

Macros

Can I export a dataset to a CSV file?

There are several ways to do this. This utility macro allows you to specify various parameters, and validates these before generating the output file.

```
%macro ds2csv ( csvfile =
                ,csvfref =
                ,openmode = replace
                ,colhead = y
                ,data      = &syslast
                ,formats   = y
                ,labels    = y
                ,sepchar   =
                ,var       =
                ,where     =
                ) ;

  options nomprint nomlogic nosymbolgen nonotes ;
  options noquotelenmax ; /*** Suppress WARNING message about quoted s
tring longer than 262 chars ***/
  %local i usecolhead useformats uselabels s itthey isare ;
  %*** Validate Parameters *** ;
  %if &csvfile ne and &csvfref ne %then
  %do ;
    %put WARNING: CSVFILE and CSVFREF have both been specified. CSVFRE
F parameter will be ignored. ;
    %let csvfref = ;
  %end ;

  %if &csvfref ne %then
  %do ;
    proc sql noprint ;
      select fileref into :extfiles separated by ' '
      from dictionary.extfiles
      ;
    quit ;
    %if %sysfunc(find(&extfiles,&csvfref,i)) = 0 %then
    %do ;
      %put ERROR: The FILEREF %upcase(&csvfref) has not been assigned.
No records will be written. ;
      %goto endmac ;
    %end ;
  %end ;

  %let openmode = %lowercase(&openmode) ;
  %if &openmode ne replace and &openmode ne append %then
  %do ;
    %put ERROR: Invalid OPENMODE parameter specified. Valid options
are REPLACE, APPEND. CSV file will not be created. ;
    %goto endmac ;
  %end ;
```

Macros

```
%let colhead = %lowercase(%substr(&colhead,1,1)) ;
%if &colhead ne y and &colhead ne n %then
%do ;
    %put ERROR: Invalid COLHEAD parameter specified. Valid options are Y, N. CSV file will not be created. ;
    %goto endmac ;
%end ;
    %else %if &colhead = y %then %let usecolhead = yes ;
    %else %let usecolhead = no ;

%let formats = %lowercase(%substr(&formats,1,1)) ;
%if &formats ne y and &formats ne n %then
%do ;
    %put ERROR: Invalid FORMATS parameter specified. Valid options are Y, N. CSV file will not be created. ;
    %goto endmac ;
%end ;
    %else %if &formats = n %then %let useformats = noformats ;

%let labels = %lowercase(%substr(&labels,1,1)) ;
%if &labels ne y and &labels ne n %then
%do ;
    %put ERROR: Invalid LABELS parameter specified. Valid options are Y, N. CSV file will not be created. ;
    %goto endmac ;
%end ;
    %else %if &labels = y %then %let uselabels = label ;

%if &data = _NULL_ %then
%do ;
    %put ERROR: No dataset specified. CSV file will not be created. ;
    %goto endmac ;
%end ;

proc sql noprint ;
    select distinct libname into :all_libs separated by ' '
    from dictionary.libnames
    ;
quit ;
%if %sysfunc(find(&data,.)) > 0 %then
%do ;
    %let lib = %upcase(%scan(&data,1,.)) ;
    %let dsn = %upcase(%scan(&data,2,.)) ;

    %if %sysfunc(find(&all_libs,&lib,i)) = 0 %then
    %do ;
        %put ERROR: The LIBREF &lib has not been assigned. CSV file will not be created. ;
```

Macros

```
%goto endmac ;
%end ;

proc sql noprint ;
    select memname into :all_dsns separated by ' '
    from dictionary.tables
    ;
quit ;
%if %sysfunc(find(&all_dsns,&dsn,i)) = 0 %then
%do ;
    %put ERROR: The dataset &dsn is not present in the library &lib.
    CSV file will not be created. ;
    %goto endmac ;
%end ;
%end ;
%else
%do ;
    %if %sysfunc(find(&all_libs,%str( USER ),i)) > 0 %then %let lib =
USER ;
    %else %let lib = WORK ;
    %let dsn = %upcase(&data) ;
%end ;

%if &csvfile = and &csvfref = %then
%do ;
    %let csvfile = %lowcase(&dsn).csv ;
%end ;

%let hexvals = ;
data _null_ ;
    do i = 0 to 255 ;
        call symputx('hexvals', catx(' ',symget('hexvals'),put(i,hex2.))
    ) ;
    end ;
run ;
%let sepchar = %upcase(&sepchar) ;
%if %sysfunc(find(&hexvals,&sepchar,i)) = 0 %then
%do ;
    %put WARNING: The SEPCHAR parameter &sepchar is not a valid hexade
cimal character. A comma-separated file will be generated. ;
    %if &sysscp = WIN %then %let sepchar = 2C ;
    %else %let sepchar = 6B ;
%end ;
%let act_sep = %sysfunc(inputc(&sepchar,$hex2.)) ;

%let invalid_vars = ;
%let valid_vars = ;
%let comma_vars = ;
%if &var ne %then
```

Macros

```
%do ;
  %let varcount = %eval(%sysfunc(count(&var,%str( ))) + 1) ;
  proc sql noprint ;
    select name into :all_vars separated by ' '
    from dictionary.columns
    where libname = "&lib"
    and memname = "&dsn"
    ;
  quit ;
  %do i = 1 %to &varcount ;
    %let ivar = %scan(&var,&i) ;
    %if %sysfunc(find(&all_vars,&ivar,i)) > 0 %then
      %do ;
        %let valid_vars = &valid_vars &ivar ;
        %let comma_vars = &comma_vars &ivar %sysfunc(ifc(&i ne &varcount,%str(&act_sep),%str())) ;
      %end ;
      %else %let invalid_vars = &invalid_vars &ivar ;
    %end ;
    %if &invalid_vars ne %then
      %do ;
        %if %sysfunc(countc(&invalid_vars,%str( ))) = 0 %then
          %do ;
            %let s = ;
            %let itthey = It ;
            %let isare = is ;
          %end ;
          %else
            %do ;
              %let s = s ;
              %let itthey = They ;
              %let isare = are ;
            %end ;
          %put WARNING: The variable&s: %upcase(&invalid_vars) &isare not
present in the dataset &lib..&dsn.. &itthey will be omitted. ;
          %put WARNING: Only the variables: %upcase(&valid_vars) will be w
ritten to the file. ;
        %end ;
      %end ;
    %else
      %do ;
        proc sql noprint ;
          select name into :comma_vars separated by "&act_sep "
          from dictionary.columns
          where libname = "&lib"
          and memname = "&dsn"
          ;
        quit ;
        %let valid_vars = %sysfunc(compress(%bquote(&comma_vars),%str(%bqu
```

Macros

```
ote(&act_sep)))) ;
%end ;

proc sql noprint ;
select sum(length) into :totlen
from dictionary.columns
where libname = "&lib"
    and memname = "&dsn"
    ;
select nobs into :totobs
from dictionary.tables
where libname = "&lib"
    and memname = "&dsn"
    ;
quit ;
%if %eval(%sysfunc(length(%bquote(&comma_vars)))+2>&totlen) %then %let totlen = %eval(%sysfunc(length(%bquote(&comma_vars)))+2) ;
%if &colhead = y %then %let totobs = %eval(&totobs + 1) ;

/****
data _null_ ;
    set &lib..&dsn ;
    file
    %if &csvfile ne %then "&csvfile" ;
    %else &csvfref ;
    dsd dlm = "&sepchar" x lrecl = &totlen &repmod ;
    %if &colhead = y %then
    %do ;
        if _n_ = 1 then put "&comma_vars" ;
    %end ;
    put &valid_vars ;
run ;
****/

proc export data      = &lib..&dsn (keep = &valid_vars
                                %if &where ne %then
                                where = (&where)
                                ;
                                )
    outfile = %if &csvfile ne %then "&csvfile" ;
              %else &csvfref ;
    dbms     = dlm
    &useformats
    &uselabels
    &openmode
    ;

delimiter = "&sepchar" x ;
putnames   = &usecolhead ;
```

Macros

```
run ;

options notes ;
%put NOTE: &totobs records have been written to the file: &csvfref&
svfile.. ;
%put NOTE: The record separator is &sepchar:      &act_sep ;
%endmac:
%mend ds2csv ;
```

Unique solution ID: #1055

Author: Alan D Rudland

Last update: 2020-07-29 11:24