Report Building

SAS and Microsoft Word VBA Made Easy

JSM

7/28/2019

Scott Kreider, MS
Rocky Mountain Poison and Drug Safety
A Division of Denver Health

OUTLINE

- Motivating example
- Big Idea
- Background
  - VBA
- SAS and VBA Code examples
- Example report
- Future directions
- References and resources
- Conclusion

MOTIVATING EXAMPLE

I have a statistical report that I want to send to a client. I do most or all of my analysis in SAS and I have some (or a LOT of) tables and figures that I want to put into a Word document, according to my internal style guide, along with some background and methods sections.

MOTIVATING EXAMPLE

I have a statistical report that I want to send to a client. I do most or all of my analysis in SAS and I have some (or a LOT of) tables and figures that I want to put into a Word document, according to my internal style guide, along with some background and methods sections.

Things I want in my report:

1) Sequential and auto-updating chapter/section numbering
2) Appropriately numbered and sequential table and figure titles within sections
3) Title page with my organization logo
4) Table of contents
5) Formatting based on internal organizational specifications
6) More!

MOTIVATING EXAMPLE

What are my options?

MOTIVATING EXAMPLE

What are my options?

1) Manually copy/paste the tables and figures into a Word document

MOTIVATING EXAMPLE

What are my options?

1) Manually copy/paste the tables and figures into a Word document
2) Hire an RA or external contractor to put the report together

MOTIVATING EXAMPLE

What are my options?

1) Manually copy/paste the tables and figures into a Word document
2) Hire an RA or external contractor to put the report together
3) Draft up some reusable SAS and VBA code to produce the results I want!

BIG IDEA

- Use SAS and Microsoft VBA to generate a statistical report based on my desired specifications

BIG IDEA

- Perform minimal formatting within SAS (raw RTF output does NOT necessarily have to look good)
- Use SAS and accompanying macros to generate the report body and to
    1) Assign MSWord STYLE tags to sections of text
    2) Include linked list indicators for auto-numbering of sections and titles

BIG IDEA

- **Use VBA code to**
  1) Update/create the necessary styles within MSWord
  2) Apply those updated styles to the tagged sections of text
  3) Update auto-numbered fields
  4) Additional formatting
  5) Insert list of tables, list of figures, table of contents

BACKGROUND

SAS is not a word processor, but…

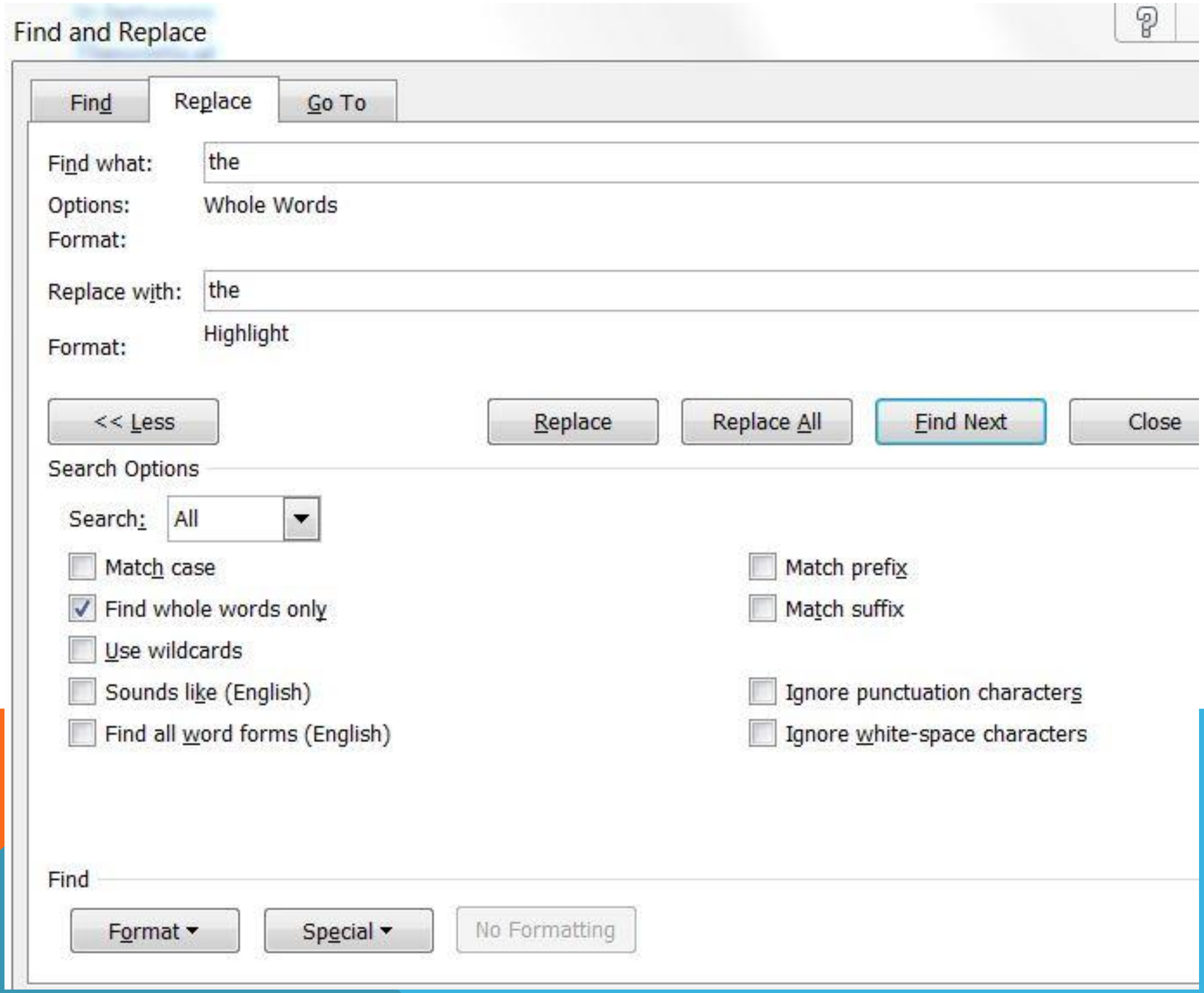
- It can be used to mimic a word processor, to an extent
- Can work in conjunction with Microsoft products (Word, Excel, etc…) to tackle this problem

VBA

- Visual Basic for Applications (VBA) is what MS products run behind the scenes to perform tasks
  - Example: I want to find, and highlight in yellow, all instances of the word “the” in the Word document. How would I do this?

VBA

Find and replace menu:



Find and Replace

Find what: the

Options: Whole Words

Format:

Replace with: the

Format: Highlight

Search Options

Search: All

- ☐ Match case
- ☑ Find whole words only
- ☐ Use wildcards
- ☐ Sounds like (English)
- ☐ Find all word forms (English)
- ☐ Match prefix
- ☐ Match suffix
- ☐ Ignore punctuation characters
- ☐ Ignore white-space characters

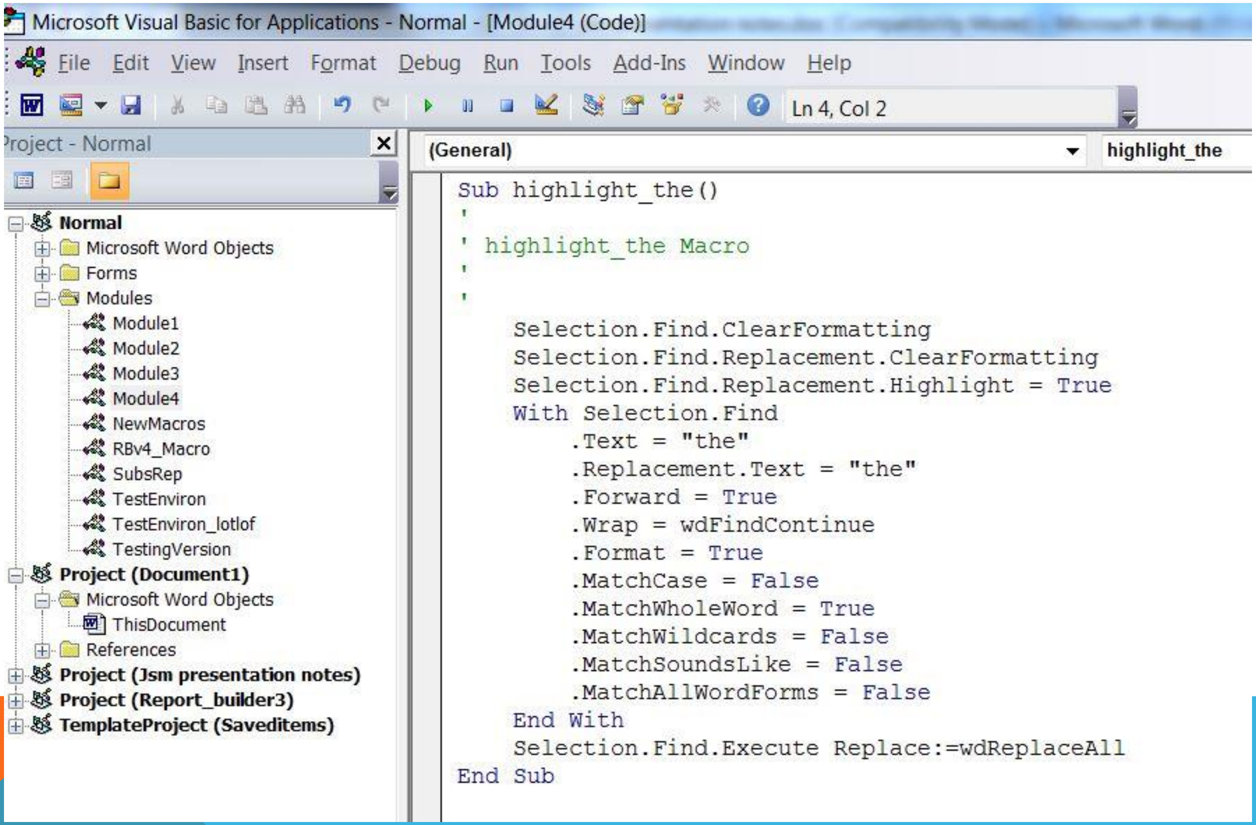
Find

VBA

OR…

VBA

Run the underlying VBA code



```
Sub highlight_the()
'
' highlight_the Macro
'
'
    Selection.Find.ClearFormatting
    Selection.Find.Replacement.ClearFormatting
    Selection.Find.Replacement.Highlight = True
    With Selection.Find
        .Text = "the"
        .Replacement.Text = "the"
        .Forward = True
        .Wrap = wdFindContinue
        .Format = True
        .MatchCase = False
        .MatchWholeWord = True
        .MatchWildcards = False
        .MatchSoundsLike = False
        .MatchAllWordForms = False
    End With
    Selection.Find.Execute Replace:=wdReplaceAll
End Sub
```

VBA

Can leverage SAS to build this underlying VBA code into the RTF output directly from SAS to generate text/field codes that Word can understand

SAS MACROS

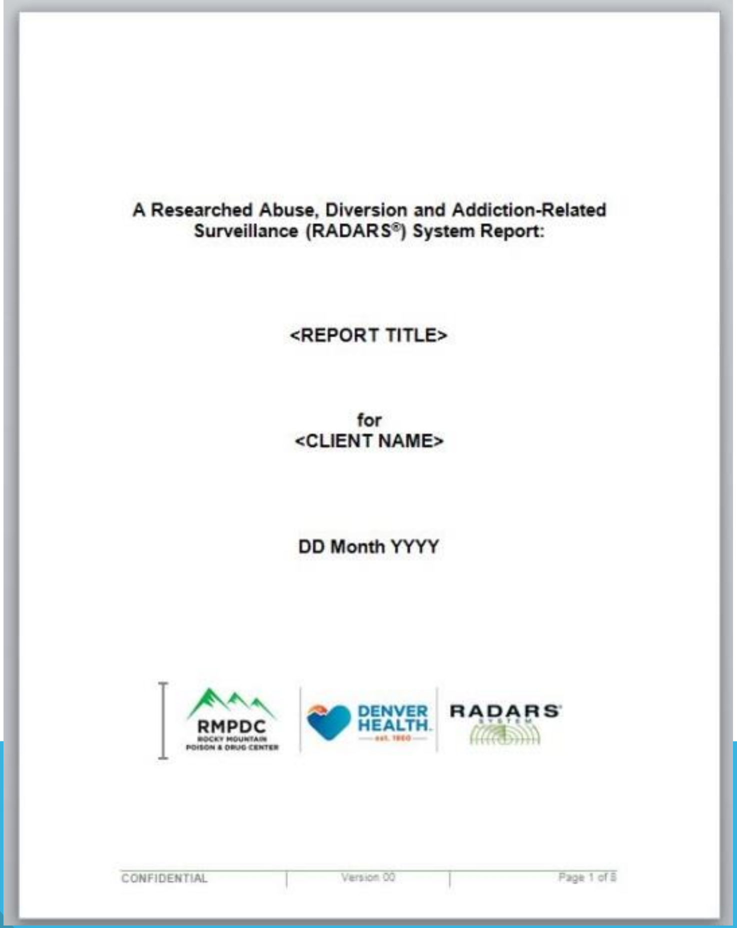
%Titlepage(title,reportnumber,effdate)

- Creates title page to my specifications
  - Title = “Report title”
  - Reportnumber = version number or report number (in document footer)
  - Effdate = effective date (default = today)

```
%titlepage(title="YOUR REPORT TITLE",reportnumber=00,effdate=10/15/2015);
```

SAS MACROS

%Titlepage(title,reportnumber,style=&globalstyle,effdate=&today1,loa=F)



A Researched Abuse, Diversion and Addiction-Related Surveillance (RADARS®) System Report:

<REPORT TITLE>

for
<CLIENT NAME>

DD Month YYYY

CONFIDENTIAL | Version 00 | Page 1 of 8

SAS MACROS

%Sectiontitle(level,description)

- Creates chapter/section/etc… title and assigns auto-numbering indicator to it
- 1 = chapter, 2 = section, 3 = subsection, 4 = subsubsection, 5 = subsubsubsection
  - Level = section level (required)
  - Description = “Section Title (preferably in quotation marks)”
  - Creates global macro variable &globallevel to be used in the subsequent macros

```
%sectiontitle(1,“Chapter Name");
```

NOTE:

- Assigns MSWord style tags Heading 1 through Heading 5 to the section titles

SAS MACROS

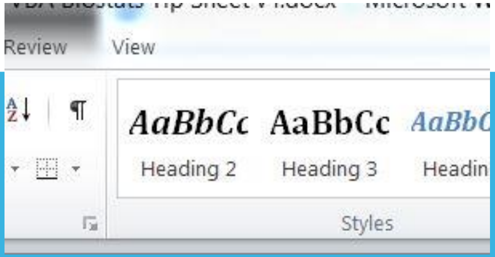
%Sectiontitle(level,description)

- Creates chapter/section/etc… title and assigns auto-numbering indicator to it
- 1 = chapter, 2 = section, 3 = subsection, 4 = subsubsection, 5 = subsubsubsection

  - Level = section level (required)
  - Description = “Section Title (preferably in quotation marks)”
  - Creates global macro variable &globallevel to be used in the subsequent macros

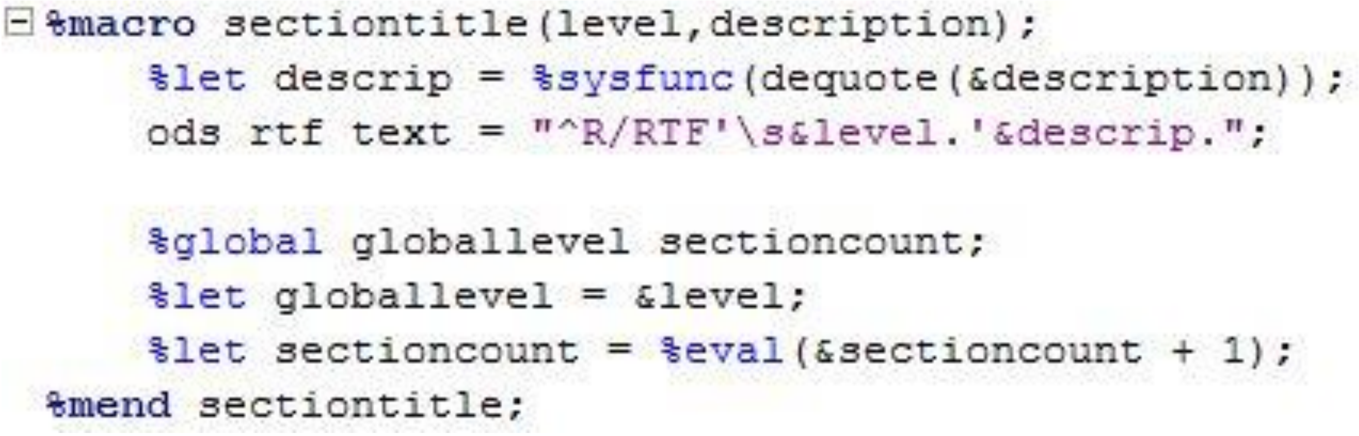
```
%sectiontitle(1,"Chapter Name");
```

NOTE:
- Assigns MSWord style tags Heading 1 through Heading 5 to the section titles



SAS MACROS

%Sectiontitle(level,description)

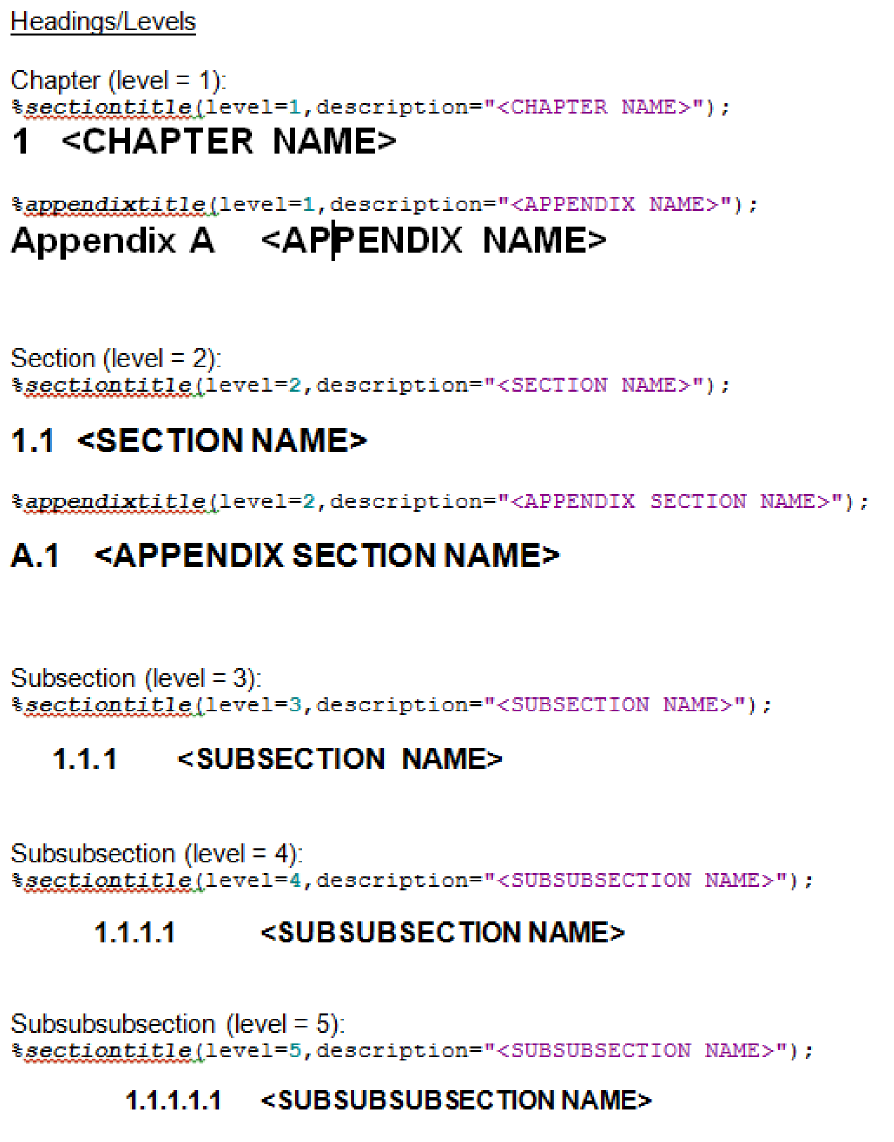


```
%macro sectiontitle(level,description);
     %let descrip = %sysfunc(dequote(&description));
     ods rtf text = "^R/RTF'\s&level.'&descrip.";

     %global globallevel sectioncount;
     %let globallevel = &level;
     %let sectioncount = %eval(&sectioncount + 1);
 %mend sectiontitle;
```

SAS MACROS

%Sectiontitle(level,description)



Headings/Levels

Chapter (level = 1):

```
%sectiontitle(level=1,description="<CHAPTER NAME>");
```

1 <CHAPTER NAME>

```
%appendixtitle(level=1,description="<APPENDIX NAME>");
```

Appendix A <APPENDIX NAME>

Section (level = 2):

```
%sectiontitle(level=2,description="<SECTION NAME>");
```

1.1 <SECTION NAME>

```
%appendixtitle(level=2,description="<APPENDIX SECTION NAME>");
```

A.1 <APPENDIX SECTION NAME>

Subsection (level = 3):

```
%sectiontitle(level=3,description="<SUBSECTION NAME>");
```

1.1.1 <SUBSECTION NAME>

Subsubsection (level = 4):

```
%sectiontitle(level=4,description="<SUBSUBSECTION NAME>");
```

1.1.1.1 <SUBSUBSECTION NAME>

Subsubsubsection (level = 5):

```
%sectiontitle(level=5,description="<SUBSUBSECTION NAME>");
```

1.1.1.1.1 <SUBSUBSUBSECTION NAME>

SAS MACROS

%Figtitle/Tabletitle(description=“”,daterange=T,bkmk=)

- Creates figure/table titles with desired information
  - By default – begins sequential numbering of tables/figures based on most recent section level
  - Description = “Figure/table description (preferably in quotation marks)”
  - Daterange = whether or not you want to include the date in the bottom row of the title
  - Bkmk = unique bookmark indicator to reference figure/table elsewhere in the text
    - see %bookmark() macro

SAS MACROS

%Figtitle/Tabletitle(description=“”,daterange=T,bkmk=)

```
%macro figtitle(level=&globallevel,description="",radars=T,daterange=T,bkmk=);
    %let descrip = %sysfunc(dequote(&description));
    %let lvl1 = %eval(&level + 1);

    %if &daterange = T %then %do;
          ...
    %end;

    %if &bkmk ^= %then %do;
        %let bmkb = {\*\bkmkstart &bkmk.};
        %let bmke = {\*\bkmkend &bkmk.};
    %end;
    %else %do;
        %let bmkb = ;
        %let bmke = ;
    %end;

    %let dot = .;

    ods rtf text = "^R/RTF'\s7'&bmkb.{Figure {\field {\*\fldinst {STYLEREF
          H&level.\\s}}}&dot.{\field{\*\fldinst{SEQ f&sectioncount. \\* ARABIC\\s
          H&level.}}}}&bmke.^S={}:
          &descrip.^n&rdrstext.%sysfunc(dequote(&program))&dater";
    %newline(l=1);
%mend figtitle;
```

SAS MACROS

%Figtitle/Tabletitle(description=“”,daterange=T,bkmk=)

```
ods rtf text = "^R/RTF'\s7'&bmkb.{Figure {\field {\*\fldinst
   {STYLEREF H&level.\\s}}}&dot.{\field{\*\fldinst{SEQ
   f&sectioncount. \\* ARABIC\\s H&level.}}}}&bmke.^S={}:
   &descrip.^n&rdrstext.%sysfunc(dequote(&program))&dater";
```

SAS MACROS

%Figtitle/Tabletitle(level=&globallevel,description=“”,daterange=T,bkmk=)

```
%tabletitle(description=“Number of Survey Completions by Drug Group",daterange=T,bkmk=);
```

Table 6.1.1.2: RADARS® System Substance Abuse Treatment Program
Number of Survey Completions by Drug Group
from 01 January 2009 through 30 June 2016

SAS MACROS

%bookmark(bkmk)

- References a unique figure/table title bookmark elsewhere in the report builder text

```
%macro bookmark(bkmk);
{\field\flddirty{\*\fldinst {REF &bkmk. \\h}}}
%mend bookmark;
```

SAS MACROS

%bookmark(bkmk)

- References a unique figure/table title bookmark elsewhere in the report builder text

```
%figtitle(description=“Figure title",radars=F,bkmk=a1234);
```

```
ods rtf text = "^R/RTF'\s11'{%bookmark(a1234) this is a test bookmark}";
```

SAS MACROS

%bookmark(bkmk)

- References a unique figure/table title bookmark elsewhere in the report builder text

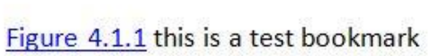


Figure 4.1.1 this is a test bookmark

SAS MACROS

%footnote(description,symbol)

- Includes a footnote in the desired location with (optional) symbol
- Applies appropriate MSWord style tag to footnotes

```
%footnote(symbol="*",description="Your footnote here");
```

* *Your footnote here*

```
%footnote(symbol=,description="Your footnote here");
```

Your footnote here

SAS MACROS

%Newline(): `ods rtf text = " ";`

%Newpage(): `ods rtf startpage = now;`

%Landscape() / %portrait(): `options orientation = landscape/portrait;`

%template(style =);

- Defines style guide borders
- Style = 1: Report style with green table in the footer
- Style = 2: Alternate style with no green table in the footer
- Assigns global macro variable &globalstyle from the input

```
%template(style=1);
```

EXAMPLE REPORT

- Example SAS code

- Example Word document

FUTURE DIRECTIONS

- **Call command prompt from SAS to**
  - automatically run R-code for plotting
  - automatically run VBA code
- **Interactive menus in Word to define styles**

REFERENCES AND RESOURCES

Funny stuff in my code using ods escapechar	Cynthia Zender. Paper 099-2007. SAS Global Forum 2007.
Enhancing RTF Output with RTF Control Words and Inline Formatting	Lori Parsons. Paper 151-2007. SAS Global Forum 2007.
Making RTF Output Pretty with SAS	Carol Matthews and Elena Kalchenko. PharmaSUG 2013 - Paper IB08. PharmaSUG 2013.

CONCLUSION

Questions?

Scott.Kreider@RMPDS.org