TransUnion

MAKE SASHELP YOU

Programming can be this easy and fun

AN AN, ADVANCED ANALYTICS INTERN,
TRANSUNION CANADA
MAY 27, 2016

We will discuss

1. What SASHELP is
2. Why we should adopt SASHELP in our everyday programming
3. A few applications to make life easy
4. Quiz
5. Q&A

What is SASHELP?

What is SASHELP?

1. SASHELP tables are data views that collect and retain information about the your current SAS session

2. Information includes library name, active datasets, column name, current system options, macro, libraries, etc.

3. PROC SQL or DATA STEP can be implemented on dictionary/view tables(memtype='VIEW'); All other adapt to PROC SQL or DATA STEP exclusively

4. Read-only

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

Why SASHELP?

Why SASHELP?

1. **SASHELP dictionary views offer an easy way to monitor the activities and status of your current SAS sessions**

2. **Can help implementing automated programming**

3. **Use PROC SQL SELECT INTO: SEPARATED BY or DATA STEP CALL EXECUTE('ARGUMENT;') to create macro variables**

Selected SASHELP VVIEWS

Overview of major SASHELP view tables

		VTABLE	VMEMBER	VCOLUMN	VVIEW	VMACRO	VCATALG
SQL NAME		DICTIONARY.TABLES	DICTIONARY.MEMBERS	DICTIONARY.COLUMNS	DICTIONARY.VIEWS	DICTIONARY.MACROS	DICTIONARY.CATALOGS
MEMBERS	LIBNAME	Y	Y	Y	Y	N	Y
	MEMNAME	Y	Y	Y	Y	N	Y
	MEMTYPE*	DATA or VIEW	DATA or VIEW or CATALOG	DATA or VIEW	VIEW		CATALOG
	NAME	N	N	Y	N	N	N
	TYPE	N	N	Y	N	N	N
	MACRO	N	N	N	N	Y	Y
APPLICATION		Every dataset and their information in every library	Paths on dataset level	Every column and their information in every dataset in every library	All view tables and their respective library, name and engine types	Names and values of all automatic and user-defined variables	Name, location and type of each SAS catalog, including MACRO, FORMAT, etc.
SUBSET*	MEMTYPE='DATA'		VSTABLE				
	MEMTYPE='DATA' or 'VIEW'		VSTABVW				
	MEMTYPE='VIEW'		VSVIEW				
	MEMTYPE='CATALOG'		VSCATALG				

Memtype='VIEW' refers to view/dictionary tables, 'CATALOG' refers to SAS catalogs, 'DATA' refers to all other SAS datasets
Purpose of subsetting is to cutting down rows and columns hence improving code efficiency and utilization of system resource

TransUnion

Real Life Applications

1. Check key variable exists in one or all datasets of a library

- It seems that every SASHELP view tables has a ‘libname’ column, but is this really the case?
- By subsetting SAS.VCOLUMN to extract only VIEW tables, then create Flag to put whichever variable is missing to SAS log
- Part of SAS code
- Snapshot of SAS log

```
PROC SORT   DATA=SASHELP.VCOLUMN
            OUT=VARCHECK(KEEP=memname name);
            WHERE LIBNAME='SASHELP'
            AND UPCASE(SUBSTR(MEMNAME, 1, 1))='V'
            AND MEMTYPE='VIEW';
            BY memname;
RUN;
```

```
WARNING: THE DATASET VALLOPT IS MISSING THE LIBNAME VARIABLE.
NOTE: There were 300 observations read from the data set WORK.VARCHECK.
NOTE: DATA statement used (Total process time):
      real time               0.03 seconds
      cpu time                0.04 seconds
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

2. Look up path of source data

- Where is my work datasets located?
- By using SASHELP.VSLIB, which contains libname and path, initiate a local macro to have SAS data step automatically create macro variable from libname and path as value
- Part of SAS code
- Simple sysfunc and tranwrd functions to put result
- Snapshot of SAS log

```
%LOCAL Y;
%LET Y=%SCAN(&DS., 1, .);
%PUT Y=&Y.; /* Y=WORK */
%PUT &&Y.; /* && enables SAS to pass macro Y within this local macro */
%PUT &&&Y.; /* &&& passes parameter WORK*/

%LET PRINTDS=%SYSFUNC(TRANWRD(&DS., &Y., &&&Y.)).sas7bdat;
%PUT PRINTDS=&PRINTDS.;
```

```
37     %GETPATH(WORK.VAR);
MPRINT(GETPATH):   DATA _NULL_;
MPRINT(GETPATH):   SET SASHELP.VSLIB;
MPRINT(GETPATH):   CALL SYMPUT(LIBNAME, COMPRESS(PATH));
MPRINT(GETPATH):   RUN;

NOTE: There were 7 observations read from the data set SASHELP.VSLIB.
NOTE: DATA statement used (Total process time):
      real time           0.12 seconds
      cpu time            0.12 seconds


Y=WORK
Y=WORK
C:\SASTemporaryFiles\_TD26688_LCANPB02613V_
PRINTDS=C:\SASTemporaryFiles\_TD26688_LCANPB02613V_.VAR.sas7bdat
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

3. Compare number of observations in datasets

- Check a new dataset against a known dataset for correct number of observations
- Ability to check multiple datasets in one macro with DO LOOP

```
%LOCAL I;

%LET I=1;

%DO %WHILE
(%SCAN(%STR(&DS.), &I.,
%STR( )) ^= ); /* DO LOOP
will execute as long as this
expression is true */
%LET Y=%SCAN(
%STR(&DS.), &I., %STR( ));


%LET I=%EVAL(&I.+1);
```

```
728   %CHECKDS(SASHELP, VVIEW VSVIEW);
MPRINT(CHECKDS):   PROC SQL NOPRINT;
MPRINT(CHECKDS):   SELECT COUNT(DISTINCT MEMNAME) INTO: MEMCOUNT FROM SASHELP.VMEMBER WHERE MEMTYPE='VIEW';
MPRINT(CHECKDS):   QUIT;
NOTE: PROCEDURE SQL used (Total process time):
      real time               0.07 seconds
      cpu time                0.07 seconds


39
MPRINT(CHECKDS):   DATA _NULL_;
MPRINT(CHECKDS):   SET SASHELP.VTABLE;
MPRINT(CHECKDS):   WHERE UPCASE(LIBNAME)=UPCASE("SASHELP") AND UPCASE(MEMNAME)=UPCASE("VVIEW");
MPRINT(CHECKDS):   IF NOBS < 39 THEN PUT "WAR" "NING: THE DATASET VVIEW HAS LESS THAN       39 OBS";
MPRINT(CHECKDS):   ELSE IF NOBS > 39 THEN PUT "WAR" "NING: THE DATASET VVIEWHAS MORE THAN       39 OBS";
MPRINT(CHECKDS):   RUN;

WARNING: THE DATASET VVIEW HAS LESS THAN       39 OBS
NOTE: There were 1 observations read from the data set SASHELP.VTABLE.
      WHERE (UPCASE(LIBNAME)='SASHELP') and (UPCASE(MEMNAME)='VVIEW');
NOTE: DATA statement used (Total process time):
      real time               0.57 seconds
      cpu time                0.57 seconds


MPRINT(CHECKDS):   DATA _NULL_;
MPRINT(CHECKDS):   SET SASHELP.VTABLE;
MPRINT(CHECKDS):   WHERE UPCASE(LIBNAME)=UPCASE("SASHELP") AND UPCASE(MEMNAME)=UPCASE("VSVIEW");
MPRINT(CHECKDS):   IF NOBS < 39 THEN PUT "WAR" "NING: THE DATASET VSVIEW HAS LESS THAN       39 OBS";
MPRINT(CHECKDS):   ELSE IF NOBS > 39 THEN PUT "WAR" "NING: THE DATASET VSVIEWHAS MORE THAN       39 OBS";
MPRINT(CHECKDS):   RUN;

WARNING: THE DATASET VSVIEW HAS LESS THAN       39 OBS
NOTE: There were 1 observations read from the data set SASHELP.VTABLE.
      WHERE (UPCASE(LIBNAME)='SASHELP') and (UPCASE(MEMNAME)='VSVIEW');
NOTE: DATA statement used (Total process time):
      real time               0.45 seconds
      cpu time                0.45 seconds
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

4. Compare datasets between directories

- Make sure datasets in different directories are same

- The magic is to let SAS data step execute macro for you

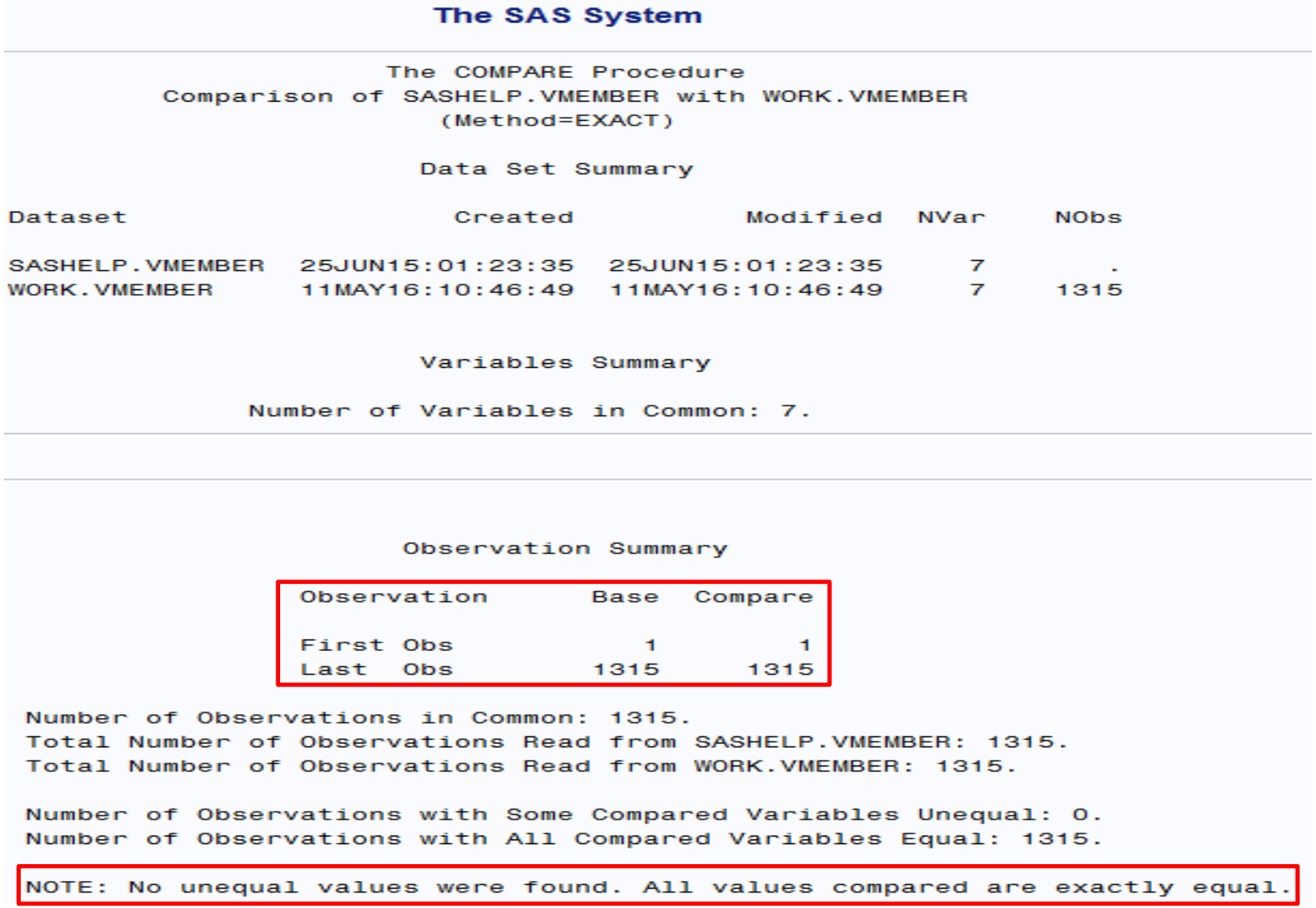
```
DATA _NULL_;
        SET SASHELP.VTABLE;
        WHERE MEMTYPE='VIEW';
        CALL EXECUTE('%COMPARE(' || MEMNAME || ');');
RUN;
```

- Result

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

4. Compare datasets between directories



The SAS System

The COMPARE Procedure
Comparison of SASHELP.VMEMBER with WORK.VMEMBER
(Method=EXACT)

Data Set Summary

Dataset	Created	Modified	NVar	NObs
SASHELP.VMEMBER	25JUN15:01:23:35	25JUN15:01:23:35	7	.
WORK.VMEMBER	11MAY16:10:46:49	11MAY16:10:46:49	7	1315

Variables Summary

Number of Variables in Common: 7.

Observation Summary

Observation	Base	Compare
First Obs	1	1
Last Obs	1315	1315

Number of Observations in Common: 1315.
Total Number of Observations Read from SASHELP.VMEMBER: 1315.
Total Number of Observations Read from WORK.VMEMBER: 1315.

Number of Observations with Some Compared Variables Unequal: 0.
Number of Observations with All Compared Variables Equal: 1315.

NOTE: No unequal values were found. All values compared are exactly equal.

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

5. Check for existing formats

- Utilizing SASHELP.VCATALG

```
PROC SQL;
                SELECT *
                FROM DICTIONARY.CATALOGS
                WHERE UPCASE(MEMTYPE)='CATALOG'
                AND UPCASE(MEMNAME) LIKE 'FORMAT%'
                ;
QUIT;
```

Library Name	Member Name	Member Type	Object Name	Object Type	Object Description	Date Created	Date Modified	Object Alias	Library Concatenation Level
WORK	FORMATS	CATALOG	NUM	FORMAT		17MAY16:09:39:47	17MAY16:09:39:47		0

Sources: Michael Davis, You Could Look It Up

QUIZ

QUIZ

1. How many SASHELP dictionary views are there in SAS 9.4?

2. What is the name of dataset that summarized entire SASHELP?

3. Can SASHELP.VBPLAYERS be used in PROC SQL?

4. Why doesn't following code work?

```
DATA X;
   SET Dictionary.members;
RUN;
```


ANSWERS

1. 39
2. SASHELP.VDCTNRY (DICTIONARY.DICTIONARIES)
3. YES
4. FAIL TO FOLLOW SAS NAME RULES IN LIBREF

Q&A

TransUnion

Special Thanks to

Advanced Analytics Dept. at Transunion Canada
GHSUG

Contact Info

Appendix
Complete SAS codes

Know your datasets - Vmember vs VTable

VIEWTABLE: Sashelp.Vmember

	Library Name	Member Name	Member Type	DBMS Member Type	Engine Name	Indexes	Pathname
1	FMTLIB				V9		C:\Production\Common\code\Formats
2	SASHELP	AACOMP	DATA		V9	yes	('C:\Program Files\SASHome\SASFoundation\9.4\nls\en\SASCFG' 'C:\Program Files\SASHome\SASFoundation\9.4\core\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\aacomp\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\cmp\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\graph\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\spdsclient\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\stat\sashelp')
3	SASHELP	AARFM	DATA		V9	yes	('C:\Program Files\SASHome\SASFoundation\9.4\nls\en\SASCFG' 'C:\Program Files\SASHome\SASFoundation\9.4\core\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\aacomp\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\cmp\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\graph\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\spdsclient\sashelp' 'C:\Program Files\SASHome\SASFoundation\9.4\stat\sashelp')

VIEWTABLE: Sashelp.Vtable

	Member Name	Member Type	DBMS Member Type	Data Set Label	Data Set Type	Date Created	Date Modified	Number of Physical Observations	Observation Length	Number of Variables	Type of Password Protection	Compression Routine	Encryption	Number of Pages	Size of File
1	AACOMP	DATA			DATA	25JUN15:01:05:47	25JUN15:01:05:47	2020	1269	4	--	CHAR	NO	5	393216
2	AARFM	DATA			DATA	25JUN15:01:07:08	25JUN15:01:07:08	61	1269	4	--	CHAR	NO	3	262144
3	ADSMSG	DATA			MSGFILE	25JUN15:01:09:46	25JUN15:01:09:46	426	280	6	--	NO	NO	3	262144
4	AFMSG	DATA			MSGFILE	25JUN15:01:06:18	25JUN15:01:06:18	1090	288	6	--	NO	NO	6	458752
5	ASSCMGR	DATA			DATA	25JUN15:01:19:20	25JUN15:01:19:20	402	592	19	--	NO	NO	4	327680

- Both contain libname, dataset name, memtype
- Vtable provides dataset information on number of obs, variables, etc.
- Vmember provides path of source
- Subsetting

TransUnion

Know your SAS variables

VIEWTABLE: Sashelp.Vcolumn

	Library Name	Member Name	Member Type	Column Name	Column Type	Column Length	Column Position	Column Number in Table	Column Label	Not NULL?	Column Format
1	SASHELP	AACOMP	DATA	locale	char	5	4	1		no	
2	SASHELP	AACOMP	DATA	key	char	60	9	2		no	
3	SASHELP	AACOMP	DATA	lineno	num	4	0	3		no	
4	SASHELP	AACOMP	DATA	text	char	1200	69	4		no	
5	SASHELP	AARFM	DATA	locale	char	5	4	1		no	
6	SASHELP	AARFM	DATA	key	char	60	9	2		no	
7	SASHELP	AARFM	DATA	lineno	num	4	0	3		no	
8	SASHELP	AARFM	DATA	text	char	1200	69	4		no	
9	SASHELP	ADSMSG	DATA	MSGID	num	8	0	1		no	
10	SASHELP	ADSMSG	DATA	MNEMONIC	char	32	12	2		no	
11	SASHELP	ADSMSG	DATA	LINENO	num	4	8	3		no	
12	SASHELP	ADSMSG	DATA	LEVEL	char	1	44	4		no	

- Contains libname, dataset name, memtype
- No catalog information as they are existent in SAS dataset format
- Subsetting

TransUnion

1. Check key variable exists in one or all datasets of a library

```
/*=====================>Check a key variable in one or all datasets in one library<=========================*/

PROC SORT DATA=SASHELP.VCOLUMN OUT=VARCHECK(KEEP=memname name);
                    WHERE LIBNAME='SASHELP'
                    AND UPCASE(SUBSTR(MEMNAME, 1, 1))='V'
                    AND MEMTYPE='VIEW';
                    /* Where clause to subset SASHELP.VCOLUMN hence reducing computer run time and resource */
                    BY memname;
RUN;

DATA _NULL_;
                    RETAIN FLAG 0;
                    SET VARCHECK;
                    BY MEMNAME;
                    /* Create Flag and output */
                    IF FIRST.MEMNAME THEN FALG=0;
                    IF UPCASE(NAME)='LIBNAME' THEN FLAG=1;
                    IF LAST.MEMNAME AND FLAG=0 THEN PUT "WAR" "NING: THE DATASET " memname "IS MISSING THE LIBNAME VARIABLE.";
RUN;
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

2. Look up path of source data

```
/*==================================>Look up the path of source data<====================================*/

%MACRO GETPATH(DS);
                DATA _NULL_;
                                SET SASHELP.VSLIB;
                                CALL SYMPUT(LIBNAME, COMPRESS(PATH)); /* Create macro variables with their values from libname and path*/
                RUN;

                %LOCAL Y;
                %LET Y=%SCAN(&DS., 1, .);
                %PUT Y=&Y.; /* Y=WORK */
                %PUT &&Y.; /* && enables SAS to pass macro Y within this local macro */
                %PUT &&&Y.; /* &&& passes parameter WORK*/

                %LET PRINTDS=%SYSFUNC(TRANWRD(&DS., &Y., &&&Y.)).sas7bdat;
                %PUT PRINTDS=&PRINTDS.;
%MEND;

%GETPATH(WORK.VAR);
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

TransUnion

3. Compare number of observations in datasets

```
/*=====================================>Compare number of observations in datasets<=================================*/

%MACRO CHECKDS(lib, ds);
                PROC SQL NOPRINT;
                                SELECT COUNT(DISTINCT MEMNAME) INTO: MEMCOUNT
                                FROM &LIB..VMEMBER
                                WHERE MEMTYPE='VIEW';
                QUIT;

                %PUT &MEMCOUNT.;

                %LOCAL I;
                %LET I=1;
                %DO %WHILE (%SCAN(%STR(&DS.), &I., %STR( )) ^= ); /* DO LOOP will execute as long as this expression is true */
                                %LET Y=%SCAN( %STR(&DS.), &I., %STR( ));

                DATA _NULL_;
                SET SASHELP.VTABLE;
                WHERE UPCASE(LIBNAME)=UPCASE("&LIB.") AND UPCASE(MEMNAME)=UPCASE("&Y.");
                IF NOBS < &MEMCOUNT. THEN PUT "WAR" "NING: THE DATASET &Y. HAS LESS THAN &MEMCOUNT. OBS";
                ELSE IF NOBS > &MEMCOUNT. THEN PUT "WAR" "NING: THE DATASET &Y.HAS MORE THAN &MEMCOUNT. OBS";
                RUN;

                %LET I=%EVAL(&I.+1); /* I=2 will render do loop express false, hence stop it from executing */
                %END;
%MEND;

%CHECKDS(SASHELP, VVIEW VSVIEW);
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

4. Compare datasets between directories

```
/*================================>Compare datasets between directories toc check for modfication<====================*/

DATA VMEMBER; SET SASHELP.VMEMBER; RUN;

%MACRO COMPARE(DS);
                PROC COMPARE BASE=SASHELP.&DS. COMPARE=WORK.&DS.; RUN;
%MEND;

DATA _NULL_;
                SET SASHELP.VTABLE;
                WHERE UPCASE(LIBNAME)='SASHELP'
                AND MEMTYPE='VIEW';
                CALL EXECUTE('%COMPARE(' || MEMNAME || ');'); /* Call execute invokes macros automatically row by row */
RUN;
```

Sources: Stacey Phillips and Gary Klein, MWSUG 15-P2

5. Check existing formats

```
/*=====================================>Check for existing formats<=====================================*/

/* Create a format */
PROC FORMAT; VALUE NUM 1 = '1'; RUN;

/* Locate this format */
PROC SQL;
                    SELECT *
                    FROM DICTIONARY.CATALOGS
                    WHERE UPCASE(MEMTYPE)='CATALOG'
                    AND UPCASE(MEMNAME) LIKE 'FORMAT%'
                    ;
QUIT;
```

Sources: Michael Davis, You Could Look It Up

TransUnion

