

```

%macro md_default_check(
    mparm_name      =
    ,md_default_mvar =
    ,md_default_value =
    ,help=N
    ,debug=N
) / store source des='V1.0.0.3' ;
%*****
*****
* FILENAME:      MD_DEFAULT_CHECK.SAS
* DEVELOPER:      (b) (6)
* PLATFORM:      SAS 9.1.3, 9.2 on PC
*
* MACROS USED: MU_CHECKRC
* ASSUMPTIONS: Called from inside another macro that has one or more parameters
*
* DESCRIPTION:
*   If MPARM_NAME has been entered and if the macro variable it represents has a
null value.
*   If both are true, the macro will look for a default value for this macro
variable.
*
*   If a value has been passed via MD_DEFAULT_MVAR, the macro looks for an existing
macro variable with this name. .
*   The name passed into this parameter must NOT include the ampersand (&)
*   If the macro variable exists, the value of the macro variable is passed into a
the calling macro variable
*   entered via the MPARM_NAME parameter.
*   If the macro variable does not exist, the program will use a secondary default
value, if it was provided
*   via macro parameter MD_DEFAULT_VALUE.
*   Otherwise it will set the &MDPARM_NAME macro variable to a null value.
*
* USAGE NOTES:
*   Example 1: If the macro variable _default_bign_in_data exists, IN_DATA will be
set to the value of &_default_bign_in_data.
*               If it does not exist, IN_DATA will be set to a null value.
* %md_default_check(
*     mparm_name      = in_data
*     , md_default_mvar = _default_bign_in_data
*     , md_default_value =
*     )
*
*   Example 2: If the macro variable _default_prcode_split exists, PRCODE_SPLIT
will be set to the value of &_default_prcode_split.
*               If it does not exist, PRCODE_SPLIT will be set to $.
*
* %md_default_check(

```

```

*      mparm_name      =prcode_split
*      , md_default_mvar =
*      , md_default_value= $
*      )
*
* PARAMETERS:
* mparm_name          = name of a macro parameter in the macro executing this macro
(REQUIRED)
*
* md_default_mvar     = name of a macro variable for which this macro will determine
if it exists or not.
*
*                     If no value is entered, program will look for macro named
_DEFAULT_whatever, where whatever is the value
*                     passed into MPARAM_NAME. (See Example 2 above)
*                     If the macro variables exists, the value of this variable will
be used to set the calling macro parameter
*                     entered in MPARAM_NAME (OPTIONAL)
*
* md_default_value = value to be used if the name of the macro variable entered in
MD_DEFAULT_MVAR does not exist (OPTIONAL)
*****
* DATE      INITIALS    MODIFICATION DESCRIPTION
*****
*
*****
* © 2011 Pharmaceutical Product Development, Inc.
* All Rights Reserved.
*****
*****;
    %local df_mvar;

    %* -- Value must be passed to parameter mparm_name for any processing to be
done --;
    %if %bquote(&mparm_name) ne %str( ) and %bquote(&&mparm_name) = %str( ) %then
%do;

        %* -- If no value is passed to parameter MD_DEFAULT_MVAR, create a default
based upon the parameter name with _DEFAULT as a prefix --;
        %if %bquote(&md_default_mvar) = %str( ) %then %let df_mvar =
_default_&mparm_name;
        %else                                     %let df_mvar =
&md_default_mvar;

        %* -- Macro variable does not exist
-----;
        %if %symexist(&df_mvar) = 0 %then %do;

            %* -- Check if secondary default value was provided -----;
            %if %bquote(&md_default_value) = %str( ) %then %let &mparm_name=; %*
-- No, Set the macro parameter to a null value--;

```

```

        %else %do;    %* Yes, set to secondary default value and print message
in log;
        %let &mparm_name = %str(&md_default_value);

        %if &help = Y %then %do;
            %MU_CHECKRC(CONDITION = 1=1
                        ,RC      = 0
                        ,rcprefix = &SYSMACRONAME
                        ,MSG1     = The Macro Parameter [&mParm_name] was set
to an internal DEFAULT value ==> %str(&md_default_value)
                        ) *;

        %end;
    %end;
%end;

    %* -- Macro variable exists, set value of macro to macro parameter and
enter a message in the log --;
    %else %do;
        %let &mparm_name= &&&df_mvar;

        %if &help = Y %then %do;
            %MU_CHECKRC(CONDITION = 1=1
                        ,RC      = 0
                        ,rcprefix = &SYSMACRONAME
                        ,MSG1     = The Macro Parameter [&mParm_name] was set to
the DEFAULT (&df_mvar) value ==> &&&df_mvar
                        ) *;

        %end;

    %end;
%end;

%mend md_default_check;

```