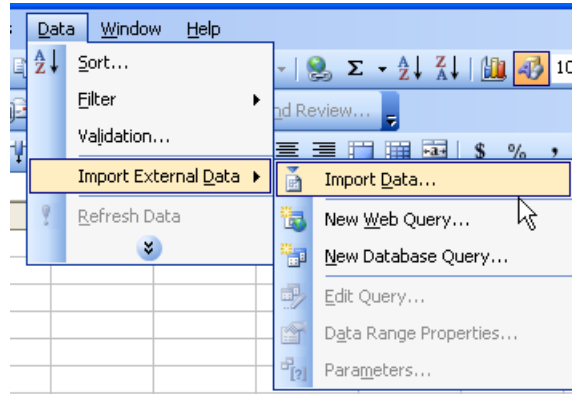


You can now use the newly-created data source in the exact same way as other data sources. The only difference is that the only tables you will see are the text files in the directory.

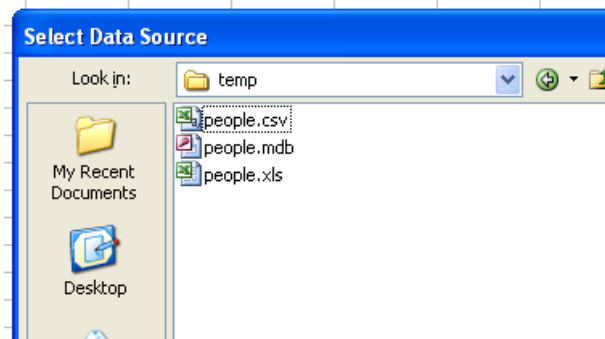
## 6 Importing Text Files

The other way of getting text data into Excel is to import it.

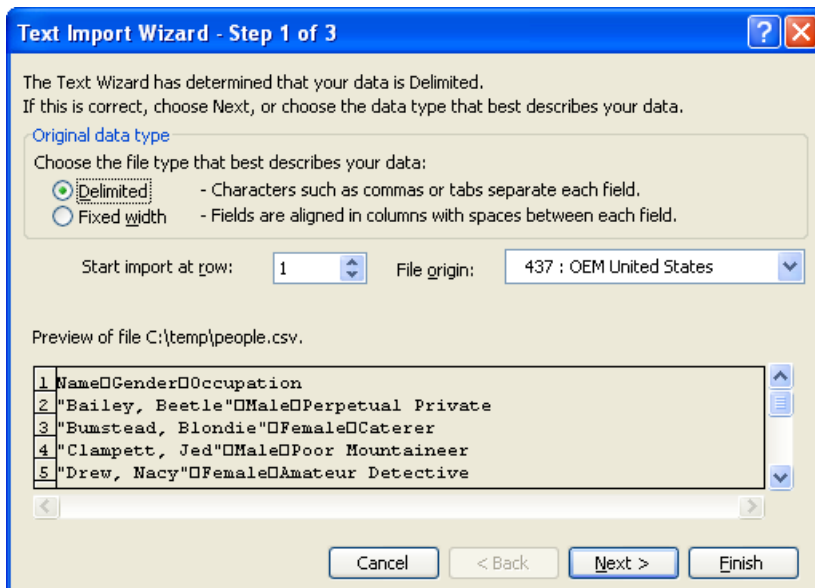
Select Data → Import External Data → Import Data



This option will also allow you to import Excel spreadsheets and Access databases, however, only the text import will be discussed here.



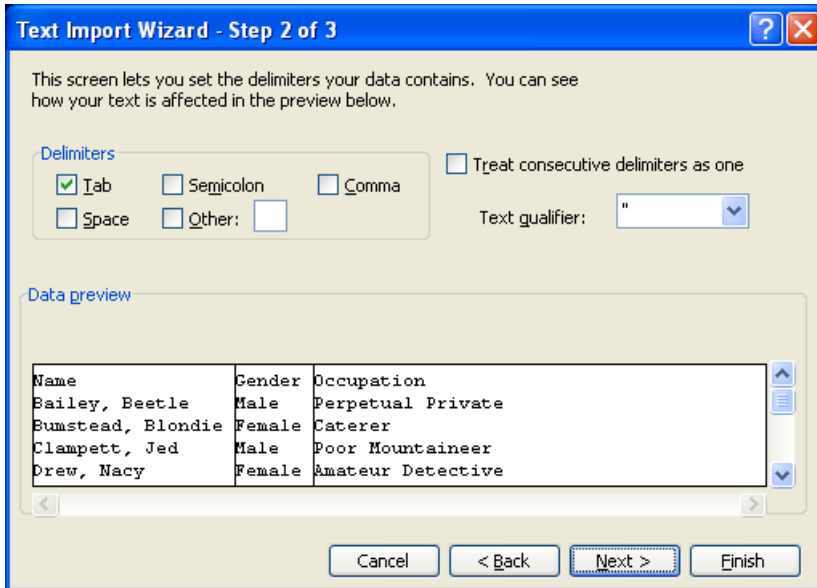
Selecting a text file brings up the following dialog box.



## USING MS-QUERY

The advantage of importing text is that you can also work with fixed formats as well as delimited formats. In addition you can start importing at other than row 1. This comes in handy when dealing with reports that are made more human readable by having additional header information.

The next dialog box lets you set up the delimiter to use.



This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

**Delimiters**

Tab     Semicolon     Comma  
 Space     Other:

Treat consecutive delimiters as one

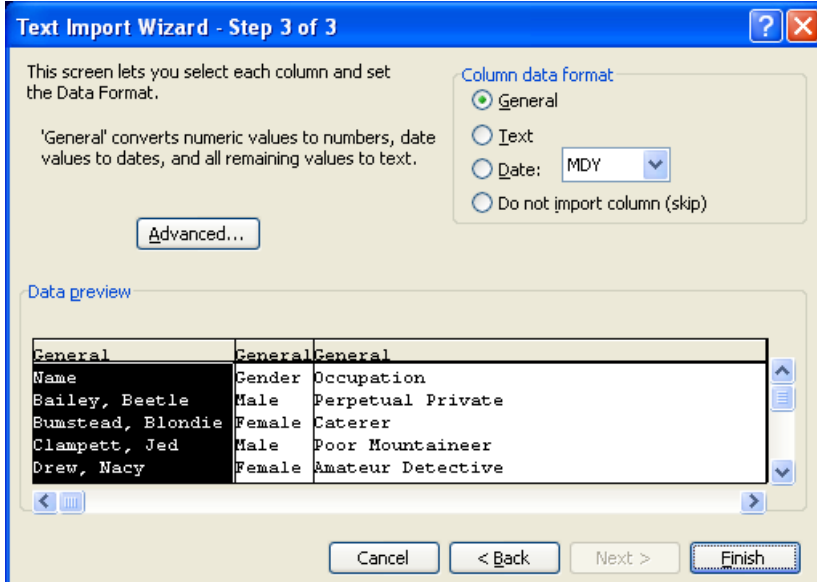
Text qualifier:

**Data preview**

Name	Gender	Occupation
Bailey, Beetle	Male	Perpetual Private
Bumstead, Blondie	Female	Caterer
Clampett, Jed	Male	Poor Mountaineer
Drew, Nancy	Female	Amateur Detective

Buttons: Cancel, < Back, Next >, Finish

The final dialog box allows you to set the data type for each of the columns.



This screen lets you select each column and set the Data Format.

'General' converts numeric values to numbers, date values to dates, and all remaining values to text.

**Column data format**

General  
 Text  
 Date:   
 Do not import column (skip)

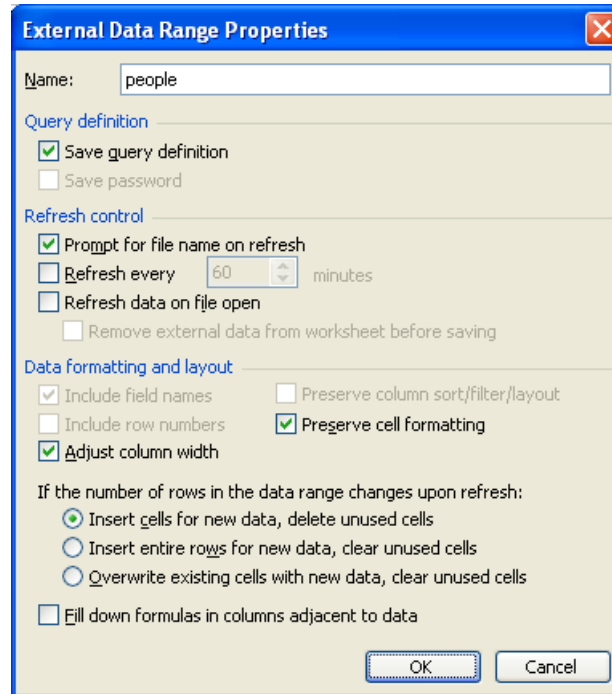
**Data preview**

General	General	General
Name	Gender	Occupation
Bailey, Beetle	Male	Perpetual Private
Bumstead, Blondie	Female	Caterer
Clampett, Jed	Male	Poor Mountaineer
Drew, Nancy	Female	Amateur Detective

Buttons: Cancel, < Back, Next >, Finish

## USING MS-QUERY

Once you've defined the import, the returned data will look like other queries. However, right clicking the data set reveals some differences.



- The first difference is that you do not have a way to filter or sort imported data
- Also, you can't suppress field names
- On the other hand, you can prompt for file name on refresh. This is a very handy feature to have if the name of the file to import changes every day such as a basic file name that might have a daily date stamp as part of its name.

## 7 Query Behaviors

Like any other piece of software, MS-Query is not without its features and bugs.

### 7.1 Push Aside

One of these features is MS-Query's ability to expand and contract to take up only as much space as it needs to in a spreadsheet.

The good thing about this is that you can "stack" queries on top of each other. For example, suppose query one initially returns 10 rows of data and query two is initially set up to be returned to row 12 (two rows below). If on a subsequent refresh, query one returns 14 rows of data, query two will now start on row 16. Likewise, if query one returns 8 rows, query two will move up to row 10.

One of the problems with this feature is it also tends to push columns aside as well, sometimes destroying the references in their formulas. If you are going to use queries with other calculations and data on a page, it is a good idea to lay out the queries first and then build the sheet around them.

## 7.2 Inserting Columns

If you decide to add a column between two existing columns and you refresh the new column will be added – as the last column. To solve this problem insert a column where you want the new column to be, cut the newly-returned column and paste it in the proper position. Then close up (delete) any vacant columns and refresh the query. It will come back OK the next time.

## 7.3 Changing Source Location (Excel Spreadsheets)

As mentioned above, Microsoft tends to latch onto drive mappings rather than absolute or relative paths. If you created queries against a mapped drive, you can edit the query and tweak the SQL code.

It is also possible to run a query from one sheet on a spreadsheet to another. One reason for doing this is to read raw data onto some sheets, perform sum summary calculations on them, and then read the resulting data to the main page.

The problem with this is that if you move the spreadsheet or rename it, it looks for the data in the old location. Putting the following code in the ThisWorkbook part of the VB editor will change the query to look in the current spreadsheet (whatever its name) in the current location (wherever that may be).

Code:

---

```

Sub ChangeSource()

Dim j As Integer

' Variables for the query strings
Dim SQLStmt() As String
Dim FromStmt() As String
Dim NewPath As String
Dim NewSQL As String
Dim NewFrom As String

' Variables for the connection strings
Dim ConnectString() As String
Dim NewConnectString As String

' For each query do ...
For j = 1 To Worksheets(1).QueryTables.Count
' Change the SQL Statement

    NewPath = "" & ThisWorkbook.Path & "\" & _
        Left(ThisWorkbook.Name, Len(ThisWorkbook.Name) - 4) & ""
    ' Split the SQL statement at "FROM"
    SQLStmt = Split(Worksheets(1).QueryTables(j).CommandText,
"FROM")

    ' Split the remaining statements at the "." The frist part of
this is
    ' the old path
    Let FromStmt = Split(SQLStmt(1), ".")

    ' Conatenate the string as far as the "FROM" and add new path
    NewSQL = SQLStmt(0) & "FROM "
```

## USING MS-QUERY

```
' Create the new from statement
FromStmt(0) = NewPath
Let NewFrom = Join(FromStmt, ".")
NewSQL = NewSQL & " " & NewFrom

Worksheets(1).QueryTables(j).CommandText = NewSQL

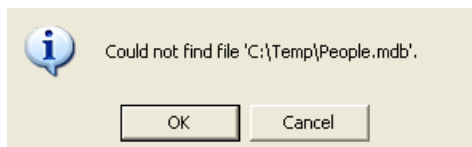
' Change the connection string
Let ConnectString =
Split(Worksheets(1).QueryTables(j).Connection, ";")
ConnectString(2) = "DBQ=" & ThisWorkbook.Path & "\" & _
    ThisWorkbook.Name
ConnectString(3) = "DefaultDir=" & ThisWorkbook.Path
Let NewConnectString = Join(ConnectString, ";")
Worksheets(1).QueryTables(j).Connection = NewConnectString
Next j
End Sub
```

### 7.4 Changing Source Code Location (MS-Access)

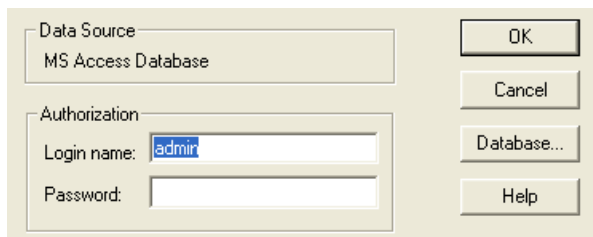
If your query is linked to an MS-Access Database and the database is moved, there is a simple and sane procedure to re-link the data to the new location.

Note: The assumption is made that the database is moved to a new location. If a copy of the database is left behind, MS-Query will detect it and think nothing is wrong. To force it to initiate this process, temporarily rename the old database.

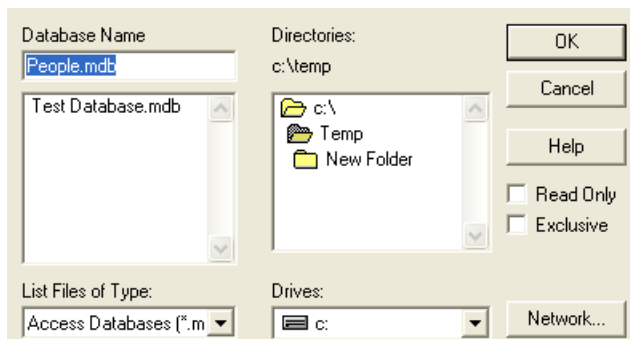
If you have moved a database, you may see an error message that looks like:



Click OK and the following screen will appear:

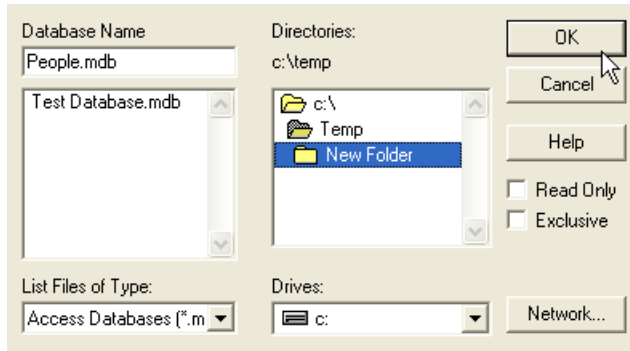


Click on Database.



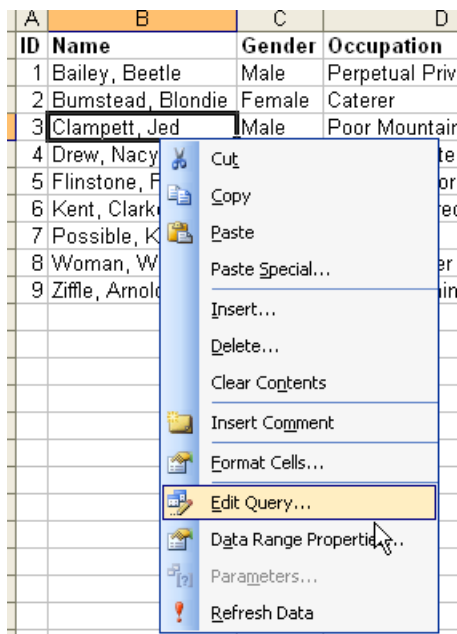
## USING MS-QUERY

Navigate to the new database location (don't forget to use a full pathname if needed).

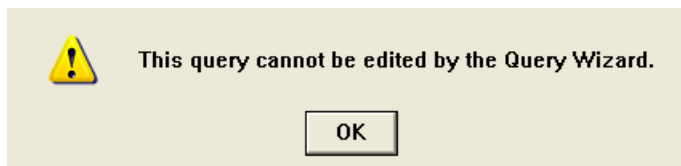


Normally, this is all it takes. However, if the error message still persists, take the following steps.

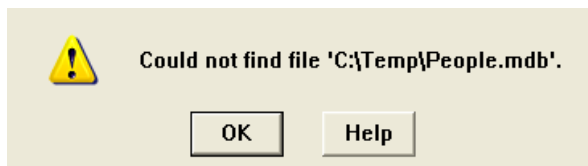
Right click on the data range, and select Edit Query.



This action should bring up the following dialog box.

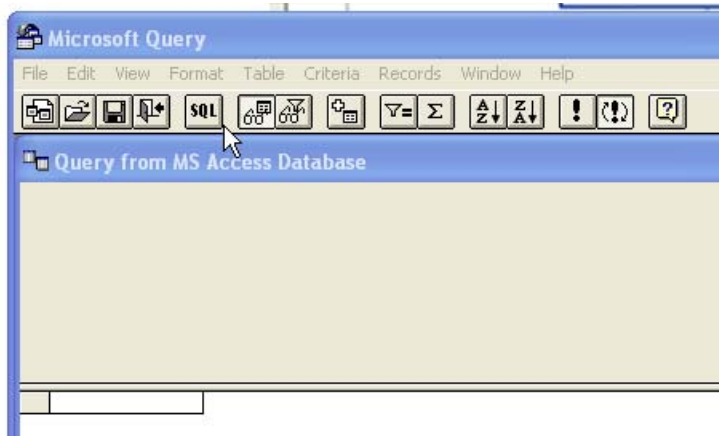


Click OK and the error message will be repeated.

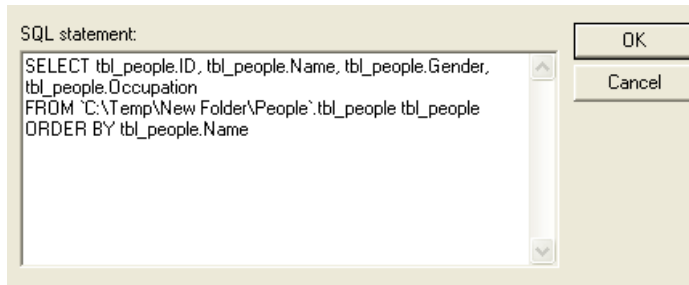


## USING MS-QUERY

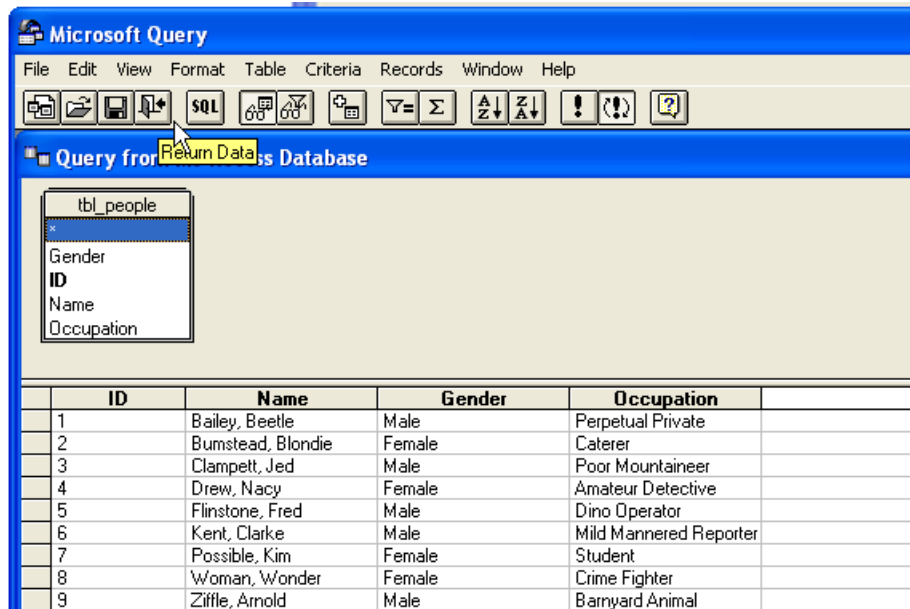
Click OK and you get no results. Select SQL



Edit the SQL statement to reflect the new path name.



Exit back to the spreadsheet.



## 8 Bugs

This function is not without its unwanted features:

### 8.1 This Data Source Contains No Visible Tables

You may see this message while trying to link to another spreadsheet as a data source. It can be fixed by launching MS-Word's Mailmerge function. (How intuitive is that?) Microsoft keeps moving the fix for this around their web site. Here is the copy as of 4/11/09.

<http://support.microsoft.com/kb/286891>

Article ID: 286891 - Last Review: January 29, 2007 - Revision: 2.2

#### **This data source contains no visible tables**

This article was previously published under Q286891

#### **SYMPTOMS**

When you work with a document and you try to open a Microsoft Excel file with Microsoft Query, you may receive the following error message:

This data source contains no visible tables.

#### **CAUSE**

This issue occurs because the ODBC driver used to connect to the data source perceives the worksheets in your Excel workbook as system tables and does not display them. System tables are objects in database files that are not displayed by default.

#### **WORKAROUND**

To work around this issue, follow these steps:

1. Start Microsoft Word, and then open your document.
2. In Word Microsoft 2002, point to **Letters & Mailings** on the **Tools** menu, , and then click **Mail Merge Wizard**.

In Microsoft Office Word 2003, point to **Letters & Mailings** on the **Tools** menu, and then click **Mail Merge**.

In Microsoft Office Word 2007, click **Start Mail Merge** in the **Start Mail Merge** group on the **Mailings** tab. Then, click **Step by Step Mail Merge Wizard**.

3. Click **Letters**, and then click **Next**.

## USING MS-QUERY

4. Click **Use Current Document** and click **Next**.
5. Click **Use an Existing List**, and then click **Browse**.
6. In the **Select Data Source** dialog box, on the **Tools** menu, click **MSQuery**.  
Note that Microsoft Query starts and the **Choose Data Source** dialog box is displayed.
7. Click to select **Excel Files**, and then click **OK**.
8. In the **Select Workbook** dialog box, browse to your Excel file and click **OK**.

**Note** that you receive the error message listed in the Symptoms section of this article.

9. Click **OK**. In the **Query Wizard - Choose Columns** dialog box, and then click **Options**.
10. In the **Table Options** dialog box, click to select the **System Tables** check box, and then click **OK**.

Note that now you can see and select individual worksheets in your Excel data source.

### **STATUS**

Microsoft has confirmed that this is a problem in the Microsoft products that are listed in the "Applies to" section.

### **APPLIES TO**

- Microsoft Office Word 2007
- Microsoft Office Word 2003
- Microsoft Word 2002 Standard Edition

### **Keywords:**

kbug kbnofix KB286891

## 8.2 No Header When Linking to a Text File

Sometimes when linking to a text file it will appear that the headers in the text files are run together. This happens when the text driver thinks that the file is comma delimited when in fact it is tab delimited. To correct this problem go to the directory in which the data sources reside and edit the schema.ini file using notepad. Modify the Format line to reflect the appropriate type delimiter.

```
[newhires.csv]
ColNameHeader=True
Format=CSVDelimited
MaxScanRows=25
CharacterSet=OEM
[today.txt]
ColNameHeader=True
Format=TabDelimited
MaxScanRows=25
CharacterSet=OEM
```