

Spec to Code in Seconds: Use Microsoft Excel for Repetitive Code Writing

Pinky Anandani Dutta, Axio Research LLC

ABSTRACT

The role of a SAS programmer in the Pharmaceutical industry has become very versatile over the years. SAS programmers are no longer just coding using SAS ® but also performing many other functions including writing programming specifications, annotating Case Report Forms (CRF) and mocks shells, preparing documents for a New Drug Application (NDA), etc. As such, knowledge of robotic features in other tools and editors can come in very handy. One such tool is Microsoft Excel, which is relatively easy to learn and widely used already. This paper will discuss how programming specifications that require repetitive code can be transformed into SAS code in seconds by taking advantage of different functions in Excel.

INTRODUCTION

With the increasing number of clinical studies, SAS programmers are always looking for new ways to reduce coding time and efficiency. Repetitive coding, like if-else for example, often seems simple to write yet consumes a lot of time when typed up manually. Manual typing of code also introduces the risks of human errors such as typos, missing semicolon, missing quotation marks, etc. With that said, this paper will detail three specific examples of the use of Excel to generate repetitive code commonly used in clinical programming.

IF – ELSE STATEMENTS

If – Else Statements are very commonly used in a SAS program to perform different actions or assign results based on different conditions. If the conditions to these statements are based on set values, those conditions and their desired results or actions can be placed in different columns of an Excel spreadsheet, and the resulting SAS code can be generated in a new column using the CONCATENATE function in Excel as follows:

	A	B	C
1	PARAMCD	PARAM	Code for PARAM
2	BMI	Body Mass Index	If paramcd='BMI' then PARAM='Body Mass Index';
3	BSA	Body Surface Area	Else if paramcd='BSA' then PARAM='Body Surface Area';
4	DIABP	Diastolic Blood Pressure	Else if paramcd='DIABP' then PARAM='Diastolic Blood Pressure';
5	HEIGHT	Height	Else if paramcd='HEIGHT' then PARAM='Height';
6	HR	Heart Rate	Else if paramcd='HR' then PARAM='Heart Rate';
7	PULSE	Pulse Rate	Else if paramcd='PULSE' then PARAM='Pulse Rate';
8	RESP	Respiratory Rate	Else if paramcd='RESP' then PARAM='Respiratory Rate';
9	SYSBP	Systolic Blood Pressure	Else if paramcd='SYSBP' then PARAM='Systolic Blood Pressure';
10	TEMP	Temperature	Else if paramcd='TEMP' then PARAM='Temperature';

Display 1. Microsoft Excel Spreadsheet Example for Generating If – Else Statements

Cells C2 to C10 can easily be copied from the above table display and pasted directly into a SAS program DATA step.

LABEL ASSIGNMENT STATEMENT

This is a main topic in the paper. This paragraph uses the PaperBody style. This paragraph uses the PaperBody style.

	A	B	C
1	Variable	Label	Code for Label
2	STUDYID	Study Identifier	label STUDYID='Study Identifier'
3	USUBJID	Unique Subject Identifier	USUBJID='Unique Subject Identifier'
4	SITEID	Subject Identifier for the Study	SITEID='Subject Identifier for the Study'
5	AGE	Age	AGE='Age'
6	AGEU	Age Units	AGEU='Age Units'
7	AGEGR1	Pooled Age Group 1	AGEGR1='Pooled Age Group 1'
8	SEX	Sex	SEX='Sex'
9	RACE	Race	RACE='Race'
10	RACEN	Race (N)	RACEN='Race (N)'

MACRO CALL STATEMENTS

During the course of clinical programming, we often find the need to use macros by passing different macro parameters and calling a macro several times. A very common situation is when we are trying to create a batch file for running all of our TLF programs in a study.

	A	B	C	D	E	F
1	Program Name	TLF Type	TLF Number	Title	Full Title	Code for Macro Call
2	t_ds	Table	1	Subject Disposition	Table 1: Subject Disposition	%mbatch(t_ds); **Table 1: Subject Disposition**;
3	t_dm	Table	2	Demographics	Table 2: Demographics	%mbatch(t_dm); **Table 2: Demographics**;
4	t_ex	Table	3	Exposure	Table 3: Exposure	%mbatch(t_ex); **Table 3: Exposure**;
5	t_aesum	Table	4	Summary of Adverse Events	Table 4: Summary of Adverse Events	%mbatch(t_aesum); **Table 4: Summary of Adverse Events**;
6	t_teae	Table	5	Treatment-Emergent Adverse Events	Table 5: Treatment-Emergent Adverse Events	%mbatch(t_teae); **Table 5: Treatment-Emergent Adverse Events**;
7	t_chem	Table	6	Laboratory Findings - Chemistry	Table 6: Laboratory Findings - Chemistry	%mbatch(t_chem); **Table 6: Laboratory Findings - Chemistry**;
8	t_hem	Table	7	Laboratory Findings - Hematology	Table 7: Laboratory Findings - Hematology	%mbatch(t_hem); **Table 7: Laboratory Findings - Hematology**;
9	l_ae	Listing	1	Adverse Events	Listing 1: Adverse Events	%mbatch(l_ae); **Listing 1: Adverse Events**;
10	l_sae	Listing	2	Serious Adverse Events	Listing 2: Serious Adverse Events	%mbatch(l_sae); **Listing 2: Serious Adverse Events**;
11	l_dth	Listing	3	Death	Listing 3: Death	%mbatch(l_dth); **Listing 3: Death**;

Display 3. Microsoft Excel Spreadsheet Example for Generating Macro Call Statements

CONCLUSION

This paragraph uses the PaperBody style.

ACKNOWLEDGMENTS

This is the text for the acknowledgments. This paragraph uses the PaperBody style.

RECOMMENDED READING

CONTACT INFORMATION <HEADING 1>

Your comments and questions are valued and encouraged. Contact the author at:

Pinky Anandani
Axio Research LLC
Phone: 651-815-1638
pnanandani@gmail.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.