Reading SAS Data Using INFILE Options

INFILE Statement: When you want to read the data from external files, The INFILE statement has to be used. The INFILE statement follows the DATA statement and must be followed by the INPUT statement. After the INFILE keyword, the file path and name are enclosed in quotation marks:

```
Data Name;
INFILE ‘path\filename’ <options>;
INPUT var1 $   var2   var3  $   ($ - character variables)…;
Run;
```

Options in the INFILE Statement

- FIRSTOBS = tells SAS at what line to begin reading data.

- OBS= options can be used anytime you want to read only a part of your data file. It tells SAS to stop reading when it gets to that line in the raw data file.

- MISSOVER option tells SAS that if it runs out of data, don’t go to the next data line. Instead, assign missing values to any remaining variable.

- TRUNCOVER When you are using column or formatted input and some data lines are short than others. Then, by default, SAS will go to the next line to start reading the variable’s value. This option tells SAS to read data for the variable until it reaches the end of the data line, or the last column specified in the format or column range, whichever comes first.

Additional Options

Reading Delimited Files with the DATA Step

Delimited files are raw data files that have a special character separating data values. SAS gives you two options for the INFILE statement that make it easy to read delimited data files: The DLM and DSD options.

1, *DLM (DELIMITER)* option in the INFILE statement allows you to read data files with other delimiter other than spaces. Comma and tab are common delimiters.

CSV files are common type of files that separated by comma and can be read with DLM option. (DLM=',')

Tab-separated values files. If the tab instead of comma between data values, we need to use DLM='09'X option. '09'X means a hexadecimal 09 in ASC II.

Additional Options

Reading Delimited Files with the DATA Step

DSD (Delimiter-Sensitive Data) option for the INFILE statement does three things.

- First, it treats two delimiters in a raw as missing value.
- Second, it does not read quotation marks as part of the data value.
- Third, it ignores delimiters in data values enclosed in quotation marks.

The DSD option assumes that the delimiter is a comma. If your delimiter is not comma, then you need the DLM= option plus DSD option to specify the delimiter.

Additional Options

Reading Delimited Files with the DATA Step

- Colon Modifier: “:$N.” or “:mmddyy10” tells SAS to read for the length of the informat, or until it encounters a delimiter, whichever comes first.

Reading external Data Using IMPORT

IMPORT Procedure: This is the another way of entering SAS data - **Reading other software’s data files (et. al. Excel data, Access data..) directly.**

IMPORT procedure can be used to import several types of PC files. For example Excel file. You don’t have to remember the detail codes. You can use the manual on the top of the SAS window, then save the codes in your named program file.

Reading external Data Using IMPORT

```
PROC IMPORT OUT= WORK.examp3
        DATAFILE= "C:\MCG-Lifang\SAS Class\Fall2014\Lecture3\EXIMP.xls"
        DBMS=EXCEL REPLACE;
    RANGE="Sheet1$";
    GETNAMES=YES;
    MIXED=YES;
    SCANTEXT=YES;
    USEDATE=YES;
    SCANTIME=YES;
RUN;
```

Reading external Data Using IMPORT

File → Import Data → Next (xlsx file) → Browse to find data → Click to open the data → Select the table (sheet name) → options → assign the member name → save the program (Import procedure codes).

Corresponding to the IMPORT, there is a procedure called “ EXPORT”:

Export the SAS data into Excel or other data format

```
PROC EXPORT DATA= WORK.EXAMP4
        OUTFILE= "C:\MCG-Lifang\SAS Class\Fall 2014\Lecture3\examp4.xls"
        DBMS=EXCEL REPLACE;
    SHEET="one";
RUN;
```

File → Export Data → Select Library and Member Data Name → Next → Select the type of data you want to export → Next → Browse to find location data will store →name the exported data → name the sheet →save the program.

SAS System Options

Common SAS options:

CENTER , NOCENTER - Controls whether output are centered or left-justified.

Default: CENTER

DATE, NODATE – Controls whether or not today’s date will appear at the top of each page of output. Default: DATE

LINESIZE = n - Controls the maximum length of output lines. Possible values for n are 64 to 256. Default: 98

PAGESIZE = n – Controls the maximum number of lines per page of output. Possible values for n are 15 to 32767. Default: 55

NUMBER, NONUMBER – Controls whether or not page numbers appear on each page of SAS output. Default: NUMBER

Common SAS options – Cont.

ORIENTATION = PORTRAIT - default

ORIENTATION = LANDCAPE

RIGHTMARGIN = n - Specifies size of margin (such as 0.75in or 2cm) to be used for printing output. Default: 0.00in.

LEFTMARGIN = n

TOPMARGIN = n

BOTTOMMARGIN = n

Common SAS options – Cont.

```
PROC OPTIONS;
RUN;
```

Generate all options in the “Log” window!