

An Obvious Yet Helpful Guide to Developing Recurring Reports in SAS®

Rachel Straney, University of Central Florida

ABSTRACT

Analysts, in particular SAS programmers, are often tasked with compiling data and generating reports at a moment's notice. It is paramount to make report generation efficient for ease of replication. The method of preparing and compiling recurring reports can be made easier when the process of reporting is well-defined. The objectives for this paper are to outline a checklist that can guide the creation of a reporting process for your data. SAS capabilities like labels and formats as well as the SAS Macro Facility play an important role in automating a reporting process. Identifying SAS procedures ahead of time, like PROC PRINT or PROC TABULATE, and developing a program that is flexible and adaptable can save time. General steps to successfully develop a reporting process will be described and detailed examples to show how to achieve this will be provided. The overall benefits of planning a reporting process are to reduce reporting errors, reduce programming time and have a well-designed report that is meaningful.

INTRODUCTION

There are four important points laid out in this paper that will guide you in the process of developing a SAS program that can efficiently and effectively generate reports on a recurring basis. These are general concepts and their applications range greatly depending on the reporting process you wish to achieve as well as the SAS platform you may be using. Examples are shared for each of these points to provide more insight and explanation.

Plan your report

Rely on data sources that are stable and reliable

Organize your reporting process by program function

Consider using SAS macros to reduce your code

SAS Enterprise Guide (EG) was used to develop the example report for this paper, but Base SAS can be used as well. The final report generated is in a PDF format using the SAS Output Delivery System (ODS) and can be viewed on the following page.

THE FINAL PDF REPORT USED FOR THE EXAMPLES IN THIS PAPER

Student Perceptions Report for Academic Program Review

Program: Nursing BSN

Class of 2017-2018

The purpose of this document is to summarize student perceptions and behaviors while earning their degree using survey data collected close to the time of graduation from the UCF Graduating Senior Survey (GSS), the First Destination Survey (FDS), and the Senior Program Specific Survey (if applicable). The report contains information on engagement activities students participated in while at UCF, perceptions on advising and other information that may be useful during the Academic Program Review process.

This report summarizes survey results for the 1,554 students who graduated with a degree in Nursing BSN from 2014-2015 to 2016-2017. The survey response rates for the Graduating Senior Survey and the First Destination Survey for Nursing BSN graduates are 84% and 88%, respectively. The number of graduates may differ slightly from current data due to the timing of survey processing.

Operational Excellence and Assessment Support

June 2017

Student Perceptions Report for Academic Program Review | Nursing BSN | 2017-2018

Academic Advising Resources

The table below summarizes which sources students indicated were the most beneficial during their last 2 years at UCF.

Table 2

Please choose up to 3 sources from which you received the MOST BENEFICIAL academic advising during your last 2 years at UCF (select all that apply)	Endorsements+					
	2014-15		2015-16		2016-17	
	Count	%	Count	%	Count	%
College advising office staff	116	26.7%	161	27.5%	153	27.8%
Athletic Advising (AGSA)	2	0.4%	6	1.0%	3	0.5%
Faculty advisor(s)	207	47.6%	289	49.4%	297	54.0%
Student advisor(s)	43	9.9%	71	12.1%	78	14.1%
Friends	154	35.4%	199	34.0%	170	30.9%
Family	87	20.0%	104	17.7%	107	19.4%
Co-workers	38	8.7%	34	5.8%	20	3.6%
Other students	200	46.0%	254	43.4%	249	45.2%
Degree audits (DARG)	132	30.4%	183	31.2%	183	33.2%
Webprinted material (including catalog)	99	22.8%	123	21.0%	102	18.5%
I did not seek help from advisors	52	11.9%	88	15.0%	60	10.9%

* Percents calculated for the question above do not add up to 100% since the question asks respondents to 'select all that apply.' The total number of respondents for 2014-15, 2015-16, 2016-17 are: 434, 585, 550, respectively.

Operational Excellence and Assessment Support
Division of Teaching and Learning

June 2017
Page 2

Student Perceptions Report for Academic Program Review | Nursing BSN | 2017-2018

Graduating Senior UCF Survey Results

Experience in the Major

The table below shows positive endorsements for survey questions related to student experiences in the major.

Table 1

Experience in the Major*	Positive Endorsements+		
	2014-15	2015-16	2016-17
Number of Respondents	434	603	570
In my major, the courses I needed were available	95.8%	95.0%	96.6%
In my major, the courses I needed were available at convenient times	94.7%	91.8%	94.0%
In my major, there was a good range of courses	92.3%	91.3%	93.0%
In my major, I was satisfied with the number of face-to-face courses available	93.7%	91.4%	93.4%
In my major, I was satisfied with the number of online courses available	92.9%	89.9%	92.4%
In my major, too many of my classes were too large (-Disagreeing is a positive response-)	44.1%	38.0%	37.2%
In my major, my professors were good teachers	88.6%	81.5%	89.2%
In my major, my professors were available for help outside of class	89.0%	85.2%	88.2%
In my major, I was provided opportunities to develop appropriate computer skills	90.5%	84.8%	86.6%
In my major, my training in computer skills prepared me for today's technology	88.9%	84.5%	86.8%
Outside my major, required courses to meet general education were available	91.7%	88.6%	91.2%
Outside my major, the courses I wanted were available at convenient times	87.8%	84.5%	87.9%
Outside my major, courses in other departments, but required by my major, were available	88.7%	85.7%	89.1%
Outside my major, too many of my classes were too large (-Disagreeing is a positive response-)	57.8%	57.6%	58.2%

* Cronbach's alpha measures how closely related a set of items are as a group and is considered a measure of scale reliability. It ranges from 0 to 1. For the set of items listed above, the Cronbach's alpha is 0.84, which is considered good internal consistency for a scale.

+ Positive endorsements include 'Strongly agree' and 'Agree' responses to items unless otherwise specified.

Operational Excellence and Assessment Support
Division of Teaching and Learning

June 2017
Page 1

Student Perceptions Report for Academic Program Review | Nursing BSN | 2017-2018

Academic Advising from College Office

The table below shows positive endorsements for survey questions related to experiences with the college advising office.

Table 3

Academic Advising from College Office*	Positive Endorsements+		
	2014-15	2015-16	2016-17
Number of Respondents	420	563	522
In general the college advising staff members were helpful.	58.4%	56.7%	59.1%
In general the college advising staff members were knowledgeable.	59.0%	57.0%	60.0%
College advising staff members were available during posted office hours.	57.1%	54.8%	56.6%
Sufficient time was available during college advising sessions.	54.8%	53.3%	55.0%
The advice I received from college advising staff was very useful for my EDUCATIONAL goals.	58.7%	56.6%	59.4%
The advice I received from college advising staff was very useful for my CAREER goals.	56.2%	54.8%	56.7%

* Cronbach's alpha measures how closely related a set of items are as a group and is considered a measure of scale reliability. It ranges from 0 to 1. For the set of items listed above, the Cronbach's alpha is 0.99, which is considered good internal consistency for a scale.

+ Positive endorsements include 'Strongly agree' and 'Agree' responses to items unless otherwise specified.

Operational Excellence and Assessment Support
Division of Teaching and Learning

June 2017
Page 3

PLAN YOUR REPORT

It is a good idea to visualize how your final report will look before programming. The first step in developing a reporting program involves creating a mockup or draft of what your report will look like. If you intend to deliver the report in excel you can develop a template using that software. A less time consuming strategy is to draw it out the old fashioned way on paper. Below is an image of a primitive example of a report mock up. It may not be fancy, but it serves its purpose.

ELEMENTS IN YOUR REPORT

Taking the time to envision the layout of your report will aid in the programming process. It can help determine the SAS procedures you will require as well as identify places in your program that can be simplified with SAS macros and other programming logic. Pay particular attention to areas of your report that will require the use of SAS procedures (PROC TABULATE, PROC MEANS). If you find the tables you wish to display within your report are similarly structured, this is an opportunity to reduce your SAS code. We will come back to this idea later.

Question	Year	n %	n %	n %
MAJAVAIL				
MAJGOOD				
MAJOPT				
MAJKAN				
MAJFEL				

USING ODS TO GENERATE THE FORMATTED FILE

SAS has the ability to generate and produce reports in different formats using ODS capability. Although the ODS statement is used for much more than generating SAS output to other file formats, leveraging ODS statements in SAS can automate the process of converting SAS output to a format you intend to share with consumers of the report. There are several formats that ODS supports including Portable Document Format (PDF), Rich Text Format (RTF), HyperText Markup Language (HTML) and Comma Separated Values (CSV). Relatively new in SAS 9.4 are ODS statements for PowerPoint and Excel, although it should be noted that the ability to export output to an Excel file has been available for a while through other means like ODS ExcelXP Tagsets and Dynamic Data Exchange (DDE). The example in this paper relies on ODS PDF. To successfully produce a PDF document using ODS, simply begin and end your program with the ODS statements shown below.

```
ods pdf;  
    <SAS program>  
ods pdf close;
```

Questions to ask yourself during this step....

- Which tables, charts or graphs will be included in the report?
- Where do you want specific text or images to be placed (headers/footers, company logo, text/write-up)?
- How will you deliver your report? Which file format? (Excel, PDF)

RELY ON DATA SOURCES THAT ARE STABLE AND RELIABLE

The most sophisticated SAS program written to efficiently generate a recurring report will quickly break if the data is not available in a structure the program needs. In other words, a report cannot be generated quickly if the data is not stable and predictable. It is important to identify element names and table sources ahead of time and ensure these will be consistent for future reporting. If you are relying on data that are maintained by a third party, it is a good idea to communicate with them and assess the stability of the source. An additional thing to consider is when new data are available, so that your report is scheduled to run at a reasonable time.

Questions to ask yourself during this step....

- Which sources of data do you need? (Tables, Elements)
- How often are data refreshed and does the schedule for reporting account for this?

ORGANIZE YOUR REPORTING PROCESS BY PROGRAM FUNCTION

Organizing your program may differ depending on the SAS platform you are using. For example, if the report you are generating is lengthy and you are working in EG it may benefit you to separate program functions into separate nodes of code as has been done in Figure 1.

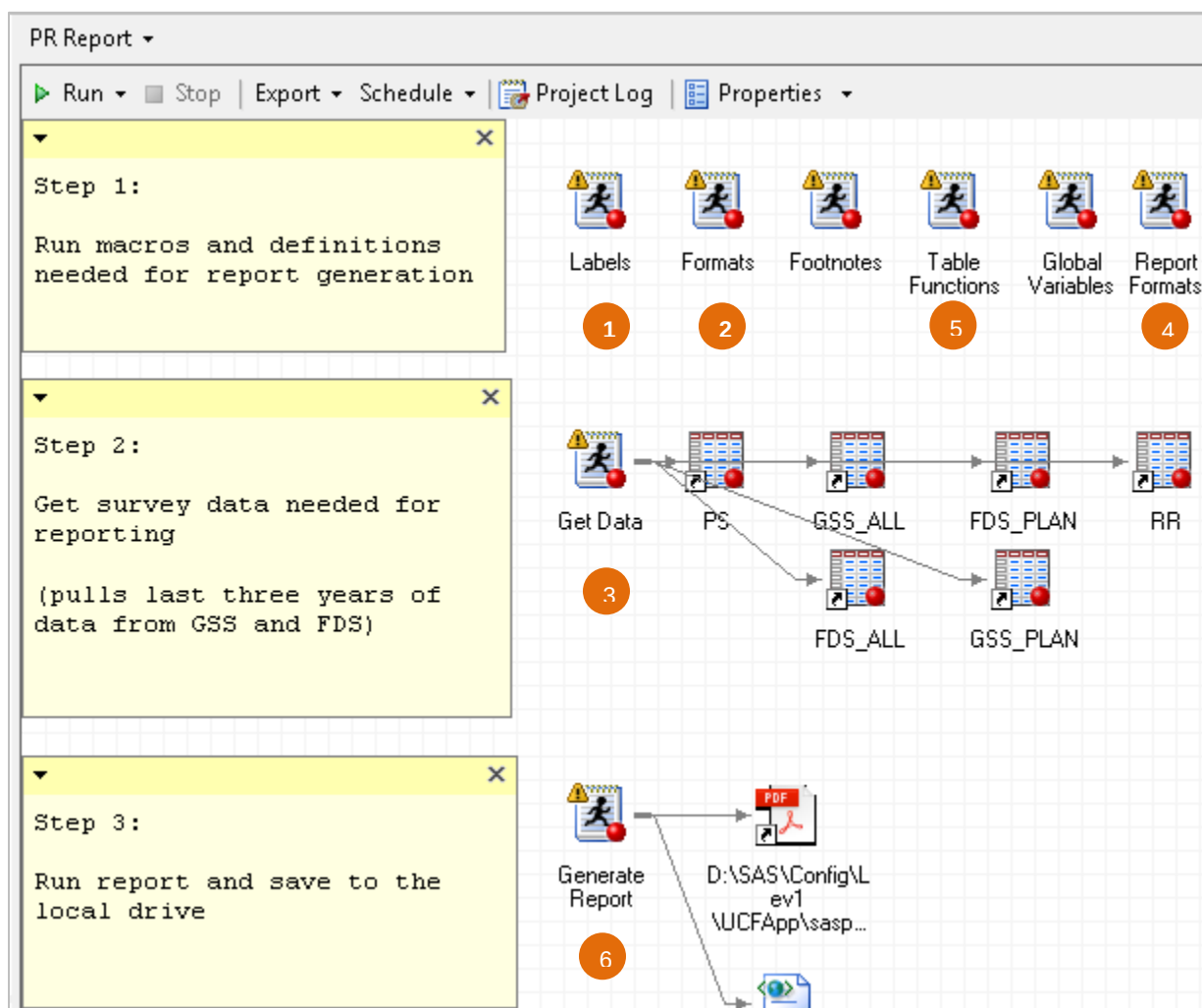


Figure 1. EG Project used to create the report

There are alternatives to structuring your SAS program, like utilizing the %include statement to call parts of your program from SAS files saved in other locations. You can always place your entire program into one program node, separating out sections with clear comments. If you have developed many SAS macro functions and you find you use them heavily across projects, a more advanced option is to create a permanent SAS Macro library by utilizing Compiled Stored Macros or the AUTOCALL options. Whatever your preference, the purpose is to clearly organize sections of your program so that editing or updating the program for any reason is easy and intuitive.

Below describes the contents of each SAS program node from Figure 1. Many of the nodes of code listed contain a single SAS macro or define several for use in other parts of the program.

Step 1

- **Labels:** SAS macro that defines and houses all the labels for elements used in the report; This macro is later called in Step 2 - Get Data
- **Formats:** SAS macro that defines and houses all the formats for elements used in the report; This macro is later called in Step 2 - Get Data
- **Footnotes:** SAS macro that places any footnotes needed at the end of tables in the report; This macro is called in Table Functions
- **Table Functions:** SAS program that houses several SAS macros used to generate tables in the report; These macros are called in Step 3 - Generate Report and mainly pass the element used to create the table
- **Global Variables:** SAS program that defines SAS macro variables that are specific to the report content (year or date placed in the report, specific names of groups one is reporting on)
- **Report Format:** SAS program that defines SAS macro variables that are specific to table or report formatting (PROC TEMPLATE, column widths for tables, background colors for table headers)

Step 2

- **Get Data:** Data processing required to structure the data for reporting (libname statements, data steps, PROC SQL)

Step 3

- **Generate Report:** Main SAS program to generate report; Uses ODS PDF statement and places text and tables as needed

Examples of how some of these SAS programs interact is explained in the next section.

CONSIDER USING SAS MACROS TO REDUCE YOUR CODE

SAS Macros are a useful way to simplify any SAS program. As was previously mentioned, much of the SAS code written throughout the example from Figure 1 uses the SAS Macro Facility Language.

LABELS AND FORMATS

The example below shows how labels and formats are defined and applied to the SAS data set you are working with. In parts 1 and 2, the labels and formats are defined in separate macros and these macros are then called later in part 3 during data processing and preparation. There are, of course, alternative ways to do this. Labels and formats can be applied to data sets permanently or they can be defined during procedure statements. The decision of how to store and apply labels and formats is really a function of how your reporting process is organized and the nature of your data.

1 Code found in "Labels" in EG Project

```
%macro labelquestions;

label

ADVISING='Please choose up to 3 sources from which you received the MOST
BENEFICIAL academic advising during your last 2 years at UCF (select all that
apply)'
ADVISING_1 = 'College advising office staff'
ADVISING_2 = 'Athletic Advising (ASSA)'
... <more SAS labels>
;

%mend;
```

2 Code found in "Formats" in EG Project

```
%macro response_format;

proc format;

picture pct
  0.005-100='009.9%'
  0-0.0049=' '
;

value $ major
  'RESPONDENT_TOTAL_COUNT' = 'Number of Respondents'
  'MAJAVAIL_NEW' = 'In my major, the courses I needed were available'
  'OTRLARGE_NEW' = 'Outside my major, too many of my classes were too large'
  ...
;

... <more SAS formats>

%mend;
```

3 Code found in “Get Data” in EG Project

<multiple library statements to access SAS datasets>

%response_format;

```
data work.GSS_ALL;
    set gss1415.gssay_new
        gss1516.gssay_new (drop=URE )
        gss1617.gssay_new (drop=timestamp) ;
    denom = 1;
    age=.; gender=.; race=. ; status=.;
    rename
        ACAD_PLAN = ACAD_PLAN_GSS finished = finished_GSS
        ACAD_YEAR_AWARDED = YEAR_GSS age_df = age CF_BOE_GENDER = gender
        CF_BOE_RACE = race CF_BOE_RCENT_TYPE = status ;
```

run;

... <more data manipulation> ...

```
data work.GSS_PLAN;
```

```
set work.GSS_ALL;
```

%labelquestions;

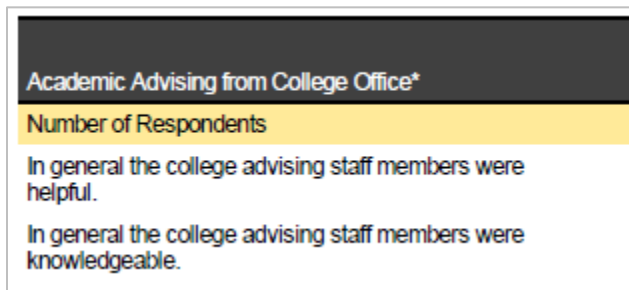
```
if YEAR_GSS = '2014-15' then year_num = 1;
else if YEAR_GSS = '2015-16' then year_num = 2;
else if YEAR_GSS = '2016-17' then year_num = 3;
where acad_plan_GSS = "&plan";
```

run;

... <more data manipulation> ...

SIMPLIFYING TABLE STYLES

By creating SAS macro variables and defining them in an easily accessible part of your program, you can make changes to your report quickly. Recall the final output of the PDF report example:



The image shows a close-up of a PDF report. It features a table with a black header row containing the text "Academic Advising from College Office*". Below this is a gold-colored row with the text "Number of Respondents". The table body contains two rows of text: "In general the college advising staff members were helpful." and "In general the college advising staff members were knowledgeable."

Academic Advising from College Office*
Number of Respondents
In general the college advising staff members were helpful.
In general the college advising staff members were knowledgeable.

Figure 2. Close up screenshot of tables from final PDF report

The black and gold background colors for headers were defined as macro variables in part 4 and were then referenced later in all macros defined in the “Table Functions” program (part 5). Recall that there are several macros defined in “Table Functions,” many of which use PROC TABULATE to structure certain types of table output for the report. An example of one of these macros is shown below. By defining the header background colors as macro variables in one place in the program, you can make changes easily if ever the style needs to be altered.

4 Code found in “Report Formats” in EG Project

```
%let respondent_bandcolor = #ffea99;  
%let header_bandcolor = #404040;  
%let header_textcolor = white;
```

5 Code found in “Table Functions” in EG Project

```
%macro matrix_questions(varlist, var_group, q_stem);  
  
<additional SAS code> ...  
  
proc report data=Report spanrows style(header) =  
{background=&header_bandcolor foreground=&header_textcolor just=c vjust=m}  
style(column) = {background=white just=l vjust=m}  
style(report)={outputwidth=100%};  
  
column _name_ ('Positive Endorsements+' ( _numeric_)) ;  
  
label &yr_labels;  
define _Name_ / "&q_stem" style(header)={just=l} ;  
define YR_1 - YR_3 / style(column)={just=c Cellwidth=&stat_width_PR.%}  
style(header)={just=c };  
  
compute _NAME_;  
if lowercase(_NAME_) = 'number of respondents' then do;  
call define(_row_, 'style', "style={background=&respondent_bandcolor}");  
end;  
endcomp;  
  
format _NAME_ $&var_group.. _numeric_ stacked.;  
run;  
  
...<additional SAS code>  
  
%mend;
```

BRINGING IT ALL TOGETHER IN A MAIN PROGRAM

I suggest having one part of your program serve as the main stage. This is where most, if not all, other parts (other SAS programs, macro functions, macro variables) are called for use. In the example from Figure 1, this is part 6 – the “Generate Report” program. Only a portion of the program from part 6 is shown as the code is quite lengthy. Any macro variables used throughout the text are highlighted and macro functions to produce tables are noted as well.

6 Code found in “Generate Report” in EG Project

```
/*paper options*/  
Option orientation=portrait papersize=letter topmargin=0.3in  
bottommargin=0.3in leftmargin=0.5in rightmargin=0.5in nonumber nodate;  
ods escapechar='^';
```

```

/*open ODS PDF statement*/
ods layout start width=8.5in height=11in ;
ods pdf style=Custom startpage=never notoc dpi=300;

/*clear ALL titles and footnotes*/
title; footnote;

/*****
/*Cover Page*/

proc odstext;
p '^S={preimage="\\TheTab_KGrgb_300ppi_85_75.png"';
p " ";
p " ";
p '^S={FONT_SIZE=18pt JUST=l FONT_FACE=arial}Student Perceptions Report for
Academic Program Review^{newline}^{newline}';
p '^S={FONT_SIZE=12pt JUST=l FONT_FACE=arial}Program:
&plan_name.^{newline}^{newline}';
p '^S={FONT_SIZE=12pt JUST=l FONT_FACE=arial}Class of &class_year";
p '^S={FONT_SIZE=12pt JUST=l FONT_FACE=arial COLOR=CXFFD700}
_____';
run;

...<additional SAS code> ...

ods pdf startpage=now; /*Page break*/
/*****

/*add headers and footers to the remaining pages*/
title HEIGHT=9PT J=LEFT F=ARIAL bold
'^S={preimage="\\TheTab_KGrgb_300ppi_85_75_80.png"
posttext="^-200y Student Perceptions Report for Academic Program Review ||
&plan_name. || &class_year."';

footnote1 HEIGHT=9PT J=LEFT F=ARIAL bold "Operational Excellence and
Assessment Support" J=RIGHT F=ARIAL bold "&date";

footnote2 HEIGHT=9PT J=LEFT F=ARIAL bold "Division of Teaching and Learning"
J=RIGHT F=ARIAL bold "Page ^{thispage}";

proc odstext;
p '^S={FONT_SIZE=11pt JUST=l FONT_FACE=arial
COLOR=&section_header_color}Experience in the Major^{newline}';
p '^S={JUST=l FONT_FACE=arial}The table below shows positive endorsements for
survey questions related to student experiences in the
major.^{newline}^{newline}';
p "Table 1";
run;

%matrix_questions(
varlist =
MAJAVAIL_NEW MAJTIME_NEW MAJRANGE_NEW MAJF2F_NEW MAJWEB_NEW MAJLARGE_NEW
PROFGOOD_NEW PROFVAI_NEW MAJCOMP_NEW MAJTECH_NEW OTRGEP_NEW OTRTIME_NEW
OTRAVAIL_NEW OTRLARGE_NEW,
var_group = MAJOR,

```

Insert table, macro defined
in "Table Functions"

```

q_stem = Experience in the Major*
);

ods pdf startpage=now; /*Page break*/
/*****/

proc odstext;
p "^S={FONT_SIZE=11pt JUST=l FONT_FACE=arial
COLOR=&section_header_color}Academic Advising Resources^{newline}";
p "The table below summarizes which sources students indicated were the most
beneficial during their last 2 years at UCF.";
p " ";
p "Table 2";
run;

%multioption_questions(ADVISING, 11);

...<additional SAS code> ...

/*close ODS layout and PDF statement*/
ods layout end;
ods pdf close;

```

Insert table, macro defined
in "Table Functions"

Questions to ask yourself during this step....

- Is data processing/manipulation needed to get the data structured the way you need it?
- How will you apply any labels or formats required for your tables and charts?
- If you have identified certain SAS procedures to produce output, can these be written as macro functions?
- Is there any text in the report that may need updating for future reports? If so, can these be written as a macro variables?

CONCLUSION

Developing a SAS program to generate recurring reports can be a somewhat time intensive process up front. However, the benefit is that producing reports is quick and updating your program for any reason is easy. Planning how your report will look beforehand can assist in the programming process like identifying SAS procedures required for the job and organizing parts of your program in an efficient way. It is important to remember that the source of your data plays a vital role in making the reporting program work for you. Last but not least, the SAS Macro Facility is a useful way to reduce code and simplify parts of your program. The next time you work on a project that will most likely be requested on an on-going basis, I encourage you to use the steps laid out in this paper and design a SAS reporting process of your own.

REFERENCES

Bessler, LeRoy. "Getting Started with, and Getting the Most out of, SAS® ODS PDF: No Mastery of PROC TEMPLATE Required." October 10-12, 2005. *Proceedings of the Pharmaceutical Users Software Exchange 2005*. Available at <http://www.lexjansen.com/phuse/2005/ts/ts05.pdf>

Carpenter, Arthur L. "Building and Using Macro Libraries." April 14-17, 2002. *Proceedings of the SUGI 27 Conference*. Orlando, FL. Available at <http://www2.sas.com/proceedings/sugi27/p017-27.pdf>

Mendez, Lisa. "Create a Main Program to Run Multiple SAS® Programs: Utilizing the %INCLUDE Statement." April 4-7, 2011. *Proceedings of the 2011 SAS Global Forum*. Available at <http://support.sas.com/resources/papers/proceedings11/100-2011.pdf>

SAS Institute Inc. 2012. SAS® 9.4 Output Delivery System: User's Guide, Fifth Edition. Available at <http://documentation.sas.com/?docsetId=odsug&docsetTarget=bookinfo.htm&docsetVersion=9.4&locale=en>

CONTACT INFORMATION

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

Rachel Straney
University of Central Florida
12424 Research Parkway, Suite 225
Orlando, FL 32826
407-882-0280
rstraney@ucf.edu

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.