INTRODUCTION

This poster presents three macros to be used with SAS/AF®: CKVAR, CKDSN, and CKNBR.

CKVAR checks that proper SAS® nomenclature has been used for naming a variable. This macro is used in situations where the user specifies a variable range (for example, NUMBER1–NUMBER8). The NAME attribute finds an error because the variable exceeds a length of 8. CKVAR parses the string into two variable names and checks each one for proper syntax.

CKDSN checks that proper SAS nomenclature has been used for naming an output dataset. Once verified, the output dataset name is compared to the datasets that exist on the user’s A disk. If a dataset of the same name is found, the user must use another dataset name. CKDSN is analogous to the INPUT attribute except it is used to check OUTPUT datasets.

CKNBR validates a number that has been parsed from a string of information. This is analogous to the DATATYP macro function except it permits the use of signs and decimal points.

THE MACRO CODE

These macros can be placed in a MACLIB and simply invoked within the SAS/AF® programs.

%MACRO CKVAR (NAME);
%* Define valid characters;
%LET CHARS=ABCDEFGHIJKLMNOPQRSTUVWXYZ;
%LET ALL=&CHARS.0123456789;
%* determine length of SAS name;
%LET LEN=%LENGTH(%QUOTE(%SUPERQ(NAME»);
%IF 1 <= &LEN AND &LEN <= 8 %THEN %DO;
%* first char must be letter;
%LET FRSTCHAR=%SUBSTR(%QUOTE(%SUPERQ(NAME),1,1);
%LET ERRNBR=%VERIFY(&FRSTCHAR,&CHARS);
%IF &ERRNBR > 0 %THEN %LET ERRNBR=3;
%ELSE %LET ERRNBR=2;
%END;
%ELSE %LET ERRNBR=3;
%MEND CKVAR;

%MACRO CKDSN(DSNAME);
%LET VALID = ABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789; 
%* gross check for errors;
%LET ERRNBR=%VERIFY(%QUOTE(%SUPERQ(DSNAME»,&VALID);
%IF &ERRNBR>0 %THEN %DO;
%LET ERRNBR=3;
%MEND CKDSN;

%MACRO CKNBR (NBR);
%* Acceptable chars for first position;
%LET CHARI=0123456789+-;
%LET SIGN=+-;
%IF %DATATYP(&NBR) = CHAR %THEN %DO;
%LET ERRNBR=1;
%* char valid?
%IF %INDEX(&SIGN,%SUBSTR(&NBR,1,1)) > 0 %THEN %DO;
%LET ERRNBR=3;
%LET NBR=%SUBSTR(&NBR,2,%LEN);
%END;
%IF %DATATYP(&NBR) = NUMERIC %THEN %LET ERRNBR=0;
%ELSE %DO:
%LET ERRNBR=1;
%* Decompose decimal into two integers;
%LET NBR1=%SCAN(&NBR,1,.);
%LET NBR2=%SCAN(&NBR,2,.);
%LET ERRNBR=1;
%END;
%MEND CKNBR;
USING THE MACROS

Illustrated below are examples SAS/AF programs which make use of the macros described above.

Illustrate use of CKVAR

%MACRO CKVAR;
%IF &_DCALL = INITIAL
%THEN %DO;
exclusive screen variables to missing;
%LET _DMSG = ;
%LET _DERRON = ;
%LET _DCURSOR = ;
%LET DSET = ;
%LET FROMVAR = ;
%LET TOVAR = ;
%END;
%ELSE %DO;
valid SAS nomenclature?;
%CKVAR(&FROMVAR);
%END;
%MACRO DRIVER;

%MACRO DRIVER;

Illustrate use of CRDSN

%MACRO CRDSN;
%IF &_DCALL = INITIAL
%THEN %DO;
* Initialize screen variables to missing;
%LET _DMSG = ;
%LET _DERRON = ;
%LET _DCURSOR = ;
%LET DSET = ;
%LET OUTDS = ;
%END;
%ELSE %DO;
valid SAS nomenclature?;
%CKVAR(&VNAME);
%END;
%MACRO DRIVER;

#MACRO DRIVER;
**CONCLUSION**

In summary, these three macros can help in handling the "housekeeping" functions which, although critical, clutter the code and hide the problem which is being solved. They can be placed in a MACLIB and called when needed.

For more information contact the author at the following address:

Karen Crandall  
Eastman Kodak Company  
Research Labs, Building 83, Floor 11  
Kodak Park  
1669 Lake Avenue  
Rochester, NY 14650

SAS and SAS/AF are registered trademarks of SAS Institute, Inc., Cary NC USA.