



How to optimize your SAS code - Tips and Tricks

Meet the Speaker



Speaker: Vaisint Singh Muthudurai of Clinical Biometrics.

Organization: Zifo RnD Solutions.

Mr. Vaisint Singh Muthudurai Engagement Lead at Zifo RnD Solution has a professional experience of 5 years in the Clinical sector. He has been responsible for execution of various clinical studies with team of 15+ people for customers in US. He has led many projects that are prepared for analyses as well as regulatory body submissions for various studies for customers including US, APAC and Europe across various therapeutic areas. He has also been active in coordinating Organization level webinars related to Standards compliance, RWE/RWD implications etc.,

Agenda

- Introduction
- Environmental Setup
- Best Programming Practices
- Quality Control Techniques
- Time Saving Tips
- Conclusion

Introduction

- ***SAS (Statistical Analysis System)** is used to perform intricate statistical analyses and streamline the drug development process.*
- *Effectiveness of SAS code can vary greatly depending on the techniques used by the programmer.*
- *Practical tips and techniques can enhance productivity and help uncover hidden efficiencies in your workflow.*

1. Environmental Set Up

A. Creating a directory

- Using the **mkdir** command creating a directory in SAS is straightforward.
- Requires manually inputting directory paths with user-specific details (e.g., "S:\Name.Initial\Study_Name").
- Alternatively, the directory location can be dynamically retrieved with the **%sysget** function. {Syntax: %sysget(sas_execfilepath)}.
- Why use sysget?
 - Ensures the current path is retrieved accurately.
 - Avoids potential errors related to user-specific paths.

Input:

```
%let filepath = %sysget (sas_execfilepath);  
(x mkdir "path_name")  
  
%put &filepath.;
```

Output:

```
3  %let filepath = %sysget(sas_execfilepath);  
14  
15  %put &filepath;  
S:\Samyuktha.M\dm.sas
```

B. %include statement

*/ Concomitant Medication Coding Dictionary version number

%let CMDICT = 2022;

*/ Adverse events/Medical History Coding Dictionary version number

%let AEMHDICT = 25.0;

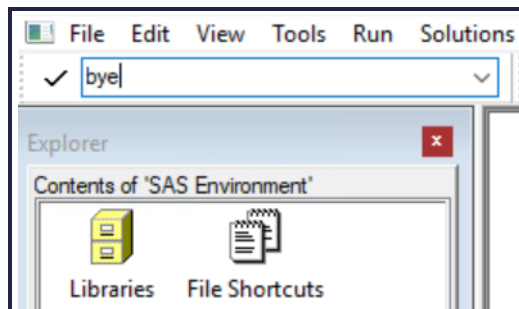
Consolidating all global macro information in a single SAS code.

%include "&proopath.\macrovar.sas";

Reference this program just once to access all the macros defined within it.

C. Quick Navigation

- To close the SAS window, simply type "bye" in the search bar.



2. Best Programming Practices

A. Excel Import

Problem: ○ *Importing Excel files dynamically without having to enter the entire file name. Eg: Deviation files that come with dates.*

Solution: ○ *Utilize wildcards, such as “?”, to import Excel files without having to specify the entire name.*

Advantages: ○ *Easier to manage and access your data.*
○ *Effortless handling of new files without altering code.*

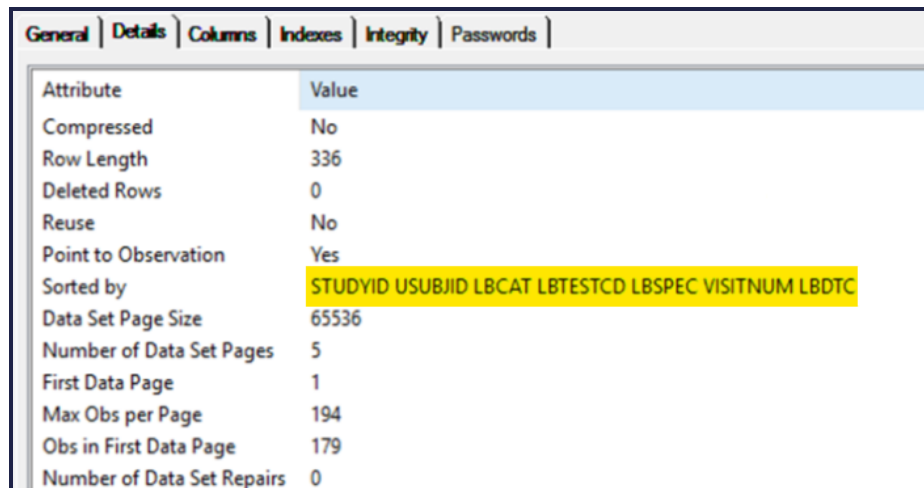
```
proc import datafile = "&SOURCEPATH.\&STUDYID._Randomization_&EXT.*?*.xls"  
    dbms = xls out = &OUT. replace;  
quit;
```

B. Viewing Sort Order

Problem: ○ *Is there any shortcut to view the sort order that is used in the dataset?*

Solution: ○ *Need to sort the dataset using the key variable (as per standards) at last to get the sort order in the properties(details) of SAS dataset.*

Advantages: ○ *Help reviewers to view the sort order easily.*



General Details Columns Indexes Integrity Passwords	
Attribute	Value
Compressed	No
Row Length	336
Deleted Rows	0
Reuse	No
Point to Observation	Yes
Sorted by	STUDYID USUBJID LBCAT LBTESTCD LBSPEC VISITNUM LBDMTC
Data Set Page Size	65536
Number of Data Set Pages	5
First Data Page	1
Max Obs per Page	194
Obs in First Data Page	179
Number of Data Set Repairs	0

C. LIST option in PROC REPORT

Problem: ○ Manually specifying *DEFINE* statements for each variable in a large dataset in *PROC REPORT*.

Solution: ○ The **LIST** option in *PROC REPORT* allows you to automatically generate all the necessary *DEFINE* statements without having to write them manually.

Advantages: ○ Saves time and effort.
○ Ensures that variables are not missed.

Input:

```
proc report data = ADSL LIST;
run;
```

Output:

```
38404
38405 proc report data = adsl LIST;
38406 run;

PROC REPORT DATA=WORK.ADSL LS=102 PS=56 SPLIT="/" CENTER ;
COLUMN STUDYID SUBJID SITEID SCRNID BRTHDAT AGE AGEU AGEGR1 AGEGRIN SEX RACE RACEOTH;

DEFINE STUDYID / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Study Identifier" ;
DEFINE SUBJID / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Subject Identifier for the Study" ;
DEFINE SITEID / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Study Site Identifier" ;
DEFINE SCRNID / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Screening Identifier" ;
DEFINE BRTHDAT / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Date of Birth" ;
DEFINE AGE / SUM FORMAT= BEST9. WIDTH=9 SPACING=2 RIGHT "Age" ;
DEFINE AGEU / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Age Units" ;
DEFINE AGEGR1 / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Pooled Age Group y" ;
DEFINE AGEGRIN / SUM FORMAT= BEST9. WIDTH=9 SPACING=2 RIGHT "Pooled Age Group y (N)" ;
DEFINE SEX / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Sex" ;
DEFINE RACE / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Race" ;
DEFINE RACEOTH / DISPLAY FORMAT= $200. WIDTH=200 SPACING=2 LEFT "Race Other" ;
RUN;

NOTE: There were 1168 observations read from the data set WORK.ADSL.
NOTE: PROCEDURE REPORT used (Total process time):
      real time      0.48 seconds
      cpu time       0.46 seconds
```

D. Indentation in RTF

Problem: ○ Sublines do not follow the indentation required in RTF.

Solution: ○ Simplify the process by using the RTF (Rich Text Format) commands.

○ First level of indentation - (^R/RTF'\li220') - ^ is odsescapechar.

○ Second level of indentation, we use (^R/RTF'\li440') - ^ is odsescapechar.

Advantages: ○ Less time-consuming – will have to add split character in other methods.

○ Achieve the desired indentation levels more efficiently.

Problem:

Type: n (%)
Category: n (%)
Subcategory1: n (%)
Subcategory2: n (%)
Any relevant protocol deviation
Any relevant eligibility deviation
Randomized or treated with study drug under > 1 subject identifier
Migraine history issue
Experienced migraines for < 1 year prior to screening
> 9 moderate or severe migraines per month in the 3 months prior to screening
Cardiovascular disease risk factor
Ischemic coronary artery disease
Other significant underlying cardiovascular disease

Solution:

Type: n (%)
Category: n (%)
Subcategory1: n (%)
Subcategory2: n (%)
Any relevant protocol deviation
Any relevant eligibility deviation
Randomized or treated with study drug under > 1 subject identifier
Migraine history issue
Experienced migraines for < 1 year prior to screening
> 9 moderate or severe migraines per month in the 3 months prior to screening
Cardiovascular disease risk factor
Ischemic coronary artery disease
Other significant underlying cardiovascular disease

E. PROC REPORT - Splitting and Repeating Variables

Problem: ○ Split the variables across multiple pages in RTF while retaining a few key variables on each page for identification purposes.

Solution: ○ **PAGE** option to specify the variable at which you want to start a new page.
○ **ID** option to indicate which variables should be repeated on each page.

ods rtf file=\\test\cars.doc
options nodate noprint;
title1 "(*ESC*)S={fontstyle=roman}Car Details";
proc report data = sashelp.cars;
column make model type origin drivetrain msrp invoice engine size;
define make / display id;
define model / display id;
define type / display;
define origin / display;
define drivetrain / display page;
define msrp / display;
define invoice / display;
define engine size / display;
run;
ods rtf close;

Page 1

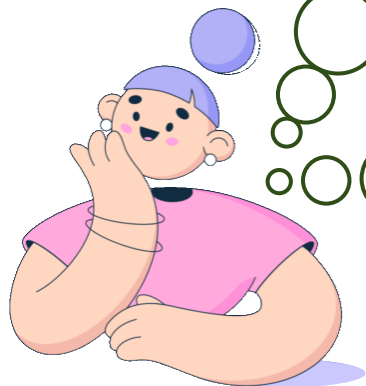
Make	Model	Type	Origin	Drivetrain	MSRP	Invoice	Engine (L)
Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5
Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2
Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2
Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5
Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5
Acura	NSX coupe 2dr manual S	Sports	Asia	Rear	\$89,765	\$79,978	3.2
Audi	A4 1.8T 4dr	Sedan	Europe	Front	\$25,940	\$23,508	1.8
Audi	A4 1.8T convertible 2dr	Sedan	Europe	Front	\$35,940	\$32,506	1.8
Audi	A4 3.0 4dr	Sedan	Europe	Front	\$31,840	\$28,846	3
Audi	A4 3.0 Quattro 4dr manual	Sedan	Europe	All	\$33,430	\$30,366	3
Audi	A4 3.0 Quattro 4dr auto	Sedan	Europe	All	\$34,480	\$31,388	3
Audi	A6 3.0 4dr	Sedan	Europe	Front	\$36,640	\$33,129	3
Audi	A6 3.0 Quattro 4dr	Sedan	Europe	All	\$39,640	\$35,992	3

Page 2

Make	Model	Drivetrain	MSRP	Invoice	Engine (L)
Acura	MDX	All	\$36,945	\$33,337	3.5
Acura	RSX Type S 2dr	Front	\$23,820	\$21,761	2
Acura	TSX 4dr	Front	\$26,990	\$24,647	2.4
Acura	TL 4dr	Front	\$33,195	\$30,299	3.2
Acura	3.5 RL 4dr	Front	\$43,755	\$39,014	3.5
Acura	3.5 RL w/Navigation 4dr	Front	\$46,100	\$41,100	3.5
Acura	NSX coupe 2dr manual S	Rear	\$89,765	\$79,978	3.2
Audi	A4 1.8T 4dr	Front	\$25,940	\$23,508	1.8
Audi	A4 1.8T convertible 2dr	Front	\$35,940	\$32,506	1.8
Audi	A4 3.0 4dr	Front	\$31,840	\$28,846	3
Audi	A4 3.0 Quattro 4dr manual	All	\$33,430	\$30,366	3
Audi	A4 3.0 Quattro 4dr auto	All	\$34,480	\$31,388	3
Audi	A6 3.0 4dr	Front	\$36,640	\$33,129	3
Audi	A6 3.0 Quattro 4dr	All	\$39,640	\$35,992	3

3. Quality Control Techniques

1. How can we identify double characters that emerge while importing external files, especially free-text entries?



Compress function with the 'g' modifier enables effective detection of double characters.

Output:

	dsterm	Graph	len	klen
1	In the 16th century, an age of great marine and terrestrial exploration, Ferdinand Magellan led the first expedition to sail around the world. As a young Portuguese noble, he served the king of Portugal, but he became involved in the quagmire of political intrigue at court and lost the king's favor.		3	1
2	。。just having double chars	。。	6	2
3	青い (あおい)	青いあおい	15	5

Input:

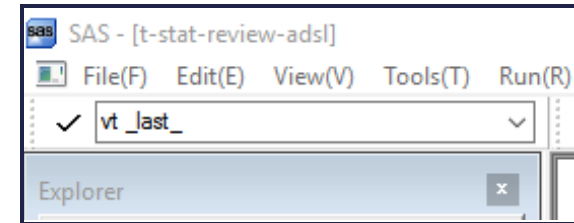
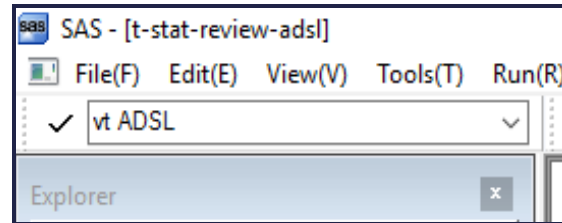
```
data CHECK;  
set DEVIATION;  
graph = compress(DSTERM," ", "g");  
len = length(DSTERM);  
klen = klength(DSTERM);  
run;
```

4. Time Saving Tips

A. Viewing dataset

- Users require a quick method to open datasets to avoid the time-consuming process of scrolling and searching for specific libraries within the SAS environment.

- TIPS:**
- Utilize the command **vt dataset_name** to open mentioned dataset and view the contents directly in the SAS environment.
 - Utilize the command **vt _last_** to open the last run dataset and view the contents directly in the SAS environment.
 - Implement a direct command to access datasets and make necessary adjustments without unnecessary delays.
 - Improve productivity and accuracy in data handling tasks.



B. Navigation between records

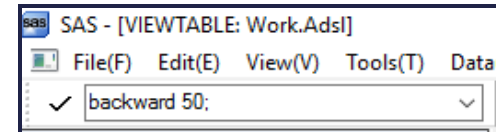
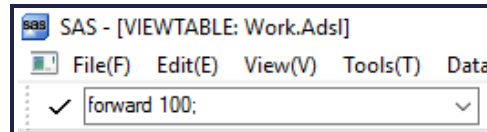
- Pinpointing a specific record for debugging in a large dataset is a difficult task.
- Require an efficient method to locate and isolate particular records.

- TIPS:**
- The **forward** and **backward** options in SAS are essential for navigating to a particular record.
 - By entering a number, users can jump to a specific position or move relative to their current position to find another record.

EG: ○ Dataset with 10,000 records, and you are currently viewing record 5,000:

○ **Forward Command:** forward 100 (moves to record 5,100)

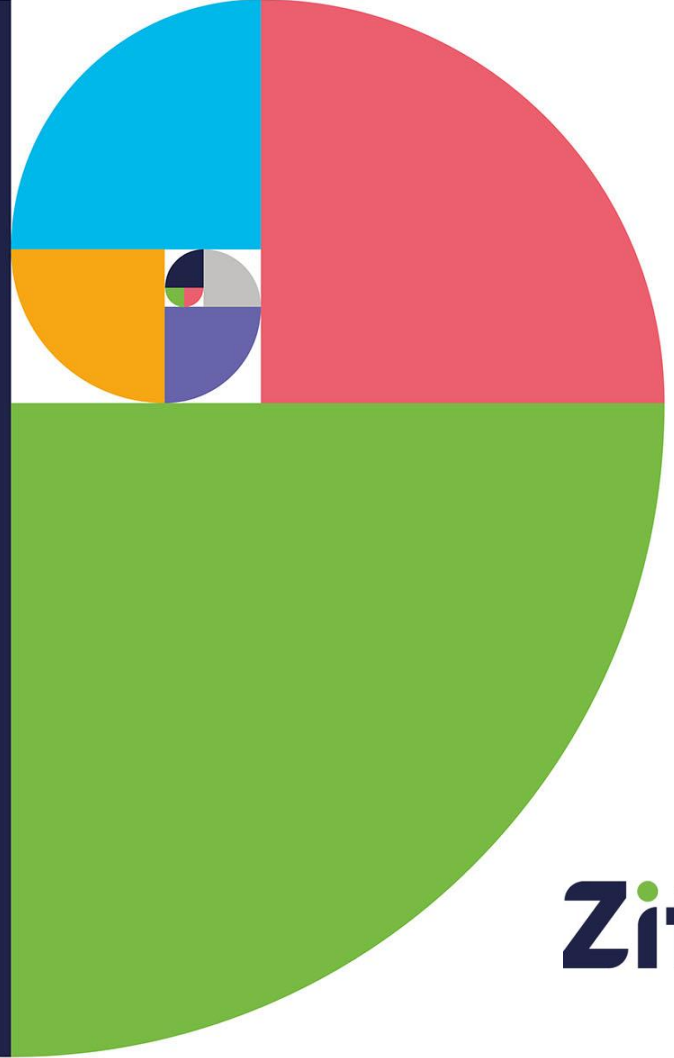
○ **Backward Command:** backward 50 (moves to record 4,950)



CONCLUSION

- *The tips and tricks we've discussed is just a small sample of the vast array of techniques available for working in SAS.*
- *There are countless strategies that can significantly enhance your efficiency and effectiveness.*
- *Primary aim - Highlight few key methods that can make a substantial difference in your SAS coding practices.*
- *These tips reduce manual effort and achieve more accurate and timely results in data analysis and reporting tasks.*

THANK YOU



Zifo.