Class Exercise 7

This exercise is based upon Chapter 8 of the SAS Advanced Certification Prep Guide. We will study DICTIONARY.COLUMNS to see what variables it identifies, and typical information that can be extracted from them.

We experimented with DICTIONARY.TABLES in class, but did not explore DICTIONARY.COLUMNS. To start this exercise, I would like you to run all the commands listed under Chapter 8 SAS code (except for the last few related to DICTIONARY.TABLES) so that you create a variety of tables and views in the WORK library.

First look at the Libraries available under the Explorer tab; you’ll likely have WORK, SASUSER, SASHELP, and MAPS, as well as a couple others. You can confirm these libraries with the following clause:

**proc sql;**

**select distinct libname from dictionary.tables;**

**quit;**

We can begin by looking in the LOG for the columns available in DICTIONARY.COLUMNS:

**proc sql;**

**describe table dictionary.columns;**

**quit;**

Remember that DICTIONARY.TABLES had a long list of columns, most of whose purpose was difficult to discern; the columns for DICTIONARY.COLUMNS seem more accessible. The WORK library has a decent number of columns, but not so much that they would be unwieldy to print. If we tried the same command below with SASHELP, we would obtain a much longer (though still manageable) output listing.

**proc sql;**

**select libname, memname, name from dictionary.columns**

**where libname='WORK';**

**quit;**

We can use DICTIONARY.TABLES to search for common column names. The following clause tries to find date-related variables in all available libraries.

**proc sql;**

**select libname, memname, name from dictionary.columns**

**where upcase(name) contains 'DATE';**

**quit;**

How many occurrences of date-related columns did you find? Suppose your WHERE clause was **where name =\* 'DATE'** –how do the two output listings compare?

Were there any “mistakes” in either list—column names related to the string DATE that really weren’t date-related?

We could actually summarize the results by library with the following code:

 **proc sql;**

**select libname, count(\*) as count from dictionary.columns**

**where upcase(name) contains 'DATE'**

**group by libname;**

**quit;**

When looking over the list of columns stored by DICTIONARY.COLUMNS, you may have noticed that the number of entries in each column was not available as a variable. However, DICTIONARY.TABLES does save this information in the column NOBS. We can join entries from both tables so that we can print the number of observations for each column in the WORK library.

**proc sql;**

**select dt.memname, dt.memtype, dt.nobs, nvar, dc.name, dc.length from dictionary.tables as dt inner join dictionary.columns as dc**

**on dt.memname=dc.memname where dt.libname='WORK';**

**quit;**