

m_utl_copy_file.sas File Reference

Utilities

Utility macro to copy an external file to another location

Description

This program uses a byte-for-byte method or uses chunks to copy the input file onto another location. Therefore there are no restrictions to copying external files, however SAS file types should be copied by the m_utl_copy_sas_file.sas macro routine to avoid issues. The chunksize option that is used in this macro is based on the binaryfilecopy by Bruno Mueller (bruno.mueller@sas.com).

Note

This program is able to work in system environments where x-command or unix pipes are not allowed or cannot be used.

Autors

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

Date

2021-09-27 00:00:00

Version

21.1.09

Link

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

Parameters

Input	help	Parameter, if set (Help or ?) to print the Help information in the log. In all other cases this parameter should be left out from the macro call.
Input	infile	Full qualified path and file name of the file to be copied from. The default value is: <code>_NONE_</code> .
Input	outfile	Full qualified path and file name of the copy. If the output folder structure does not exist yet, it will be created during runtime.
Input	chunksize	Optional. Parameter to specify the number of bytes to be processed in one operation. This will affect the time it takes to copy a file. Smaller values mean longer processing time to copy a file. Valid byte lengths for CHUNKSIZE are 1, 512, 1024, 4096, 8192, 16384 and 32767. The default value for CHUNKSIZE is: 8192.
Input	overwrite	Boolean [Y N] parameter to decide if an existing copy file is to be overwritten. The default value is: Y.
Input	show_err	Boolean [Y N] parameter to show or hide warnings or errors in the log. The default value is: Y.
Input	debug	Boolean [Y N] parameter to provide verbose mode information. The default value is: N.

Returns

- The copied file as mentioned in OUTFILE.

Calls

- [m_utl_create_dir.sas](#)
- [m_utl_print_message.sas](#)
- [m_utl_print_mtrace.sas](#)

Usage

Example 1: Show help information:

```
%m_utl_copy_file(?)
```

Example 2: Copy an external file to another directory (byte-for-byte):

```
%m_utl_copy_file(  
  infile    = %sysget(SASROOT)/core/sashelp/class.sas7bdat  
  , outfile  = %sysfunc(getoption(WORK))/backup/class.sas7bdat  
  , overwrite = Y  
  , debug    = Y  
  );  
  
libname OUT "%sysfunc(getoption(WORK))/backup";  
  
proc print data=OUT.class label;  
run;  
  
libname OUT clear;
```

Example 3: Copy an external file to another directory (using chunks):

```
%m_utl_copy_file(  
  infile    = %sysget(SASROOT)/core/sashelp/class.sas7bdat  
  , outfile  = %sysfunc(getoption(WORK))/backup/class.sas7bdat  
  , chunksize = 8192  
  , overwrite = Y  
  , debug    = Y  
  );  
  
libname OUT "%sysfunc(getoption(WORK))/backup";  
  
proc print data=OUT.class label;  
run;  
  
libname OUT clear;
```

Example 4: Avoid copying to an existing external file (throwing an error):

```
%m_utl_copy_file(  
  infile    = %sysget(SASROOT)/core/sashelp/class.sas7bdat  
  , outfile  = %sysfunc(getoption(WORK))/backup/class.sas7bdat  
  , chunksize = 8192  
  , overwrite = N  
  , show_err  = Y  
  , debug    = Y  
  );  
  
libname OUT "%sysfunc(getoption(WORK))/backup";  
  
proc print data=OUT.class label;  
run;  
  
libname OUT clear;
```

Example 5: Avoid copying to an existing external file (no error thrown):

```
%m_utl_copy_file(  
  infile      = %sysget(SASROOT)/core/sashelp/class.sas7bdat  
  , outfile   = %sysfunc(getoption(WORK))/backup/class.sas7bdat  
  , chunksize = 8192  
  , overwrite = N  
  , debug     = Y  
  );  
  
libname OUT "%sysfunc(getoption(WORK))/backup";  
  
proc print data=OUT.class label;  
run;  
  
libname OUT clear;
```

Copyright

Copyright 2008-2021 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/>>.