

```

%macro mu_help_debug / store SOURCE des='V1.0.0.1' ;
%*****
FILENAME: mu_help_debug.sas
DEVELOPER: (b) (6)
PLATFORM: PC SAS 9.1.3, 9.2
DATE: 20120210
PURPOSE: This macro will determine if there are _default_ values for
         HELP and DEBUG that have been assigned elsewhere. If there are, and
         the the local parameters of HELP and DEBUG from where the macro this
         is called are blank, then the local macro will inherit the _default_
         values

MODIFIED BY:          DATE:
%*****;

%** is _default_debug an assigned macro var?;
proc sql noprint;
    select name from dictionary.macros
    where upcase(name) = '_DEFAULT_DEBUG';
    %let exist_debug =&sqllobs;
quit;

%** if _default_debug is assigned then inherit value.
    if not, then set debug to NO;
%if &exist_debug eq 1 and &debug = %str() %then %do;
    %let debug = &_default_debug;
%end;
%else %if &exist_debug eq 0 and &debug = %str() %then %do;
    %let debug = NO;
%end;

%** is _default_help an assigned macro var?;
proc sql noprint;
    select name from dictionary.macros
    where upcase(name) = '_DEFAULT_HELP';
    %let exist_help =&sqllobs;
quit;

%** if _default_help is assigned then inherit value.
    if not, then set help to NO;
%if &exist_help eq 1 and &help = %str() %then %do;
    %let help = &_default_help;
%end;
%else %if &exist_help eq 0 and &help = %str() %then %do;
    %let help = NO;
%end;

%let debug = %upcase(%substr(&debug, 1, 1)) ;
%if &debug = 1 %then %let debug = Y;

```

```
%if &debug = 0 %then %let debug = N;  
%let help = %upcase(%substr(&help, 1, 1)) ;  
%if &help = 1 %then %let help = Y;  
%if &help = 0 %then %let help = N;  
  
%mend mu_help_debug;
```