

```

%macro mw_sentence_case(termfile =
                        ,indata =
                                ,outdata=&indata.
                        ,var =
                                ,rsvdterm =
                                ,listchar = @
                                ,tcase = NO
                                ,debug = YES
                        ,help = YES
                        ) / store source des='V1.0.0.3';

%*****
%*****;
%*      MACRO NAME:  mw_sentence_case

;
%*      SAS VERSION:  SAS V9.3, SAS V9.4 on SAS grid (Windows 10)

;
%*      PURPOSE:  Parent macro that calls MW_CREATE_TERM_LIST, MW_MARK_TERMS,
and MW_ANALYZE_SENTENCE.
%*      USAGE NOTES:  A reserved term list must be present.
;
%*

;
%*  INPUT PARAMETERS:

;
%*      TERMFILE = An absolute path to the reserved term list.
;
%*      (Required, (NO DEFAULT)

;
%*      INDATA  = Data set with string variable present.

;
%*      OUTDATA = The resulting data set post sentence casing. (default value
is the input data set).
%*      VAR = The string variable name in INDATA to sentence case.

;
%*      RSVDTerm = Macro variable containing list of reserved terms.

;
%*      LISTCHAR = The special character that will be used to identify reserved
terms. (Default value is @)
;
%*      TCASE = If set to Y(es), then words with at least four bytes will be

```

```
proper case. (Default value is NO) ;
%*      HELP = set to Y(es) to output helpful hints to the log (Default
= YES)
```

```

%*          DEBUG = set to Y(es) to save all intermediate datasets and to
turn on mprint. Set to NO to delete all intermediate datasets and to not      ;
%*          turn on mprint (maintains original
option setting) (Default = YES)

```

```
%*          OUTPUT FILES:  NONE
```

%; AUTHOR: (b) (6)

```
%*
DATE_CREATED: 31Dec2015
```

```
%*      MODIFICATION LOG: ;
```

```

%*****
*****.

```

%*	DATE	INITIALS	MODIFICATION	DESCRIPTION

```
%* 14FEB2019      (b) (6)      Added Return Codes, debug macros, and parameter
checks.           ;
```

```
%* 21MAR2019 (b) (6) Add parameter and test for RSVDTerm, macro
variable for _help and _debug, update header.
```

```

;
%* 26MAR2019 (b) (6) Add code to re-set RSVDTERM parameter value.

```

08APR2019	(b) (6)	Add LISTCHAR parameter.
-----------	---------	-------------------------

```
%* 31MAY2019      (b) (6)      Add TCASE parameter.
```

```

%* 19JUN2019 (b) (6) Modify TCASE so that it can take a value of Y(es)
or N(o).

```

%* 020CT2019 (b) (6) Per PM, list version as Version 1.0.0.0. Remove
references to Windows 7. ;

```

%*****
%***** ;
%*****
%*****;
%*   (c) 2019 PPD

;

%*   All Rights Reserved.

;

%*****
%*****;

%PUT ----- ;
%PUT INFO: (&SYSMACRONAME) ;
%PUT INFO: Version 1.0.0.3 ;
%PUT START;
%PUT ----- ;

%*****;
%** Parameter checks.          **;
%*****;

%global &sysmacroname._rc;
%let &sysmacroname._rc = 0;

%** RETURN CODES
0 = ran without error
1 = rsvdterm parameter is null and will be set to the default value of rsvdtrms
2 = rsvdterm parameter has a value of rsvdterm and will be re-set to rsvdtrms
3 = Macro is being executed by encoding other than WLATIN1
;

%*****;
%** Check if ARTZ defaults has DEBUG and HELP. **;
%*****;

%global _debug _help _listchar _tcase;

%let _debug = &debug;
%let _help = &help;
%let _listchar = &listchar;

%if %upcase(&tcase) = Y or %upcase(&tcase) = YES %then %do;
    %let _tcase = YES;
%end;
%else %if %upcase(&tcase) = N or %upcase(&tcase) = NO %then %do;
    %let _tcase = NO;
%end;

```

```

%mu_help_debug;

%*****;
%** Get list of datasets in the work library.  **;
%*****;

%md_workinfo(
    debug          = &debug
    ,_workdata      = WORK_DATASETS_DATA
    ,_workview      = WORK_DATASETS_VIEW
    ,_mprinttoggle = mprint_setting
) *;

%*****;
%** Check value of RSVDTERM parameter.      **;
%*****;

%if %sysevalf(%superq(rsvdterm)=,boolean) %then %do;

    %let rsvdterm = rsvdtrms;

    %put
    -----;
    %put ALERT_I: &SYSMACRONAME - RSVDTERM parameter is null.;
    %put ALERT_I: &SYSMACRONAME - RSVDTERM will be set to the;
    %put ALERT_I: &SYSMACRONAME - name, RSVDTRMS;
    %let &sysmacroname._rc = 1;
    %put INFO: &sysmacroname._RC = &&sysmacroname._RC;
    %put
    -----;
    -----;

%end;

%else %if %upcase(&rsvdterm) = RSVDTERM %then %do;

    %let rsvdterm = rsvdtrms;

    %put
    -----;
    -----;
    %put ALERT_I: &SYSMACRONAME - RSVDTERM parameter has a value;
    %put ALERT_I: &SYSMACRONAME - of RSVDTERM and will be re-set;
    %put ALERT_I: &SYSMACRONAME - to a value of RSVDTRMS;
    %let &sysmacroname._rc = 2;
    %put INFO: &sysmacroname._RC = &&sysmacroname._RC;
    %put
    -----;
    -----;

```

```

%end;

%*****;
%** Check if REQUIRED parameters are populated **;
%*****;

%mu_check_req_parameters(
    parameters_to_check = termfile indata var
    ,help=no
);

%*****;
%** Check encoding Session **;
%*****;

%let session_encoding=%upcase(%sysfunc(getoption(encoding)));

    %mu_checkrc(condition = "&session_encoding." ^= "WLATIN1"
        ,setabort = Y
        ,rc = 3
        ,rcprefix = MW_SENTENCE_CASE
        ,msg1 = Please note macro has not been fully validated
within &session_encoding.
        ,msg2 = INFO: MW_SENTENCE_CASE_RC = 3
    );

%*****;
%** Start macro. **;
%*****;

%global &rsvdterm;

%mw_create_term_list(fileloc = &termfile.
    ,outvar = &rsvdterm.
    );

%mw_mark_terms(indata = &indata.
    ,outdata = &indata._1
        ,var = &var.
        ,listvar = &rsvdterm.
    );

%mw_analyze_sentence(indata = &indata._1
    ,outdata = &outdata.
        ,var = &var.
    );

%mend mw_sentence_case;

```