

m_utl_ds2csv.sas File Reference

Utilities

Utility macro to export a SAS dataset into a CSV file

Description

This program converts a SAS dataset into a comma separated values (CSV) file. This is a method to transport datasets or tables independently from one host or system to another.

Note

The following word(s) are not allowed as part of the input SAS dataset or table name: JOIN.

Authors

Paul Alexander Canals y Trocha (paul.canals@gmail.com)

Date

2022-11-08 00:00:00

Version

22.1.11

Link

<https://github.com/paul-canals/toolbox>

Parameters

Input	help	Parameter, if set (or ?) to print the Help information in the log. In all other cases this parameter should be left out from the macro call.
Input	base_ds	Full LIBNAME.TABLENAME name of the SAS dataset. The default value is: <code>_NONE_</code> .
Input	csv_file	Specifies the name of the CSV file and path where the formatted output is to be written. If the file that you specify does not exist, then it is created for you. The default value is: <code>_NONE_</code> .
Input	savemode	Indicator [REPLACE APPEND] to decide whether the new CSV output overwrites the information that is currently in the specified file or if the new output is appended to the end of the existing file. The default value is: REPLACE.
Input	runmode	Indicator [B S] specify whether the macro is running in batch or server mode. Batch mode means that the macro is to be included into a SAS program. The default value is: B.
Input	colhead	Boolean [Y N] parameter to indicate whether to include column headings in the CSV file. The column headings that are used depend on the setting of the LABELS argument. Column headings are included as the first record of the CSV file. The default value is: Y.
Input	formats	Boolean [Y N] parameter to indicate whether to apply the dataset defined variable formats to the values in the CSV file. The formats must be stored in the data set in order for them to be applied. The default value is: N.
Input	labels	Boolean [Y N] parameter to indicate whether to use the variable labels that are defined in the dataset as your column headings. If a variable does not have a SAS label, then use the name of the variable instead. The default value is: N.
Input	sepchar	Specifies the two-character hexadecimal code for the separator character. The default value is 2C which represents a comma. Other possible values are for example 3B which represents a semicolon, or 09 which represents a tab as the separator character. The default value is: 2C.
Input	password	Specifies the that is needed to access a password-protected dataset. This argument is required if the dataset has READ or PW password. There is no need to specify this argument if the dataset has only WRITE or ALTER passwords set.
Input	varlist	Specifies the variables that are to be included in the file and the order in which they should be included. To include all of the variables in the data set, do not specify this argument. If you want to include only a subset of all the variables, then list each variable name and use single blank spaces to separate the variables. Do not use a comma in the list of variable names.
Input	where	Specifies a valid WHERE clause that selects observations from the SAS dataset. Using this argument subsets your data based on the criteria that you supply for where-expression.
Input	debug	Boolean [Y N] parameter to provide verbose mode information. The default value is: N.

Returns

- Output CSV file.

Calls

- [m_util_create_dir.sas](#)
- [m_util_print_message.sas](#)
- [m_util_print_mtrace.sas](#)

Usage

Example 1: Show help information:

```
%m_utl_ds2csv(?)
```

Example 2: Create a CSV file from SASHELP.class:

```
%m_utl_ds2csv(  
    base_ds = SASHELP.class  
    , csv_file = %str(%sysfunc(getoption(WORK))/class.csv)  
    , labels = Y  
    , where = %str(Sex = 'M')  
    , debug = Y  
    );  
  
filename class "%sysfunc(getoption(WORK))/class.csv";  
  
data _null_;  
    infile class;  
    input;  
    put _infile_;  
run;
```

Copyright

Copyright 2008-2022 Paul Alexander Canals y Trocha.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.