Moving Data From Excel To SAS

And

Back Again

By
Ben Cochran
The Bedford Group
bedfordgroup@nc.rr.com

Contents

.

2. Using DDE to Import Excel Data

Dynamic Data Exchange (DDE) is a way of dynamically exchanging information between Windows applications.

You can use DDE with:

- the DATA step
- the SAS Macro facility
- a SAS/AF application
- any operation in the SAS System that requests and generates data

To use DDE in SAS, issue a FILENAME statement with this general form:

FILENAME *fileref* DDE **'***DDE-triplet* **'** | **'**CLIPBOARD**'** <DDE - options>**;**

where:

fileref	is any valid fileref
DDE	is a required keyword
'*DDE- triplet* **'**	**'**CLIPBOARD**'** is the name of the DDE external file
DDE - options	include HOTLINK, NOTAB, COMMAND

triplet ⇨

For more information on DDE options, refer to SAS Companion for the Microsoft Windows Environment from SAS Institute.

Using DDE to Import Excel Data

The DDE triplet is application dependent and takes the following form**:**

‘ application **-** name **|** topic **!** item ‘

where:

application - name	winword (for MS Word) or excel (for Excel spreadsheets)
topic	is the topic of conversation between SAS and the DDE application… typically the full path filename of the file.
item	is the range of conversation. For example, in a spreadsheet this is usually a range of cells.

To determine the exact value of the DDE triplet for a spreadsheet application, do the following**:**

- from the spreadsheet, select the range of cells, then copy them into the clipboard,
- go to the command line / bar in SAS and issue the DLGDDE command:

DLGDDE ⇨

example ⇨

Using DDE to Import Excel Data

Task: Read a range of cells from Basis.xls located in 'c:\unzipped'. The range is from row3, column1 to row19 column4. The spreadsheet is shown below.

Basis.xls [Read-Only]

	A	B	C	D	E
1					
2					
3	1/16/2002	10.75	10.75		1/16/2002
4	1/17/2002	10.75	10.75		1/17/2002
5	1/18/2002	10.75	10.75		1/18/2002
6	1/21/2002	10.75	10.75		1/21/2002
7	1/22/2002	10.75	10.75		1/22/2002
8	1/23/2002	10.75	10.75		1/23/2002
9	1/24/2002	10.75	10.75		1/24/2002
10	1/25/2002	10.75	10.75		1/25/2002
11	1/28/2002	10.75	10.75		1/28/2002
12	1/29/2002	10.25	10.25		1/29/2002
13	1/30/2002	10.75	10.75		1/30/2002
14	1/31/2002	10.75	10.75		1/31/2002
15	2/1/2002	10.75	10.75		2/1/2002
16	2/4/2002	10.75	10.75		2/4/2002
17	2/5/2002	9.4	9.4		2/5/2002
18	2/6/2002	9.4	9.4		2/6/2002
19	2/7/2002	9.4	9.4		2/7/2002

pgm ⇨

Using DDE to Import Excel Data

Look closely at the FILENAME statement.

```
filename  in  dde  'Excel | c: \ unzipped \ [basis.xls]Sheet1!R3C1:R19C4';

data spreadsheet_test;
   infile in dlm='09'x notab dsd missover;
   input start_date: mmddyy10. start_amt end_amt blank $;
run;

proc print data=spreadsheet_test;
  format start_date weekdate25.;
run;
```

where:

notab	tells SAS not to convert the tabs sent from Excel into blanks,
dlm	defines the delimiter as hexadecimal representation of the tab character.
dsd	tells SAS to treat 2 consecutive delimiters as a missing value
missover	tells SAS to set all variables in the current observation to missing if values are not found in the current record or row.

Note: with this approach, the spreadsheet needs to be **open.**

log ⇨
6

Using DDE to Import Excel Data

```
277
278  filename in dde 'Excel|c:\unzipped\[basis.xls]Sheet1!R3C1:R19C4';
279  data spreadsheet_test;
280    infile in dlm='09'x notab dsd missover;
281    input start_date: mmddyy10. start_amt end_amt blank $;
282  run;

NOTE: The infile IN is:
      DDE Session,
      SESSION=Excel|c:\unzipped\[basis.xls]Sheet1!R3C1:R19C4,
      RECFM=V,LRECL=256

NOTE: 17 records were read from the infile IN.
      The minimum record length was 17.
      The maximum record length was 22.
NOTE: The data set WORK.SPREADSHEET_TEST has 17 observations and 4 variables.
NOTE: DATA statement used:
      real time           3.94 seconds


283  proc print data=spreadsheet_test;
284    format start_date weekdate25.;
285  run;

NOTE: There were 17 observations read from the data set WORK.SPREADSHEET_TEST.
NOTE: PROCEDURE PRINT used:
      real time           1.32 seconds
```

output ⇨

Using DDE to Import Excel Data

Obs	start_date	start_ amt	end_amt	blank
1	Wednesday, Jan 16, 2002	10.75	10.75	
2	Thursday, Jan 17, 2002	10.75	10.75	
3	Friday, Jan 18, 2002	10.75	10.75	
4	Monday, Jan 21, 2002	10.75	10.75	
5	Tuesday, Jan 22, 2002	10.75	10.75	
6	Wednesday, Jan 23, 2002	10.75	10.75	
7	Thursday, Jan 24, 2002	10.75	10.75	
8	Friday, Jan 25, 2002	10.75	10.75	
9	Monday, Jan 28, 2002	10.75	10.75	
10	Tuesday, Jan 29, 2002	10.25	10.25	
11	Wednesday, Jan 30, 2002	10.75	10.75	
12	Thursday, Jan 31, 2002	10.75	10.75	
13	Friday, Feb 1, 2002	10.75	10.75	
14	Monday, Feb 4, 2002	10.75	10.75	
15	Tuesday, Feb 5, 2002	9.40	9.40	
16	Wednesday, Feb 6, 2002	9.40	9.40	
17	Thursday, Feb 7, 2002	9.40	9.40	

The example on the next page shows a DDE application that **Opens** a spreadsheet, reads it, then closes it ⇨…

Using DDE to Import Excel Data

```
options noxwait noxsync;

*--- Open Excel, put SAS to Sleep, and open a specific spreadsheet ---*;

x 'start excel';
data _null_;
     x=sleep(12);
run;

filename cmds DDE 'Excel|system';
data _null_;
   file cmds;
   put '[FILE-OPEN("C:\ben\UO_Rooms_Sold_06_05_05.xls")]' ;
run;

filename in dde 'Excel|C:\Ben\[Rooms_Sold.xls]PBH_Sold!R1C1:R30C40';

data PBH_Sold3;
     infile in dlm='09'x notab dsd missover;
     input (var1-var3) (:$20.) (var4-var40) (:$12.);
run;

*--- Close the Spreadsheet  ---*;

data _null_;
   file cmds;
   put "[FILE-CLOSE()]" ;
   put "[QUIT]";
run;
```

2. Using the Import / Export Wizard

Demo

3. Using PROC IMPORT / EXPORT

Demo

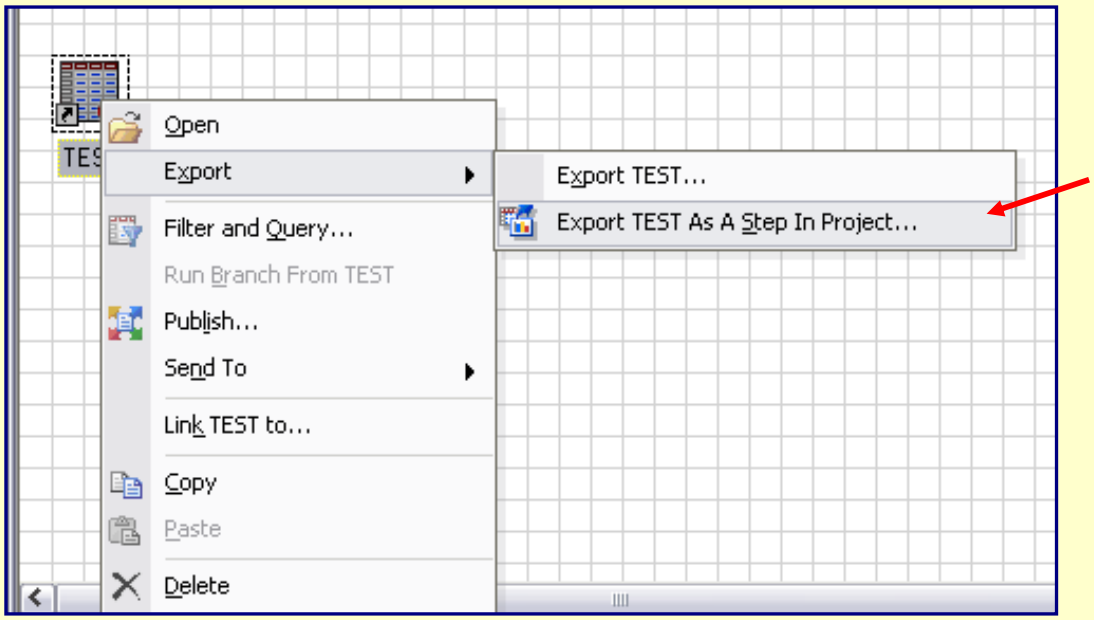
.

4. Using Enterprise Guide

Demo: Bringing Excel Data into EG

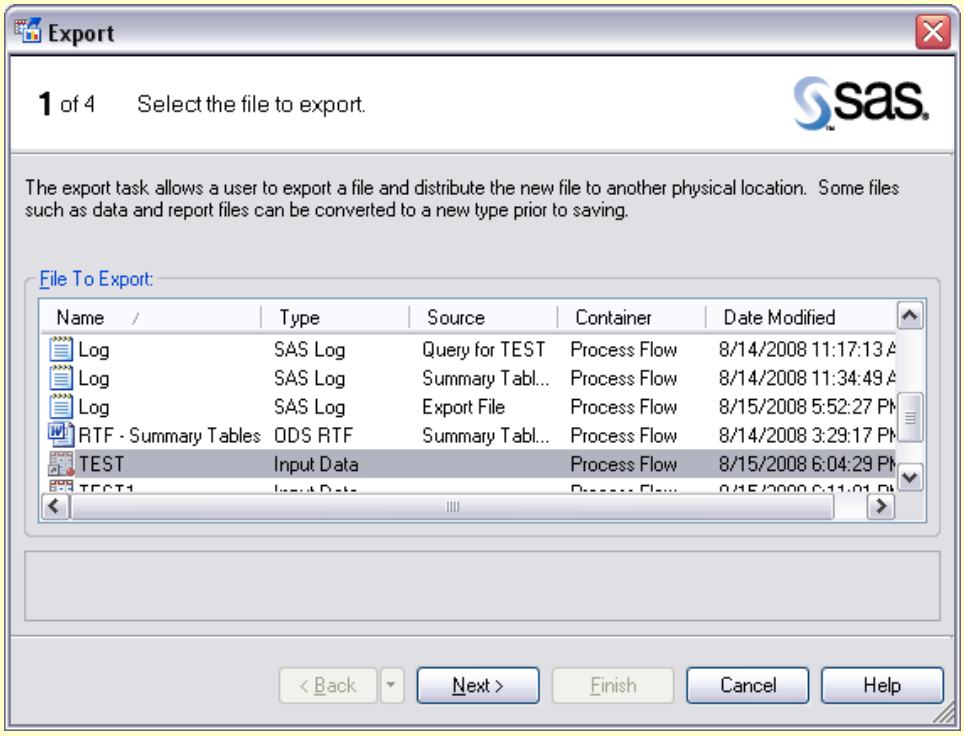
Exporting Datasets to Excel

From within the Enterprise Guide project, select the dataset you want to export to Excel. Click on it and select **Export → Export *dataset name* As A Step in Project ...**



Exporting Datasets to Excel

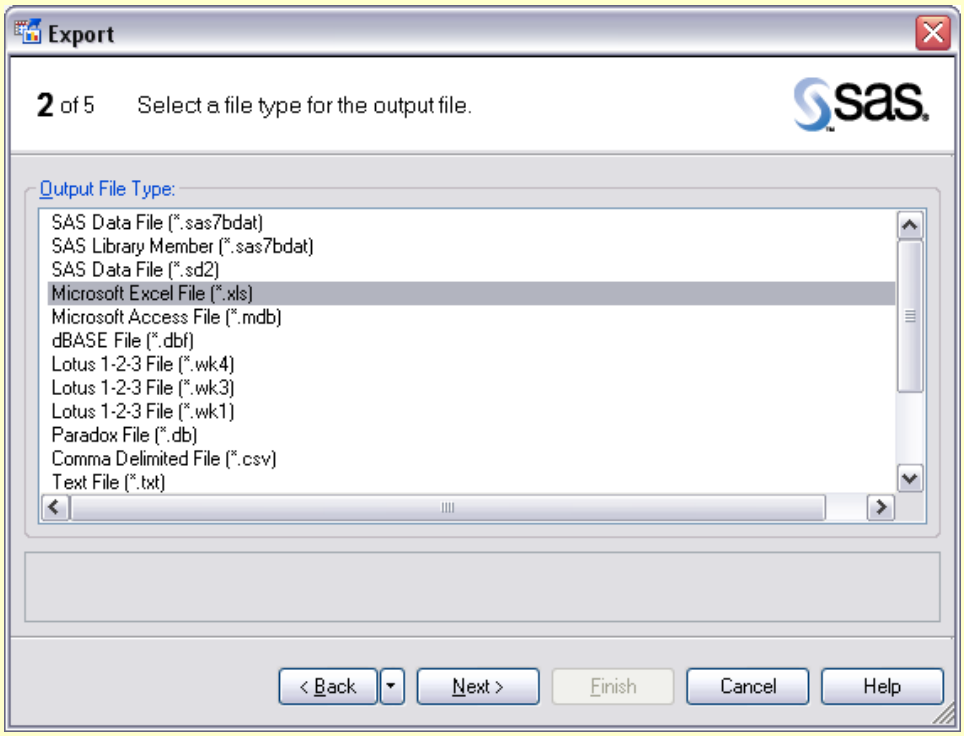
From the Export window, first select the dataset you want to export...



… then select **A Next>** .

Exporting Datasets to Excel

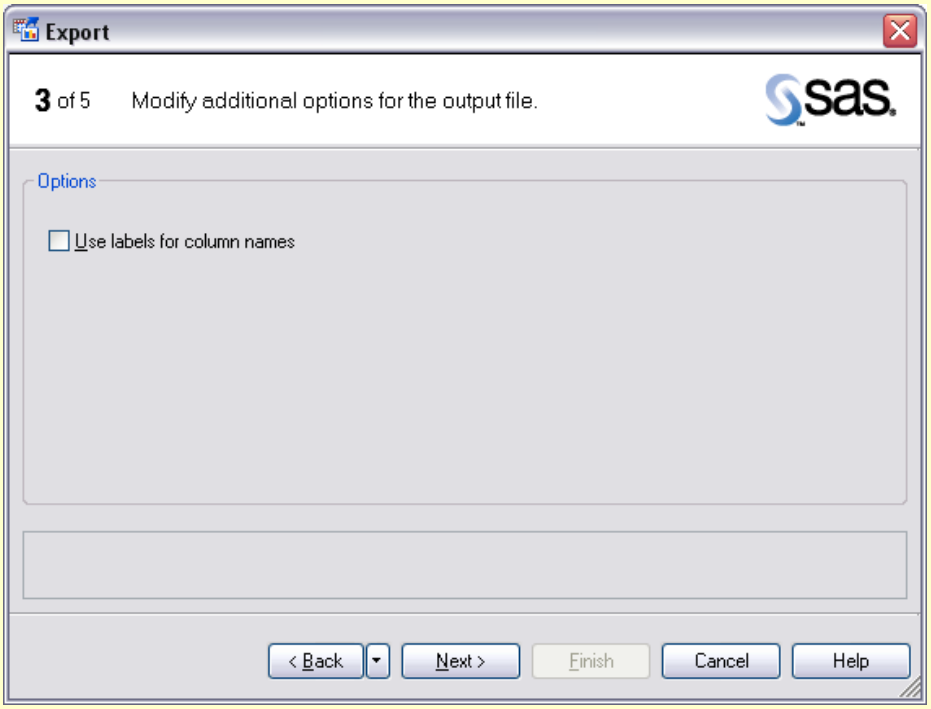
Next, select the type of file (Excel) for the output file …



… then select **Next>.**

Exporting Datasets to Excel

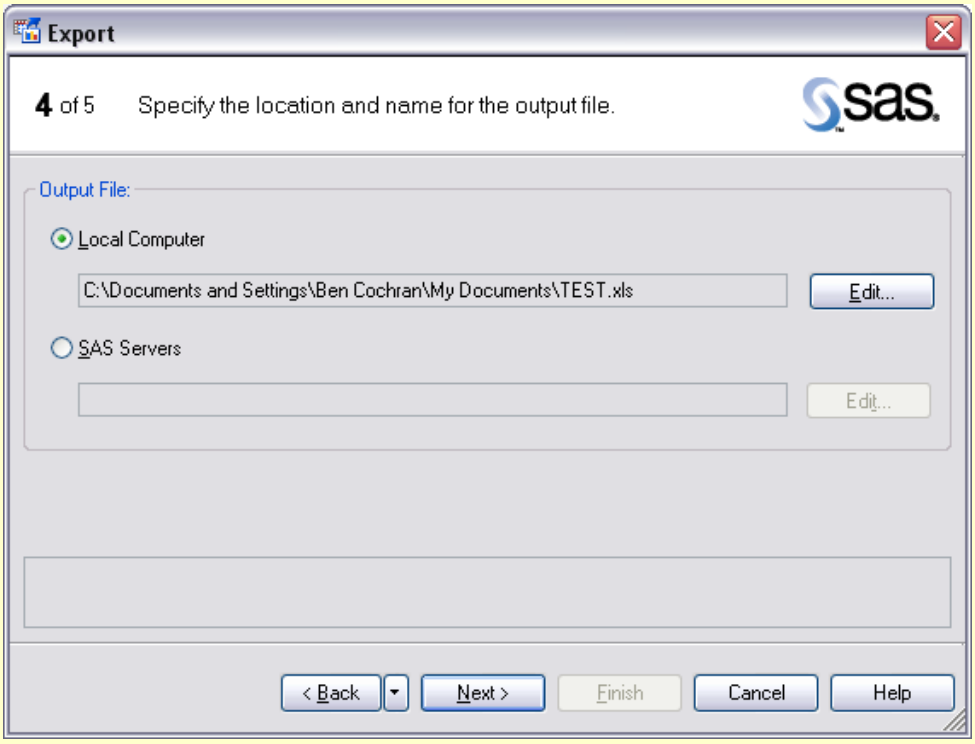
Choose whether you want to use the column names or column labels as headers in the new file …



… Select **Next> .**

Exporting Datasets to Excel

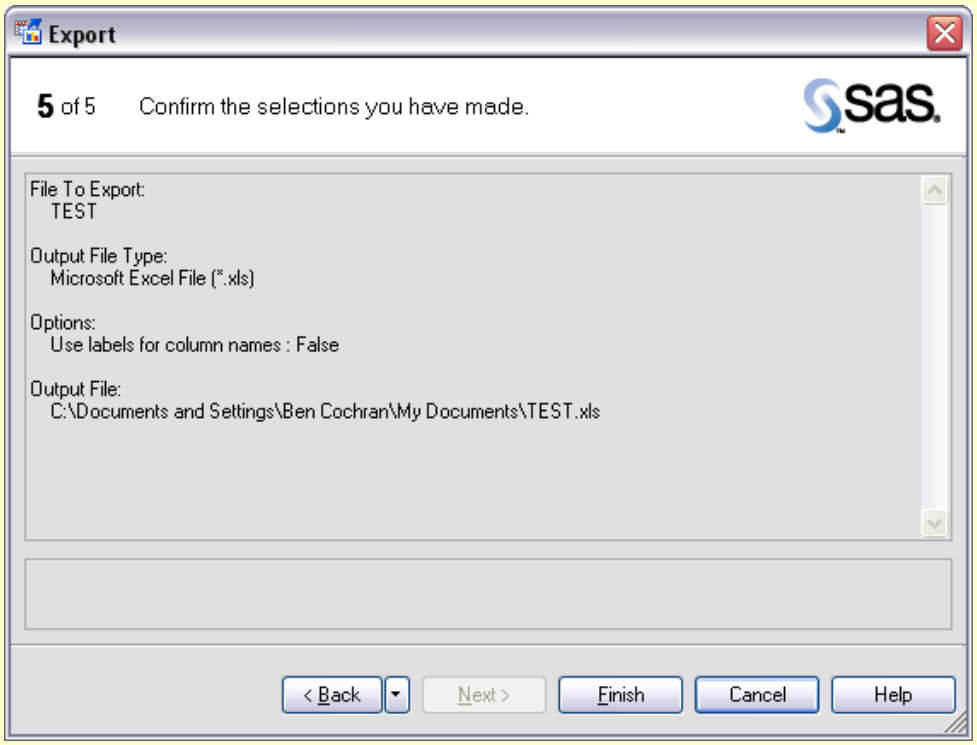
Select a location for the new spreadsheet.



… Select **Next> .**

Exporting Datasets to Excel

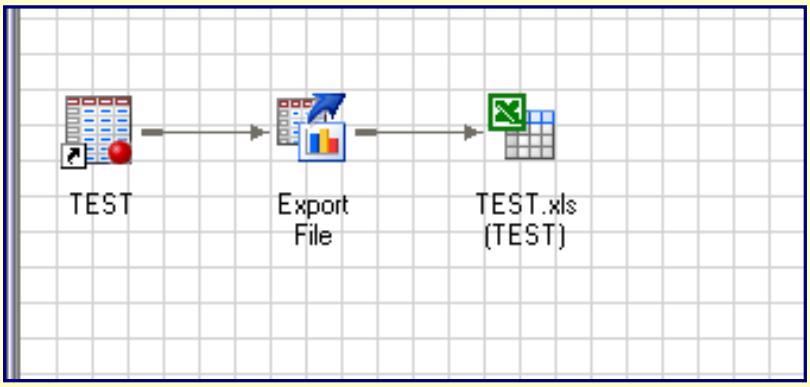
Verify the correctness of the information...



… then select **Finish.**

Exporting Datasets to Excel

When the Process Flow window reappears, the Export File task is in place.



…

5. Using The Add-in for Microsoft Office Product

Overview

The **objectives** for this section include**:**

- Illustrate the capabilities of the SAS Add-in for Microsoft
- Examine the SAS servers available from the SAS Add-in for Microsoft Office

Q. What does it do?

A. Extends the capabilities of Microsoft Office by enabling the user to harness the power of SAS data sources from within Microsoft Word and Excel.

Q. Why is it Important?

A. There are many business users who can benefit from the power of the SAS system, but they are not comfortable working in a traditional programming environment.

The SAS add-in brings SAS analytic capabilities to the Microsoft Office environment.

Objectives

Q. What are the Capabilities?

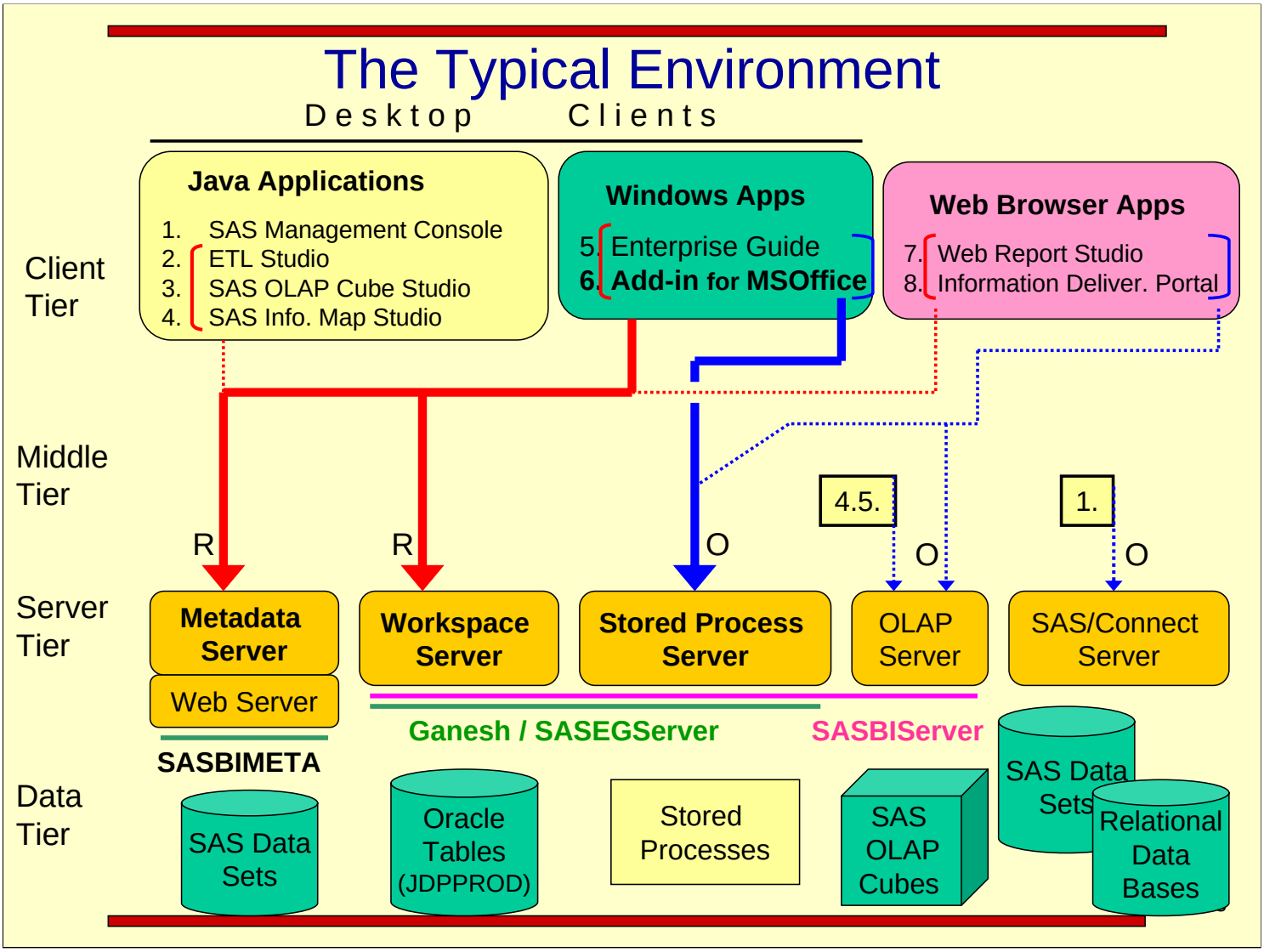
A. You can**:**

1. Create SAS reports in Microsoft Office applications using stored processes,
2. Access SAS data sources.
3. Access non-SAS data sources that are available from the SAS Server.
4. Analyze SAS or Excel data using analytic tasks.

Q. What are the requirements?

1. Windows NT4, Windows 2000, Windows XP,
2. Office 2000 or greater.
3. SAS Add-in for Microsoft Office.

The Typical Environment



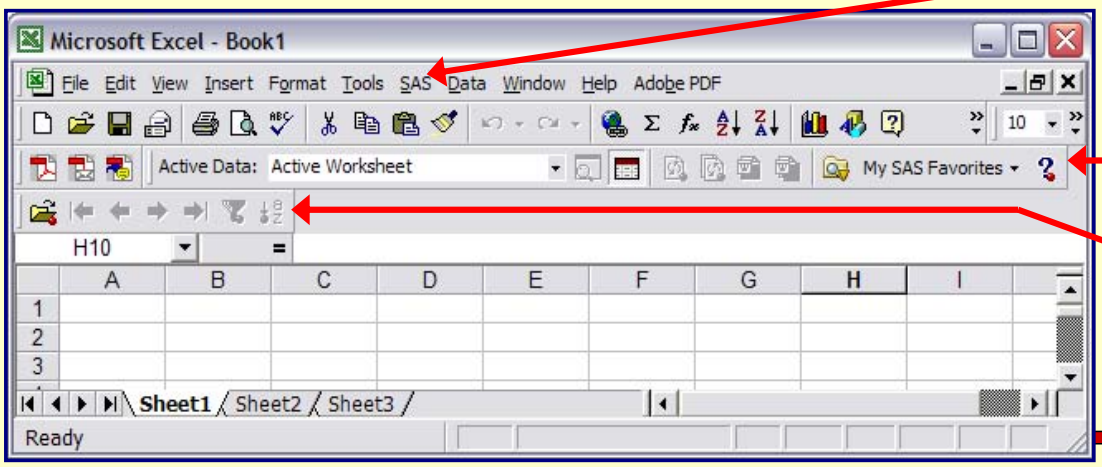
Using the SAS Add-in for Microsoft Office

The **objectives** for this section include**:**

- ❑ Examine the SAS Add-in to Microsoft Office user interface
- ❑ Explore the capabilities of the SAS Analysis Toolbar.
- ❑ Explore the capabilities of the SAS Data Analysis Toolbar.
- ❑ Examine the tools available to filter and sort data on the SAS server from the SAS Add-in to Microsoft Office.

The SAS Add-in for Microsoft Office **adds** a**:**

- ❑ **SAS menu** to the menu bar
- ❑ **SAS Analysis** toolbar
- ❑ **SAS Data Analysis** toolbar (Excel only)



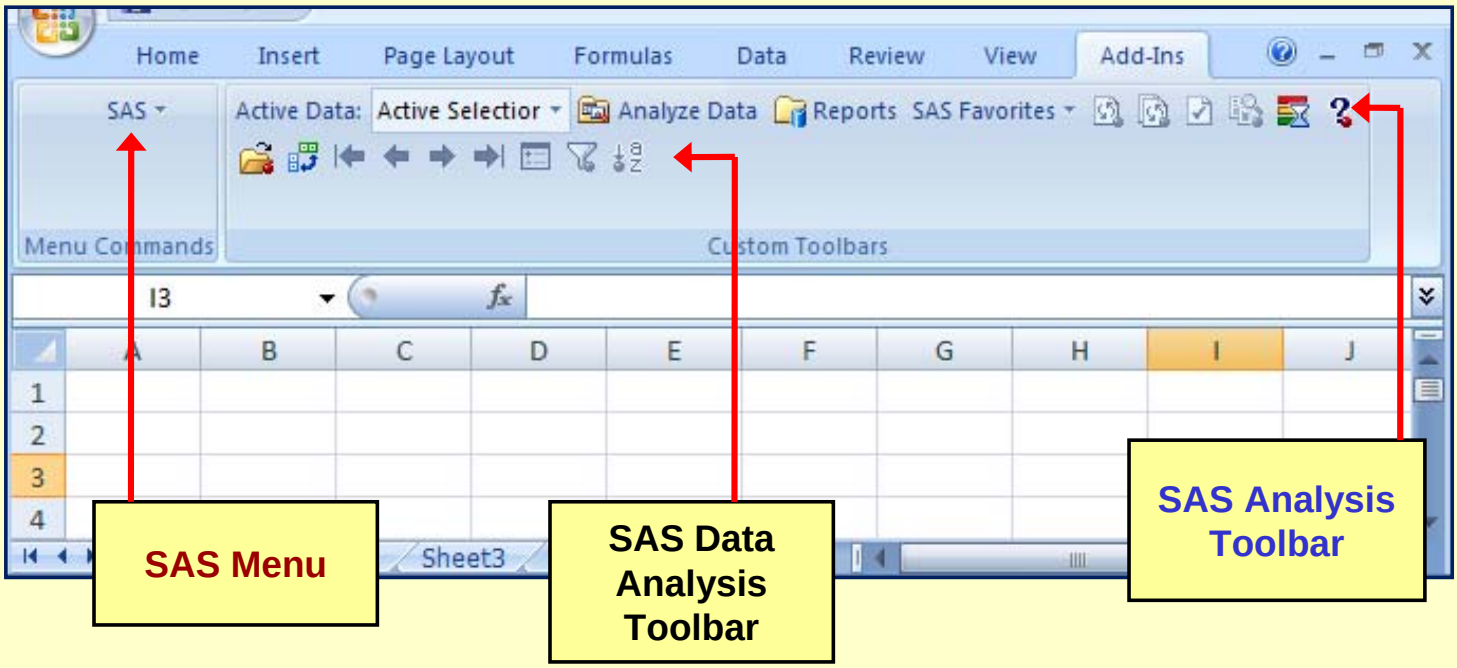
SAS Menu

SAS Analysis Toolbar

SAS Data Analysis Toolbar

Using the SAS Add-in for Microsoft Office

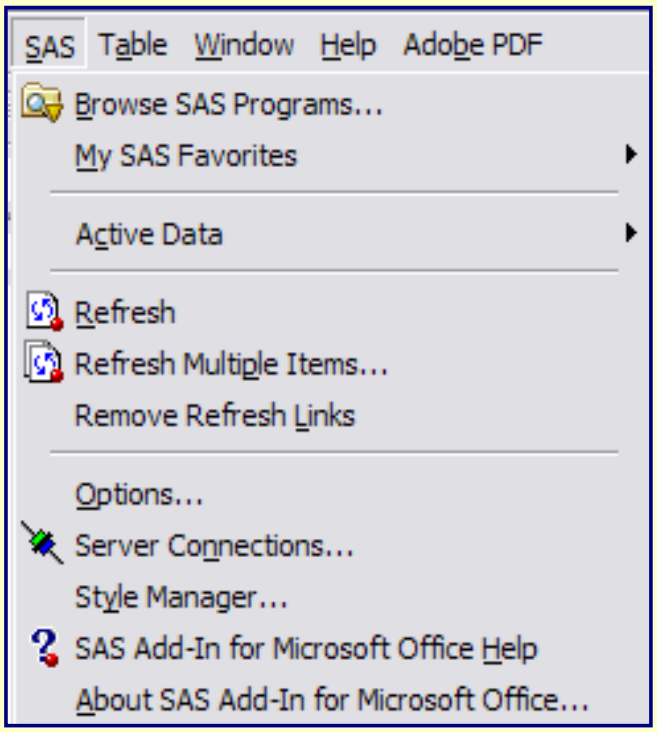
If you have **Office 2007** products installed on your computer, the **SAS Add-in** product would have the following appearance.



SAS Menu for Microsoft **Word**

The **SAS menu** in Microsoft Word provides access to the power of SAS while working in a Word document. You can**:**

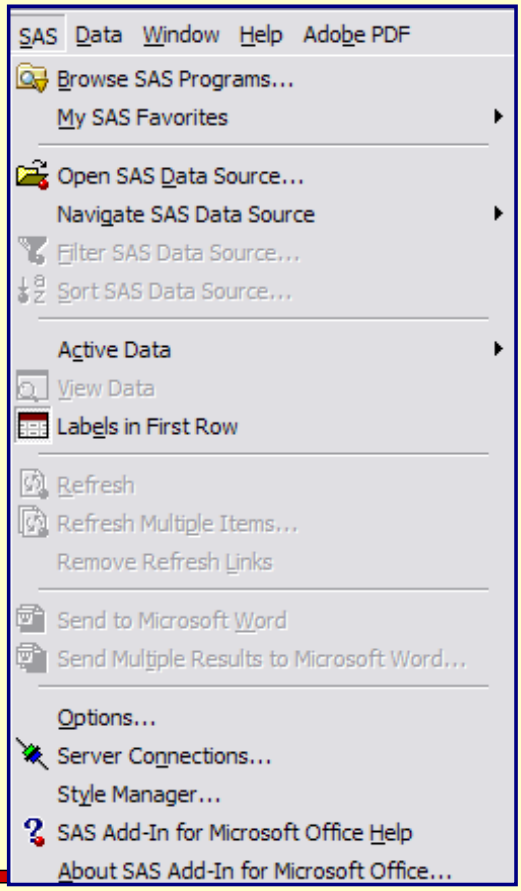
- run stored processes or SAS tasks
- access options for the SAS Add-in
- the ability to specify the server
- use Style manager to customize the appearance of the results returned to Microsoft Excel.



SAS Menu for Microsoft Excel

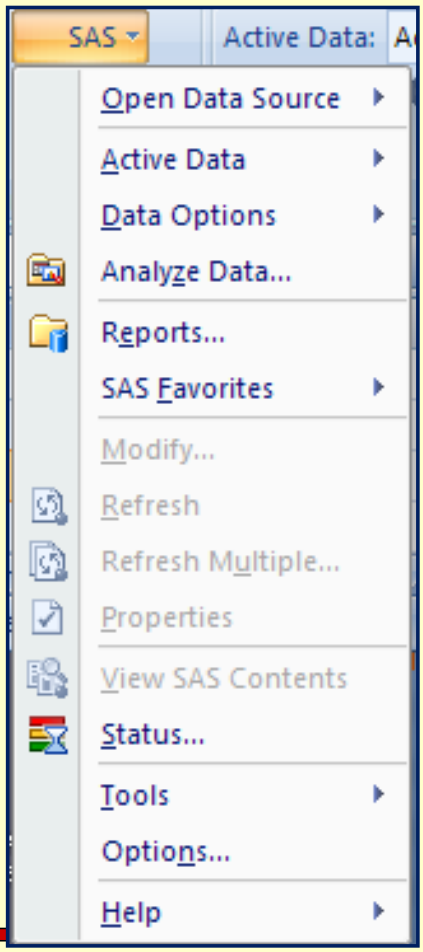
The **SAS menu** is what provides the SAS functionality in Microsoft Excel. The following can be done**:**

- the ability to run stored processes or SAS tasks
- access tools to work with a **SAS data source**
- access options for the SAS Add-in
- the ability to specify the server
- use Style manager to customize the appearance of the results returned to Microsoft Excel.



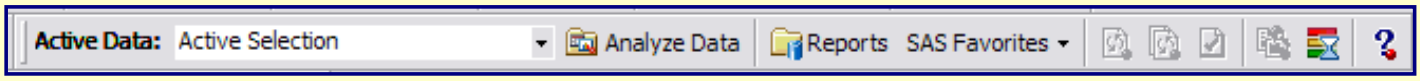
SAS Menu for Microsoft Excel

If you have **Office 2007** products installed on your computer, the **SAS menu** would have the following appearance in Excel.



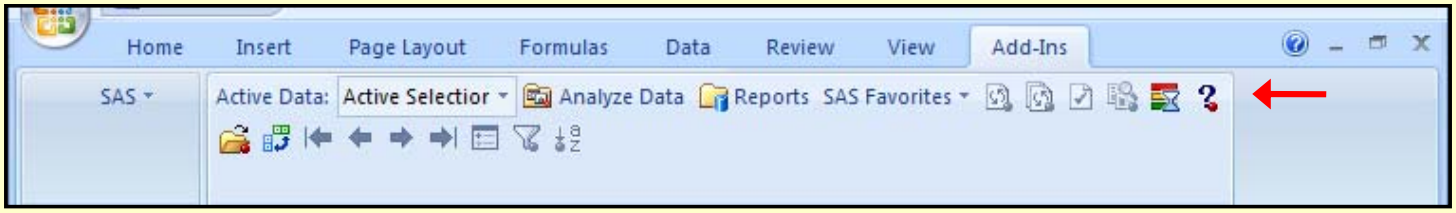
Exploring the **SAS Analysis Toolbar** in **Excel**

The **SAS Analysis Toolbar** provides access to some of the same options as the SAS menu.



Some of the tasks that can be performed include the following:

- changing the active data source
- showing lables in the first row
- refreshing the results
- viewing the data
- browsing SAS programs
- exporting results to Microsoft Word
- accessing your SAS Favorites.



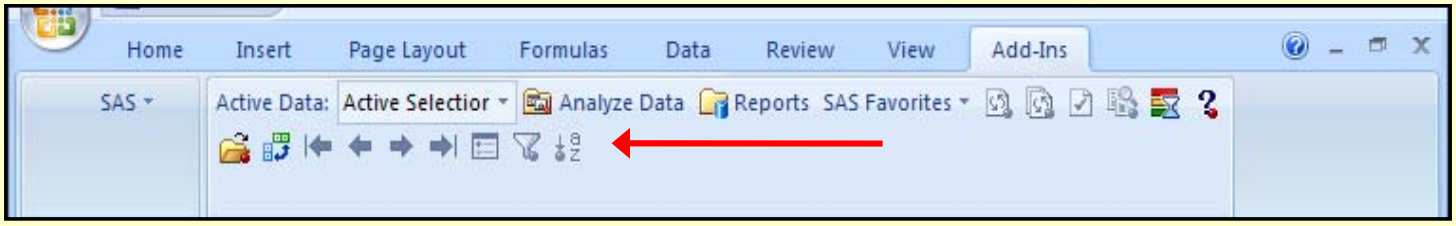
Exploring the **SAS Data Analysis Toolbar** in **Excel**

The SAS DATA Analysis toolbar provides access to options for working with SAS data.

Some of the tasks that can be performed include the following:

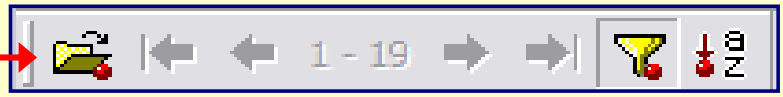
- opening a SAS data source
- navigating through the data
- using a filter
- sorting the data.

The **SAS DATA Analysis** toolbar is not available in Microsoft Word.

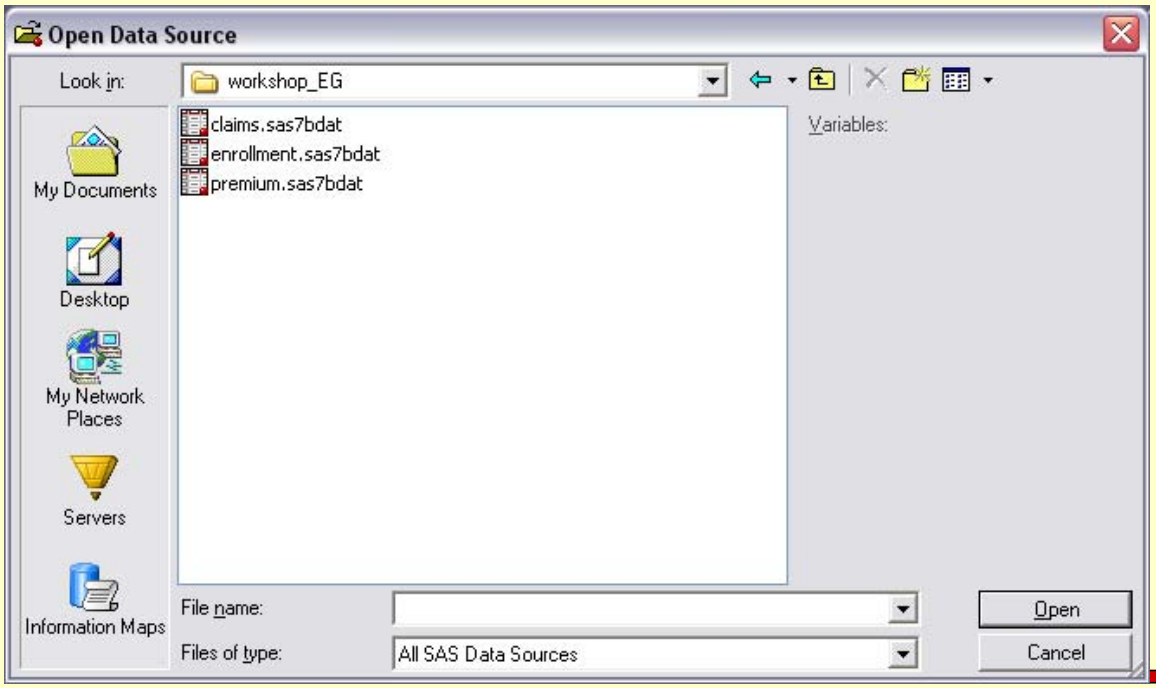


Exploring the **SAS Data Analysis Toolbar** in **Excel**

Filtering SAS Data in Microsoft **Excel**



Select the **'Folder'** (double-clicking) to open a SAS data set then browse to find it, .

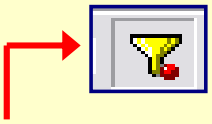


31

.

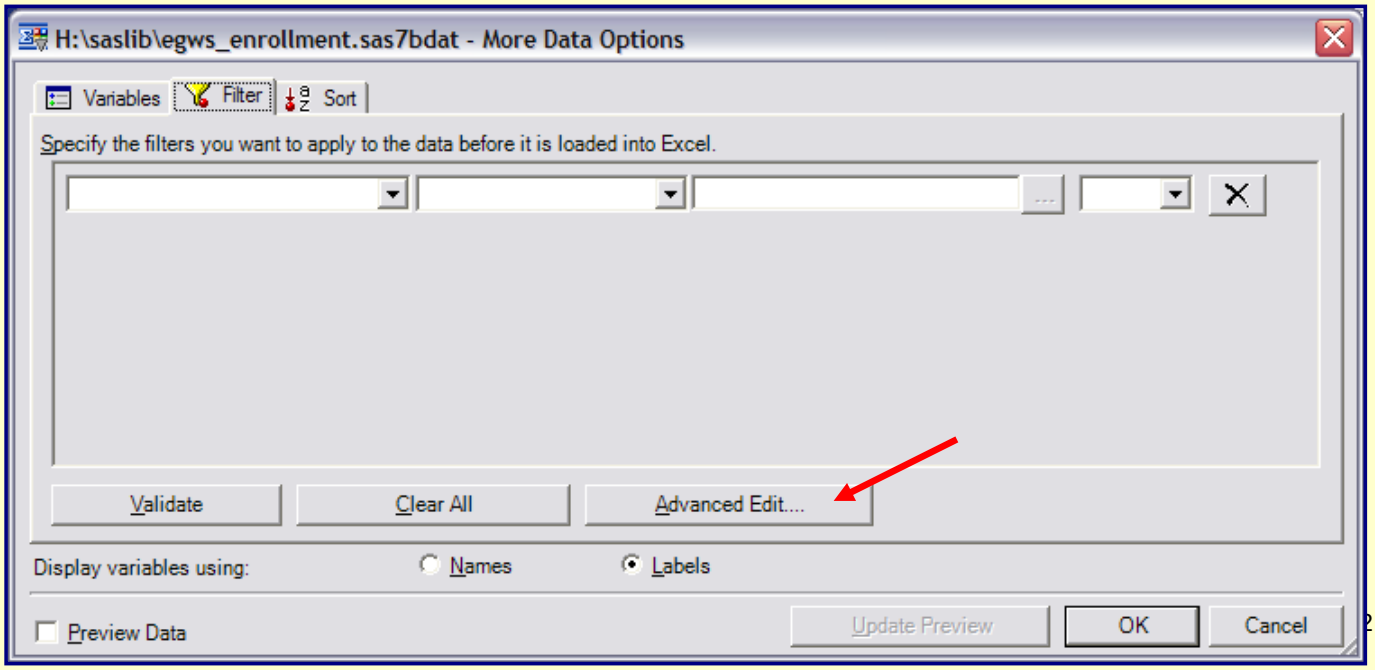
Exploring the **SAS Data Analysis Toolbar** in **Excel**

Filtering SAS Data in Microsoft **Excel**



Select the **'Funnel'** to specify a filter.

Select the **Advanced Edit ...** button.



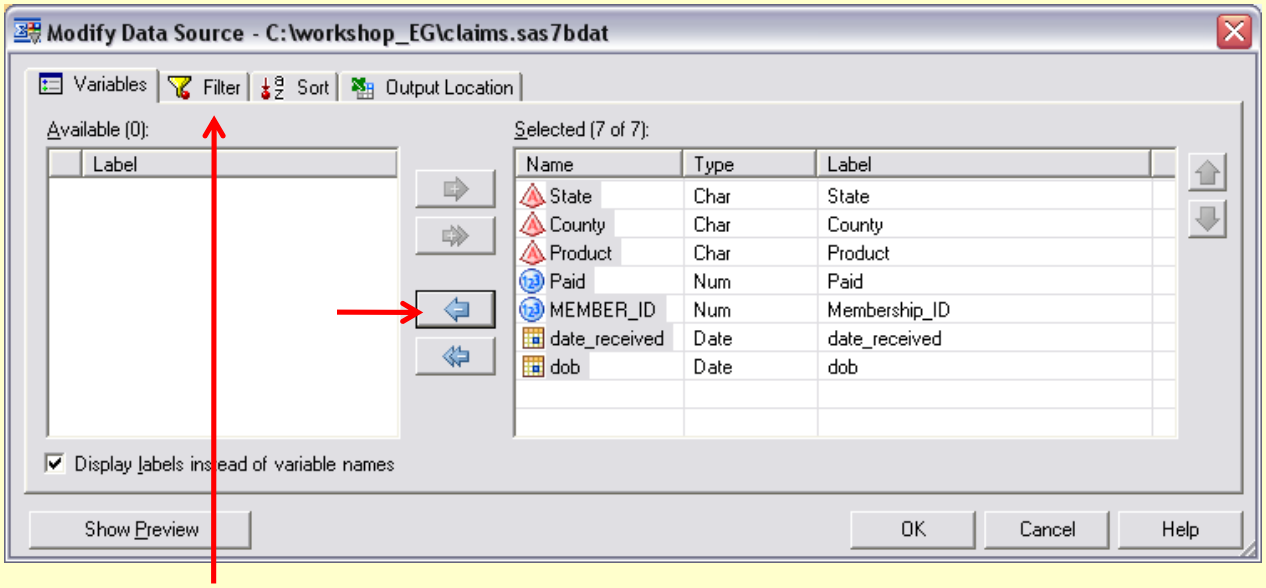
Notice the Advanced Edit pushbutton.

Using the SAS Add-in for Microsoft Office

Add SAS Data to Excel

The SAS Add-in for Microsoft Office allows Excel to access SAS Data from a server or your local machine and add it to an Excel spreadsheet.

1. From an Excel session, select **SAS** ⇨ **Open Data Source** ⇨ **Into Worksheet.**
2. Select the **Claims** data set. Then indicate you want **all** the variables**.**

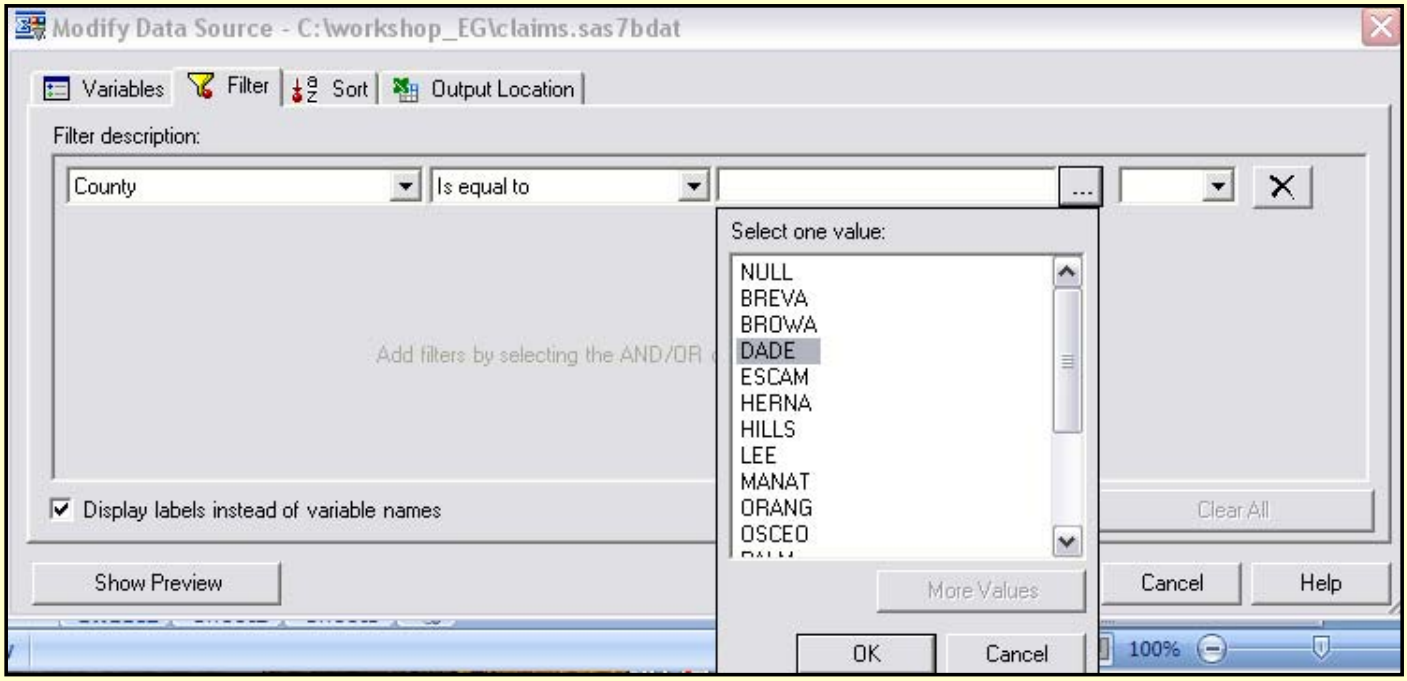


3. Select the **'filter'** tab**.**

Using the SAS Add-in for Microsoft Office

Add SAS Data to Excel

Excel (2003 and earlier) limits the number of rows available in a spreadsheet to 65,536 and the columns to 256. While we are not near that limit here, we still may want to **filter** the data.



4. Filter the data for DADE county. Select **OK,** then **OK** again to run the filter**.**

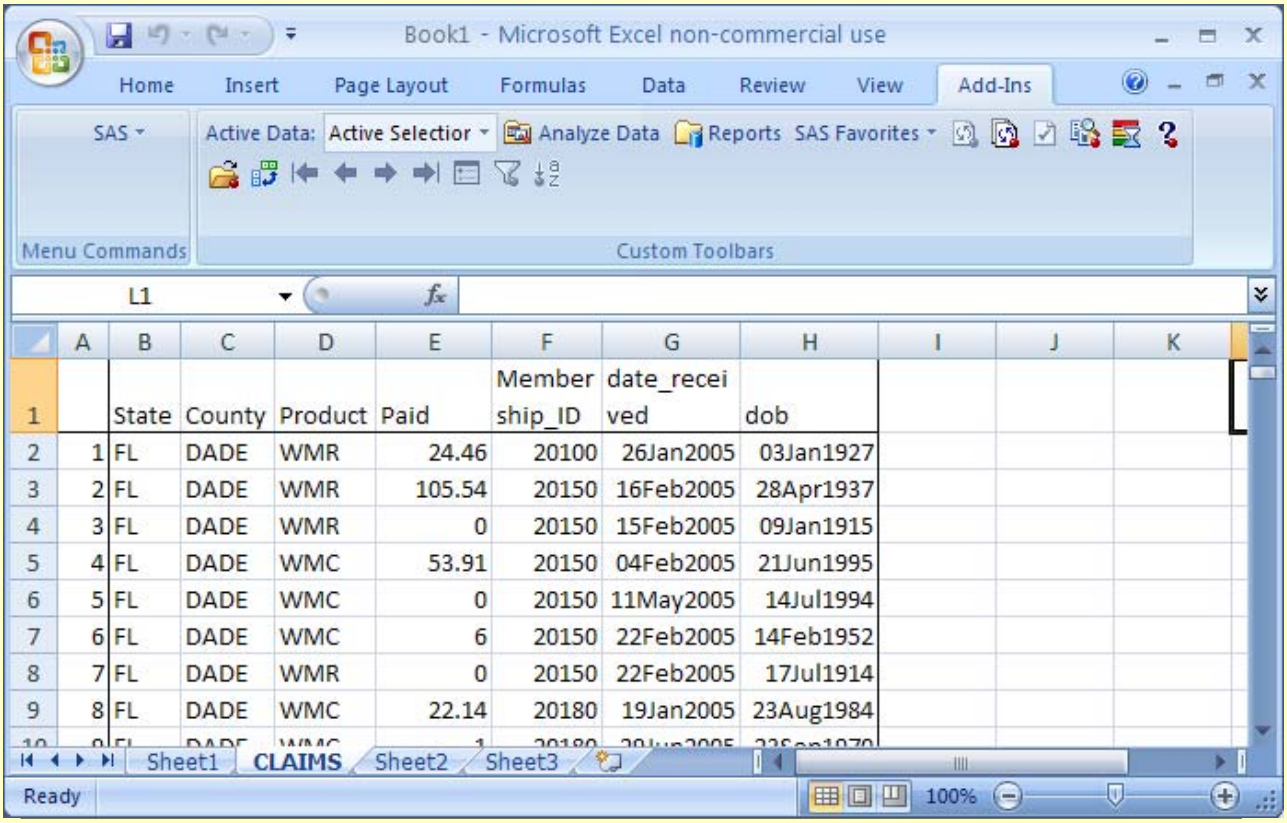
34

For Office 2007 installations, you sort the data from this window.

Using the SAS Add-in for Microsoft Office

Add SAS Data to Excel

4. The filtered data appears in the spreadsheet.



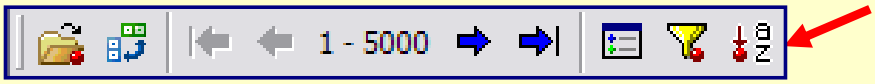
	A	State	County	Product	Paid	Membership_ID	date_received	dob
2	1	FL	DADE	WMR	24.46	20100	26Jan2005	03Jan1927
3	2	FL	DADE	WMR	105.54	20150	16Feb2005	28Apr1937
4	3	FL	DADE	WMR	0	20150	15Feb2005	09Jan1915
5	4	FL	DADE	WMC	53.91	20150	04Feb2005	21Jun1995
6	5	FL	DADE	WMC	0	20150	11May2005	14Jul1994
7	6	FL	DADE	WMC	6	20150	22Feb2005	14Feb1952
8	7	FL	DADE	WMR	0	20150	22Feb2005	17Jul1914
9	8	FL	DADE	WMC	22.14	20180	19Jan2005	23Aug1984

35

Using the SAS Add-in for Microsoft Office

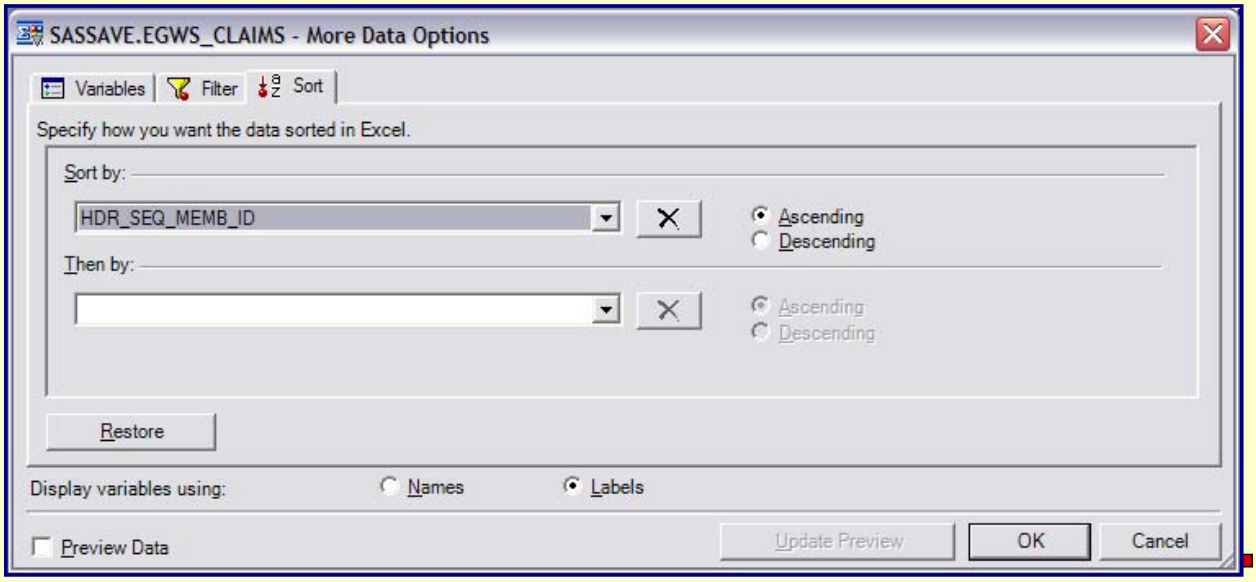
Sorting Data in Excel

1. Select the **Sort** button on the **SAS Data Analysis** Toolbar.



2. Select the downward arrow and choose **HDR_SEQ_MEMB_ID.** Keep Ascending as the default direction. Then select **OK.**

.

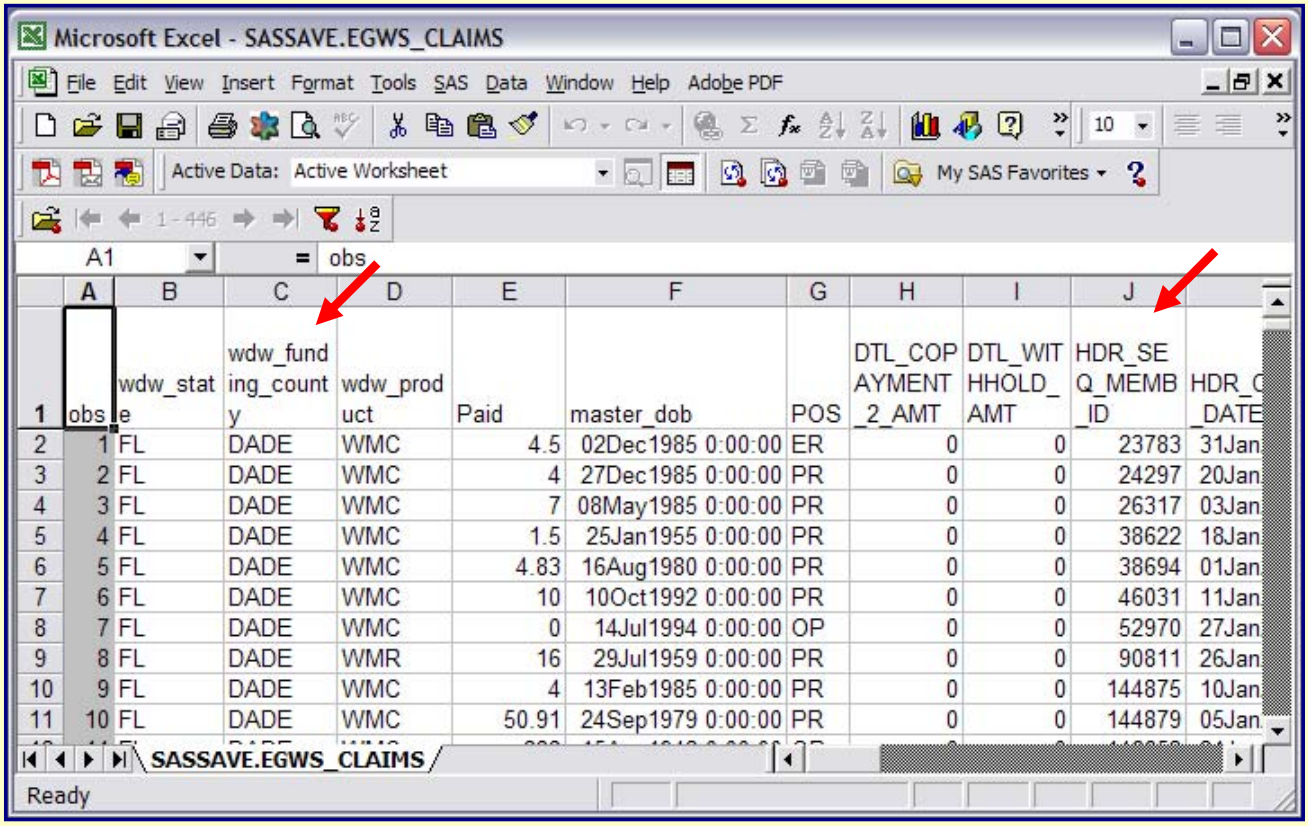


For some installations of Office 2007, you have to sort the data as you bring it into Excel.

Using the SAS Add-in for Microsoft Office

Sorting Data in Excel

3. View the results in the spreadsheet. Notice that it is the Filtered data that is sorted.



	A	B	C	D	E	F	G	H	I	J	
1	obs	wdw_state	wdw_funding_county	wdw_product	Paid	master_dob	POS	DTL_COPAYMENT_2_AMT	DTL_WITHHOLD_AMT	HDR_SEQ_MEMB_ID	HDR_C_DATE
2	1	FL	DADE	WMC	4.5	02Dec1985 0:00:00	ER	0	0	23783	31Jan
3	2	FL	DADE	WMC	4	27Dec1985 0:00:00	PR	0	0	24297	20Jan
4	3	FL	DADE	WMC	7	08May1985 0:00:00	PR	0	0	26317	03Jan
5	4	FL	DADE	WMC	1.5	25Jan1955 0:00:00	PR	0	0	38622	18Jan
6	5	FL	DADE	WMC	4.83	16Aug1980 0:00:00	PR	0	0	38694	01Jan
7	6	FL	DADE	WMC	10	10Oct1992 0:00:00	PR	0	0	46031	11Jan
8	7	FL	DADE	WMC	0	14Jul1994 0:00:00	OP	0	0	52970	27Jan
9	8	FL	DADE	WMR	16	29Jul1959 0:00:00	PR	0	0	90811	26Jan
10	9	FL	DADE	WMC	4	13Feb1985 0:00:00	PR	0	0	144875	10Jan
11	10	FL	DADE	WMC	50.91	24Sep1979 0:00:00	PR	0	0	144879	05Jan

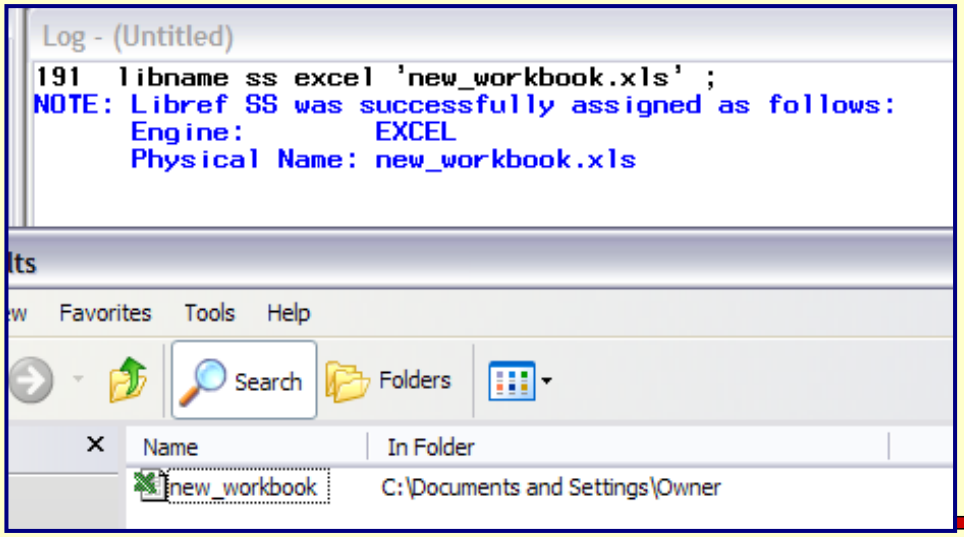
6. Using The EXCEL Engine

The SAS Excel LIBNAME Engine

In Version **9** of the SAS System, new enhancements to the LIBNAME statement provide direct, transparent access to Microsoft Access (97, 2000, or 2002) and Microsoft Excel (95, 97, 2000, or 2002) data.

The typical form of the LIBNAME statement:

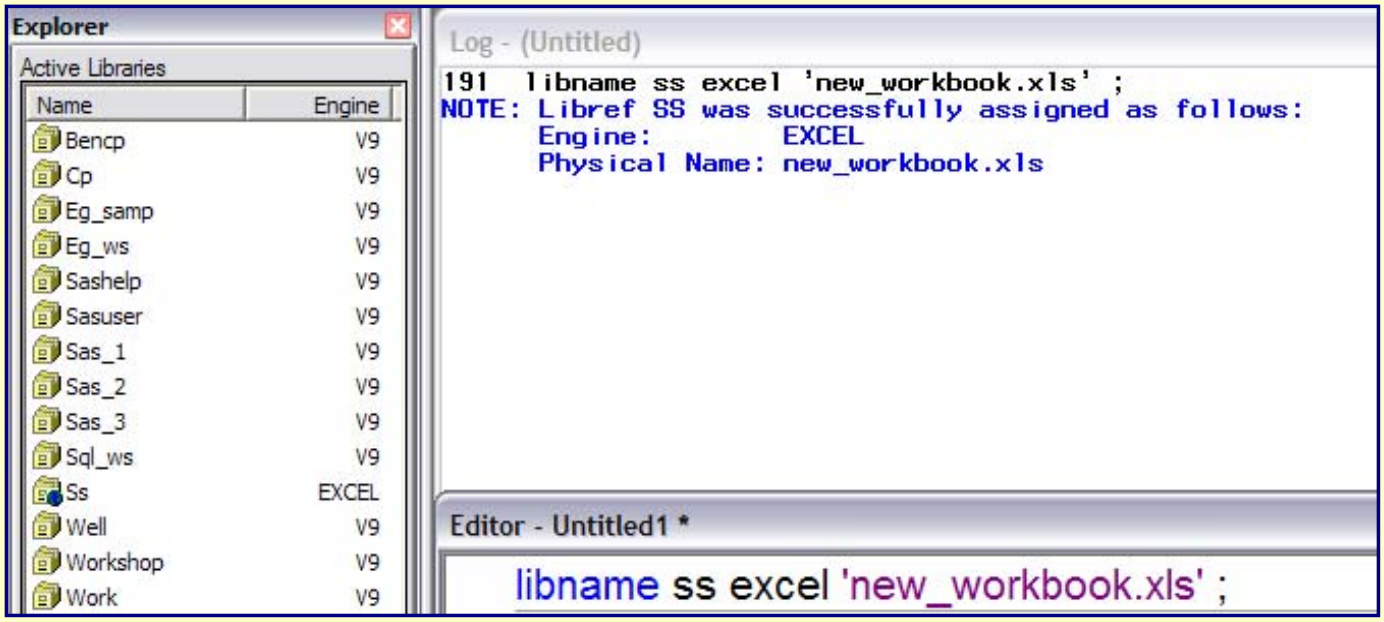
LIBNAME *libref engine-name*
physical file name < libname-options > ;



This LIBNAME statement creates and opens **new_workbook.xls** in the user's default directory. (Notice the path). The number of sheets are limited only by available memory.

The SAS Excel LIBNAME Engine

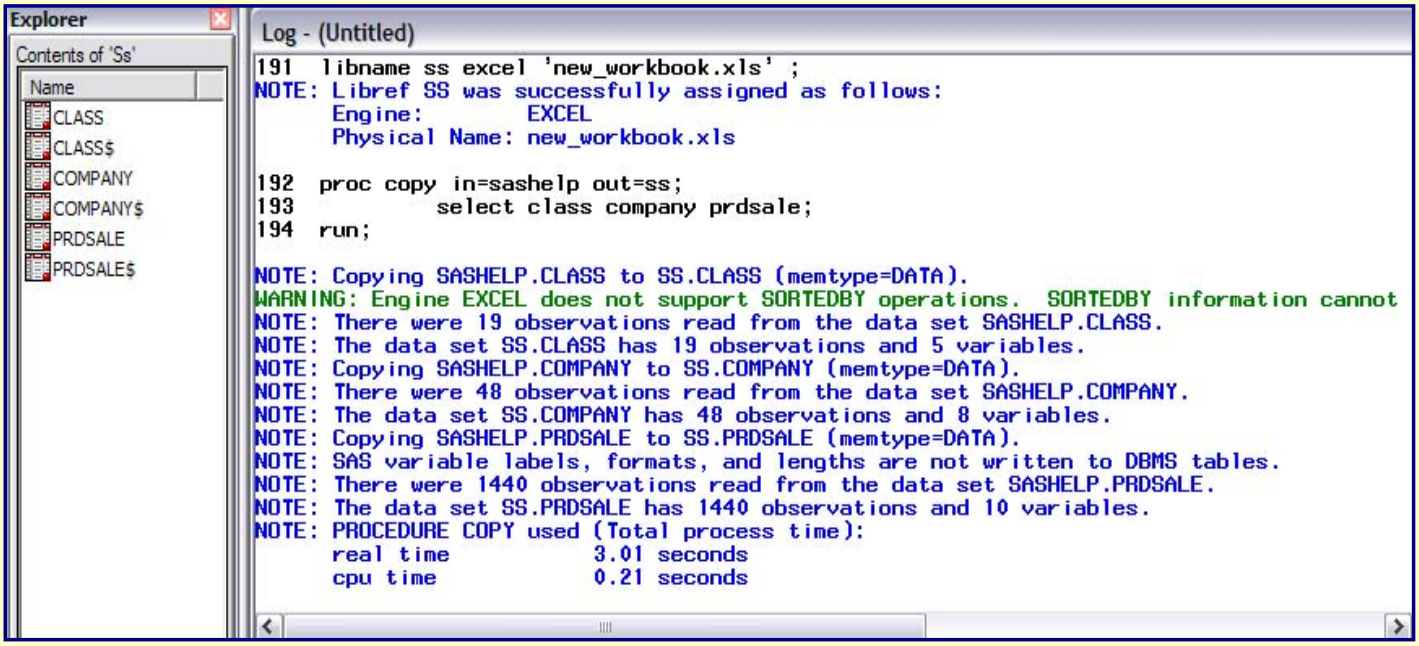
When the LIBNAME statement is issued, then a new SAS library, **SS,** is created. Notice the **SS** library in the SAS Explorer window.



This library is just like any other SAS library. You can write to it as well as read from it.

The SAS Excel LIBNAME Engine

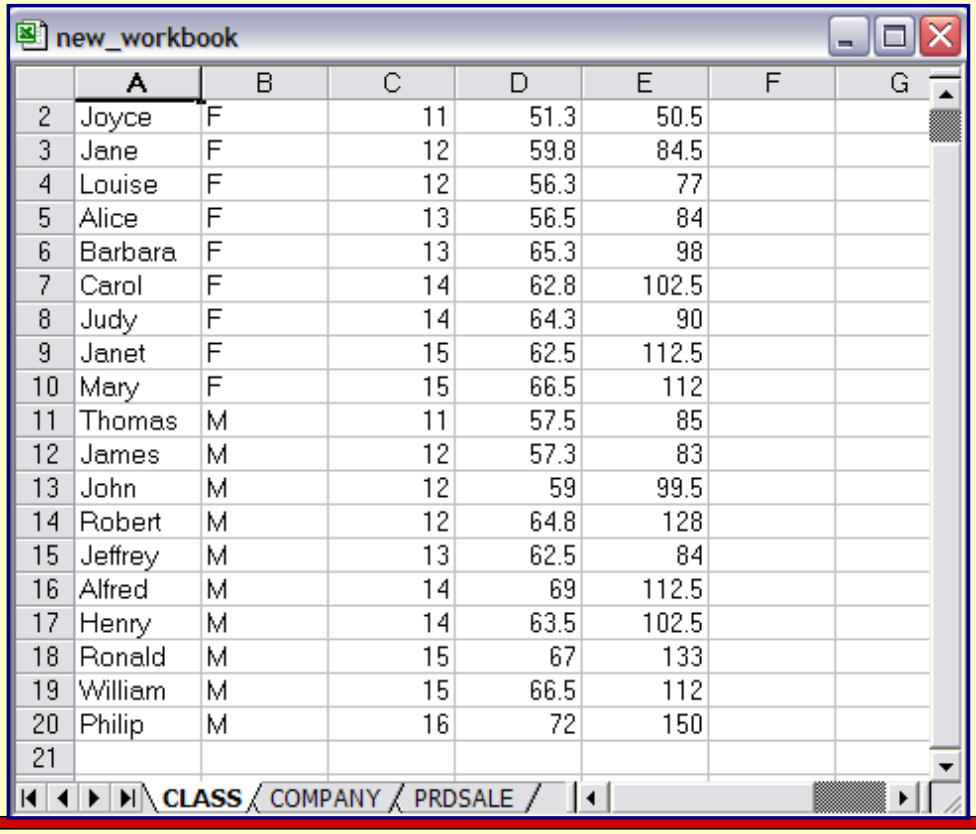
Submit the PROC COPY step below, then open the SS Library. Notice the log.



Notice the **'$'** character on the datasets. Unlike a typical data source, data in an Excel spreadsheet need not be left and top aligned. For this, Excel has ***named ranges*** which allow data to be placed anywhere inside a spreadsheet. By default, SAS reads and writes data from named ranges on spreadsheets, but will also read spreadsheet data directly in the absence of a named range.

The SAS Excel LIBNAME Engine

When the **new_workbook** spreadsheet is accessed, the 'tabs' created from each dataset, can be seen.



	A	B	C	D	E	F	G
2	Joyce	F	11	51.3	50.5		
3	Jane	F	12	59.8	84.5		
4	Louise	F	12	56.3	77		
5	Alice	F	13	56.5	84		
6	Barbara	F	13	65.3	98		
7	Carol	F	14	62.8	102.5		
8	Judy	F	14	64.3	90		
9	Janet	F	15	62.5	112.5		
10	Mary	F	15	66.5	112		
11	Thomas	M	11	57.5	85		
12	James	M	12	57.3	83		
13	John	M	12	59	99.5		
14	Robert	M	12	64.8	128		
15	Jeffrey	M	13	62.5	84		
16	Alfred	M	14	69	112.5		
17	Henry	M	14	63.5	102.5		
18	Ronald	M	15	67	133		
19	William	M	15	66.5	112		
20	Philip	M	16	72	150		
21							

The SAS Excel LIBNAME Engine

When a new SAS dataset is created in an Excel library, SAS creates both a spreadsheet and a named range, each is given the same name, with the spreadsheet denoted by a trailing '$'.

SAS can do the following with an EXCEL engine:

- create new workbooks
- create a new spreadsheet with a named range and write data to the range
- write data to an empty existing named range
- append data to spreadsheet data or named range
- read data from a pre-existing spreadsheet
- read data from a pre-existing named range
- delete all data from a named range
- do the above without Excel on the computer.

The SAS libname engine can not do:

- rename spreadsheets within a workbook
- delete spreadsheets from a workbook, or delete entire workbooks
- change or apply formatting
- write a formula into a cell

The SAS Excel LIBNAME Engine

Use PROC Report to write a separate page to the same spreadsheet. Note the **OUT=** option.

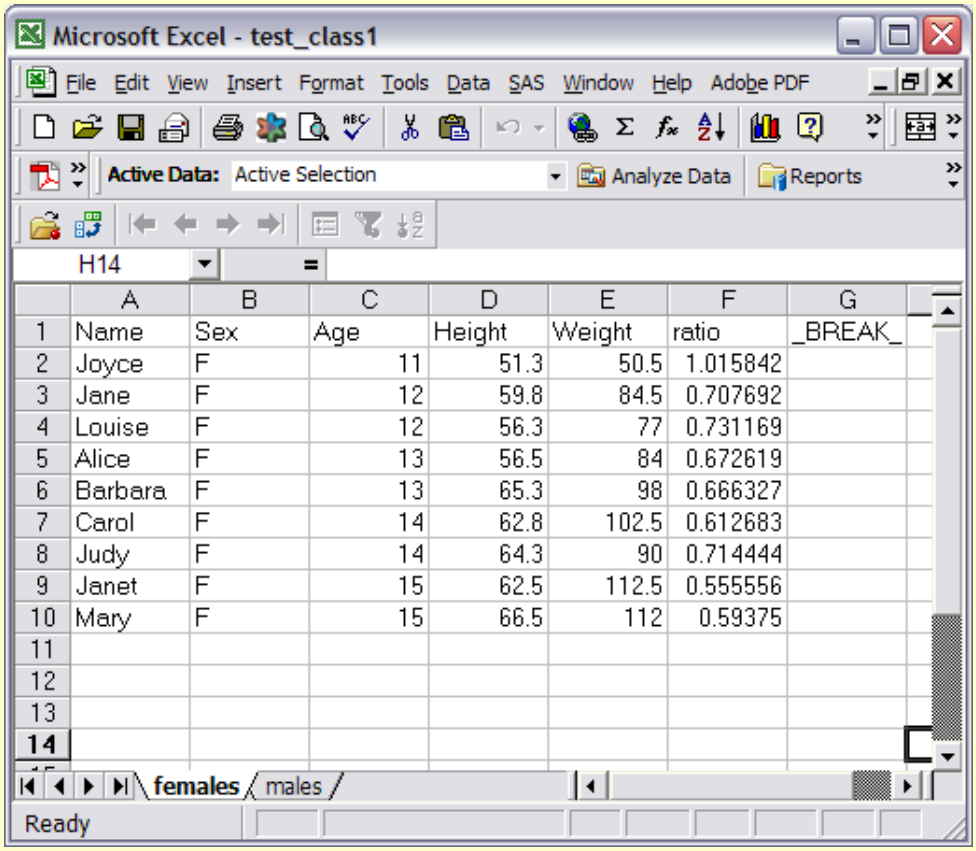
```
libname ss excel 'c:\test_class1.xls';

proc report nowd data=sashelp.class(where=(sex='F')) out=ss.females;
      columns name sex age height weight ratio;
      define height / analysis mean f=6.1;
      define weight / analysis mean f=6.1;
      define ratio / computed  format = 6.2;
      compute ratio;
         ratio = height.mean / weight.mean;
         if ratio > .67 then call define(_row_, 'style', 'style=[background=cyan]');
      endcompute;
run;
proc report nowd data=sashelp.class(where=(sex='M')) out=ss.males;
      columns name sex age height weight ratio;
      define height / analysis mean f=6.1;
      define weight / analysis mean f=6.1;
      define ratio / computed  format = 6.2;
      compute ratio;
         ratio = height.mean / weight.mean;
         if ratio > .67 then call define(_row_, 'style', 'style=[background=cyan]');
      endcompute;
run;

libname ss clear;
```

The SAS Excel LIBNAME Engine

A tab is created for each output dataset from PROC REPORT.



	A	B	C	D	E	F	G
1	Name	Sex	Age	Height	Weight	ratio	_BREAK_
2	Joyce	F	11	51.3	50.5	1.015842	
3	Jane	F	12	59.8	84.5	0.707692	
4	Louise	F	12	56.3	77	0.731169	
5	Alice	F	13	56.5	84	0.672619	
6	Barbara	F	13	65.3	98	0.666327	
7	Carol	F	14	62.8	102.5	0.612683	
8	Judy	F	14	64.3	90	0.714444	
9	Janet	F	15	62.5	112.5	0.555556	
10	Mary	F	15	66.5	112	0.59375	
11							
12							
13							
14							