

m_utl_print_mtrace.sas File Reference

Utilities

Utility macro to print a list of all nested executing macros

Description

The macro is used to obtain a list of all nested macros that are being executed from a starting point upto a given nested macro level end point, and print this resulting macro trace list as an information statement in the SAS system log:

```
MTRACE(macro_1.macro_2. .. macro_N);
```

Note

This macro only executes on systems where SAS version 9.3 or higher is installed, since it relies on the SYSMEXECNAME and SYSMEXECDEPTH functions which were introduced for SAS 9.3.

Autors

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

Date

2020-09-07 00:00:00

Version

20.1.09

Link

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

Parameters

Input	help	Parameter, if set (Help or ?) to print the Help information in the log. If set to Start or END in combination with the correct SYSMECDEPTH value, an additional MTRACE entry is written to the log destination. In all other cases this parameter should be left out from the macro call.
Input	point	Specifies an integer value that represents the starting point of the macro trace list. If set to zero, the macro execution depth starts with Open Code. The default value for POINT is: 1.
Input	depth	Specifies an integer value that represents the end point of the macro trace list. If set to blank or zero, the macro trace also includes this macro. The default value for DEPTH is: -1.
Input	print	Boolean [Y N] parameter to generate the output by a SAS proc report step with style HtmlBlue. The default value for PRINT is: N.
Input	debug	Boolean [Y N] parameter to provide verbose mode information. The default value is: N.

Returns

- List of nested macros that are being executed to current level.

Calls

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

Usage

Example 1: Show help information:

```
%m_utl_print_mtrace(?)
```

Example 2: Print the macro execution trace with default values:

```
%macro main_macro;  
    %macro sub_macro;  
        %m_utl_print_mtrace(print=Y);  
    %mend sub_macro;  
    %sub_macro;  
%mend main_macro;  
%main_macro;
```

Example 3: Print the macro execution trace including all macros:

```
%macro main_macro;  
    %m_utl_print_mtrace(  
        point = 0  
        , depth = 0  
        , print = Y  
    );  
    %macro sub_macro;  
        %m_utl_print_mtrace(  
            point = 0  
            , depth = 0  
            , print = Y  
        );  
    %mend sub_macro;  
    %sub_macro;  
%mend main_macro;  
%main_macro;
```

Example 4: Print the macro execution trace with start/end values:

```
%macro main_macro;  
    %m_utl_print_mtrace(start)  
    %macro sub_macro;  
        %m_utl_print_mtrace(start)  
        %put this is the submacro;  
        %m_utl_print_mtrace(end)  
    %mend sub_macro;  
    %sub_macro;  
    %m_utl_print_mtrace(end)  
%mend main_macro;  
%main_macro;
```

Copyright

Copyright 2008-2020 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/>>.