B2.3 - Importing Data into SAS

To-Do Date: Oct 28 at 11:59pm

Libraries are an incredibly efficient and elegant way for you to directly access data and use it in a progran need to access ***unstructured*** data and to do that, you need to import the file and create a copy as a SAS

Let's start with text files as an example. Text files are simply strings of characters to your computer. SAS c with an engine. We must import the data into a structured format, such as a SAS table, in order to use the

There are a number of ways to import data. If you are interested in a point-and-click approach, Enterprise SAS windowing environment all offer an Import Wizard that enables you to read various file types, specify SAS table. If you would like to see these wizards, I encourage you to try them out in the practices.

But because this is a programming class, we are going to teach you a simple programming option: the IM

PROC IMPORT reads data from an external data source and writes it to a SAS table. SAS can import de character acting as the delimiter. Let's say we want to read a comma-delimited file. You use the DATAFIL and complete file name, the DBMS= option to define the file type as CSV, and the OUT= option to provide SAS output table that you want to create. By default, SAS assumes column names are found in the first ro

Common DBMS identifiers that are included with Base SAS:

- CSV – comma-separated values.
- JMP – JMP files, JMP 7 or later.
- TAB – tab-delimited values.
- DLM – delimited files, default delimiter is a space. To use a different delimiter, use the DELIMITER= statement.

Additional DBMS identifiers included with SAS/ACCESS Interface to PC Files:

- XLSX – Microsoft Excel 2007, 2010 and later
- ACCESS – Microsoft Access 2000 and later

Other DBMS identifiers can be viewed **here (http://go.documentation.sas.com/?docsetId=acpcref&docsetTarget=p0jf3o1i67m044n1j0kz51ifhpvs.htm&docsetVersion=9.4&locale=en#n1s** in the SAS Help Center.

There are some options you might want to specify in the PROC IMPORT step. The REPLACE option can SAS output table should be replaced if it already exists.

By default, SAS scans the first 20 rows of the data to make its best guess for the column attributes, includ possible that SAS might incorrectly assume a column's type or length based on the values found in those GUESSINGROWS= option to provide a set number or use the keyword MAX to examine all rows. SAS sc you specify to determine type and length of each column in the imported table.

Demo: Importing a Comma Delimited (CSV) File

Files needed:

- Demo file: p102d03.sas
- Data file: storm_damage.csv

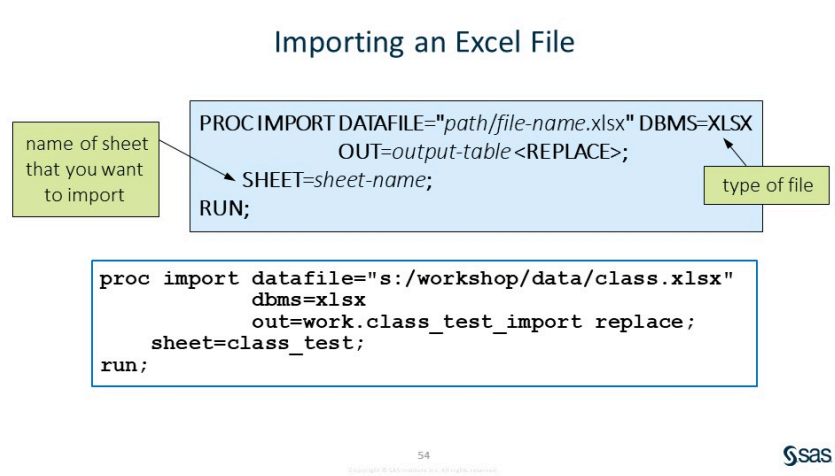
2_3 - Demo - Importing a Comma Delimited (CSV) File.pdf (https://clemson.box.com/s/clttch2n1rxq59h3bvoccfpa2fqx8lcf)

Activity 2.09:

Open **p102a09.sas** (https://clemson.instructure.com/courses/237270/files/23074295/download?wrap=1) from the **activities** folder and perform the following tasks:

1. This program imports a tab-delimited file. Run the program twice and carefully read the log. What is different about the second submission?
2. Fix the program and rerun it to confirm that the import is successful.

Click here for Solution.



You know you can use the XLSX library engine to read and write Excel data directly, but you might prefer to import a copy of your Excel data as a SAS table and use that SAS table in your program. If you have SAS/ACCESS to PC Files licensed, you can use PROC IMPORT to accomplish this. You simply change the DATAFILE= value and the DBMS option

	to reference XLSX. You also need to use the SHEET= option to tell SAS which worksheet you want to read from the workbook. PROC IMPORT can read only one spreadsheet at a time, and by default it reads the first worksheet.

Think about it:

What is the difference between using the XLSX LIBNAME engine and PROC IMPORT to read EXCEL data in a SAS program?