m_utl_ds2json.sas File Reference

Utilities

Utility macro to export a SAS dataset or table into a JSON file

---

Description

This program converts a SAS dataset or database table into a javascript object notation (JSON) file.
The program can be run in SAS environments prior to SAS version 9.4 M3. The following JSON data types are implemented:
Number: a signed decimal number.
String: a sequence of zero or more Unicode characters.
Boolean: either of the values true or false.
Array: an ordered list of zero or more values.
Object: an unordered collection of key-value pairs
Null: an empty value, using the word null.

Note

An array is only created up to one level deep, however with no restriction on the number of key-value pairs. An _ordered-list_ type array is not supported. A _NULL_ value is supported as a missing value representation for a character type value only.

Autors

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	Specifies the full LIBNAME.TABLENAME name of the input SAS dataset or database table. The default value is: _NONE_.
Input	json_file	Specifies the name of the JSON 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: _NONE_.
Input	runmode	Indicator [C|S] specify whether the macro is running in (C)ustom or (S)ystem mode. System mode means that the macro uses the SAS internal PROC JSON procedure which is available sincs SAS 9.4 M3. The default value is: S.
Input	sastags	Boolean [Y|N] parameter to determine whether to include or suppress SAS metadata at the top of the JSON output file. The metadata consists of the SAS export version, exported SAS dataset name, and any non-default option specification. The default value for SASTAGS is: N.
Input	keys	RUNMODE=S only. Boolean [Y|N] parameter to determine whether to include or suppress SAS variable names in the JSON output file. The default value for KEYS is: Y.
Input	scan	RUNMODE=S only. Boolean [Y|N] parameter to determine whether the PROC JSON scans and encodes input strings to ensure that only characters that are acceptable, are exported to the JSON output file. The default value for SCAN is: Y.
Input	pretty	Boolean [Y|N] parameter to specify how to format the JSON output structure. The argument PRETTY=Y creates a more human-readable format that uses indention to illustrate the JSON structure. The argument PRETTY=N writes the output in a single line. The default value for PRETTY is: Y.
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 JSON file.

Calls

- m_utl_create_dir.sas
- m_utl_print_message.sas
- m_utl_print_mtrace.sas

Usage

Example 1: Show help information:

```
%m_utl_ds2json(?)
```

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

```
%m_utl_ds2json(
   base_ds   = SASHELP.class
 , json_file = %str(%sysfunc(getoption(WORK))/class_f.json)
 , runmode   = S
 , sastags   = Y
 , where     = %str(Sex = 'F')
 , debug     = Y
   );

filename class "%sysfunc(getoption(WORK))/class_f.json";

data _null_;
   infile class;
   input;
   put _infile_;
run;
```

Example 3: Create a JSON file from SASHELP.class:

```
%m_utl_ds2json(
   base_ds   = SASHELP.class
 , json_file = %str(%sysfunc(getoption(WORK))/class_m.json)
 , runmode   = C
 , pretty    = N
 , sastags   = Y
 , where     = %str(Sex = 'M')
 , debug     = Y
   );

filename class "%sysfunc(getoption(WORK))/class_m.json";

data _null_;
   infile class;
   input;
   put _infile_;
run;
```

Example 4: Create a JSON file with missing values (NULL):

```
data WORK.class;
   set SASHELP.class;
   if name in ('John','James') then do;
      sex = '';
      weight = .;
   end;
run;

%m_utl_ds2json(
   base_ds   = WORK.class
 , json_file = %str(%sysfunc(getoption(WORK))/class.json)
 , runmode   = C
 , pretty    = Y
 , sastags   = Y
 , debug     = Y
   );

filename class "%sysfunc(getoption(WORK))/class.json";

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/>.