

TS589B

Importing Excel files to SAS Datasets

There are several ways to import Excel files into SAS.
They are as follows:

Note: Excel 97 format is currently not supported.

1) Using SAS/ACCESS to PC File Formats:

SAS/CORE, SAS/BASE and SAS/Access Interface to PC File Formats must be licensed and installed at your site. The Excel file must be saved as Microsoft Excel 5.0/95 Workbook..

Using a programmatic method:

Use the following code:

```
Proc Access dbms=xls;
Create sasuser.sales.access;
Path='c:\sales.xls';
Worksheet='sheet';      /* optional */
                        /* The following 4 statements are recommended. */
scantype=yes;          /* Scans entire column to determine variable type. */
getnames=yes;          /* Reads first row for column names. */
assign=yes;             /* Assigns SAS variables the names read above. */
mixed=yes;             /* Allows numeric-to-character comparison in mixed data columns. */
create sasuser.sales.view;
select all;
Run;

Data sasuser.new;
Set sasuser.sales;
Run;                    /* Creates a SAS dataset, sasuser.new. */
```

Using the Import Wizard:

(The Import Wizard is only available in SAS 6.12 or later.)

- 1) Save the file as Microsoft Excel 5.0/95 Workbook.
- 2) Click **File** → **Import**
- 3) Select **Excel**(and the version you are trying to read) from the list of file types.
- 4) Click **NEXT**.
- 5) Specify the full path to the Excel file you are trying to import into a SAS dataset.
- 6) Click **NEXT**.
- 7) Pick the Library where you want the dataset to reside.
If library is not listed, you will have to issue a libname statement before using the Import Wizard.
- 8) Name the dataset. Use the list of datasets which reside in that library to overwrite an existing SAS dataset; or type in a new dataset name.
- 9) Click **FINISH**.

2) Using SAS/ACCESS to ODBC:

- 1) SAS/CORE, SAS/BASE and SAS/ACCESS Interface to ODBC must be licensed and installed at your site.
- 2) A Microsoft Excel ODBC driver must be installed and configured on your PC.
- 3) Use the following code to import the Microsoft Excel file into a SAS dataset.

```
Libname sasuser 'c:\foo';

Proc SQL;
Connect to ODBC (prompt);
Create table sasuser.TEST
As select * from connection to ODBC
(select * from tablename);
disconnect from odbc;
quit;
```

- '*Prompt*' will activate a pop-up window where you can choose your data source name. Alternatively, you can use 'DSN=' in the parenthesis and specify your data source name.
- '*sasuser.TEST*' is the name of the SAS dataset you will create from this query.
- '*tablename*' must be the specific unit of data that we are reading. For Excel files, the DSN references the Excel workbook, and the '*tablename*' refers to the sheet name, 'Sheet1\$'.

3) Using the Data Step:

If you do not have SAS/ACCESS Interface to PC File Formats or SAS/ACCESS Interface to ODBC licensed and installed at your site, another option to import the Microsoft Excel file would be with the Data Step. This will require that you create a valid input statement to match the layout of the column ordering and types.

- 1) SAS/CORE and SAS/BASE must be licensed and installed at your site.
- 2) Save the Excel file as an .CSV file (comma separate file).
- 3) Use the following code to read the .CSV file into sas.

```
Filename house 'c:\house.csv';

Data sasuser.house;
Infile house dlm=', ' missover;
Input var1 var2 var3 $ var4;
Run;
```

4) Using Dynamic Data Exchange (DDE):

If you do not have SAS/ACCESS to PC File Formats or SAS/Access Interface to ODBC licensed and installed at your site, another option to import the Microsoft Excel file would be with the DDE facility. DDE is similar to the Datastep method above and requires that an input statement be written to match the layout of your Excel columns.

- 1) SAS/CORE and SAS/BASE must be licensed and installed at your site.
- 2) Invoke Microsoft Excel and open the Excel file you wish to import.
- 3) Use the following code to import the Microsoft Excel file.

```
Filename test dde 'Excel|Sheet1 !R1C1:R50C50' notab;
```

```
Data test;  
Infile test dlm='09'x;  
Input a b $;  
Run;
```

** Reference the "SAS Companion for the Microsoft Windows Environment" or the Tech Support document "TS-325" for more examples of using this method.