Pass SAS Dataset Through Multiple Stored Processes Using SAS Sessions

Presented By:

Christopher Lampron

Contact Information:

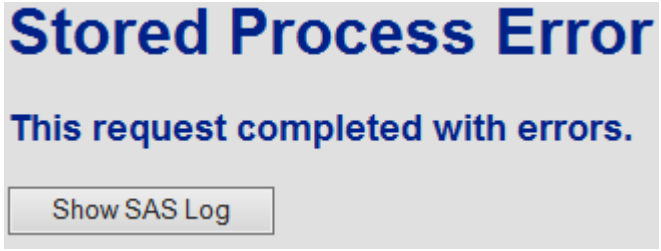
Christopher.Lampron@Siemens.com

Uploading and Printing Dataset Without SAS Sessions

- The first stored process allows the user to select a file to upload
  - Browse and upload button using HTML forms
- The second stored process uploads the file and converts it to a SAS dataset
  - Proc Import
  - Auto submit HTML code to next stored process to print data set.
- The third stored process prints the dataset
  - Proc print
  - Print the data set to HTML output.

Result of Proc Print Without SAS Sessions

- When going from the second to third stored process, the SAS dataset (Work.Excel2) is not carried over
- The result is a stored process error when the stored process tries to print the dataset.




```
30          proc print data=work.excel2 noobs;
ERROR: File WORK.EXCEL2.DATA does not exist.
31          run;
```

Creating SAS Session

```
%let rc=%sysfunc(stpsrv_session(create));
```

- Syntax used In macro
- A session is the data that is saved from one stored process to the next
- It is scoped so that all users have independent sessions.
- Session creates a session library SAVE.
- Session creates the following macro variables:
  - _SESSIONID, _THISSESSION
- Global variables are saved in the session with the prefix ‘SAVE_’
- The SAS Session variables can be used to create URLs that can be used to chain stored processes together

Creating HTML Form To Upload Files

```
%let rc=%sysfunc(stpsrv_session(create));
```

```
data form;
  format infile $char256.;
  input;
  infile=resolve(_infile_);
  cards;
<HTML>
<BODY>

<FORM ACTION="&_THISSESSION" method="post" enctype="multipart/form-data">
<INPUT TYPE="HIDDEN" NAME="_program" VALUE="Stored Process Path/Stored Process Name">
<INPUT TYPE="HIDDEN" NAME="submit" VALUE="submitted">
<INPUT TYPE="HIDDEN" NAME="Attach1" VALUE="&Attach1">
<INPUT TYPE="file" NAME="Attach1" />
<td><INPUT TYPE="submit" value="Upload">

</FORM>
</BODY>
</HTML>
;
run;
```

```
data _null_;
  file _webout;
  set form;
  put infile;
run;
```

- &_THISSESSION is the reserved macro variable from the SAS session
  - /SASStoredProcess/do?_sessionid=xxxxxx
- Running this HTML form creates additional reserved macros that will be used to upload the data
  - Stored process described in the HTML form is executed

Importing The Uploaded File

```
proc import out=save.excel2
    file="&_webin_fileref"
    dbms=xlsx
    replace;
    getnames=yes;
run;
```

- The second stored process can be used to upload the Excel file
- Use PROC IMPORT to import the uploaded Excel file
- Use the SAVE session library to create the SAS dataset
- The file reference is the reserved macro &_webin_fileref
  - Created in the previous stored process
- Running the above code will create a SAS dataset called ‘Save.Excel2’

Chaining Another Stored Process

```
data _null_;
    file _webout;
    put '<html>';
    put '<body>';
    put '<form name="myform"
    action="https://servername:port/'"&_THISSESSION"'&_program=/Stored Process Path/Stored Process Name"
    method="post">';
    put '</form>';
    put '<script type="text/javascript">document.myform.submit();</script>';
    put '</body>';
    put '</html>';
run;
```

- Within the same stored process that the uploaded file was imported to a SAS datset
- HTML code uses a FORM ACTION to chain to another stored process
  - The SAS Session variable _THISSESSION is used to identify that the next stored process is part of the session
- The Javascript code after </FORM> is used to auto-submit the form
  - No window will be visible for the user
  - The uploaded file will be imported and then will move onto the next process

Printing Uploaded File

```
goptions reset=all;
title "Output of Uploaded Excel File";
proc print data=save.excel2 noobs;
run;
```

- The third chained stored process prints the file that was uploaded in the first stored process
- Using PROC PRINT, the uploaded file is printed
- The dataset created from the PROC IMPORT on the previous stored process is called in the Print PROC using the SAS session library, SAVE.

Output

Output of Uploaded Excel File

Make	MPG	WEIGHT	PRICE
AMC Concord	22	2930	4099
AMC Pacer	17	3350	4749
AMC Spirit	22	2640	3799
Buick Century	20	3250	4816
Buick Electra	15	4080	7827

- The file was selected from the navigation window in the first stored process
- The file was imported as a SAS dataset in the second stored process
- The SAS dataset was printed in the third stored process

References

- **Aanderud, T., & Hall, A. (2012).** ***The 50 keys to learning SAS stored processes*****. S.I: Siamese Publishing.**

Questions?