

Automation of reports for web publishing: Using ODS and SAS-Macro technology

Venkata Putcha, Thames Cancer Registry, Kings College London

Abstract

Cancer registries has a routinely (seasonally) produce reports. In this modern technology world, fast way of producing and publishing reports can be possible only through *Internet* or *Intranet*. SAS has a good programming features in producing customer-based reports without touching HTML programming editors.

In my talk, I will demonstrate some of the SAS programs which produces the customer based reports using the ODS and SAS-macro technology. Later these reports will be useful in web publishing. I also demonstrate some of the undocumented tricks with the above said SAS programs.

Introduction

Since the introduction of the output delivery system (ODS) in version 7 of the SAS system, production of the reports electronically has become easier with user specified formats (HTML, PDF or RTF etc.,). Population registries collect the data on day-to-day bases and create huge databases or elsewhere in the commercial world this is known as a transaction databases. One the regular time points registry produces the reports for monitoring the trends or further research purposes. Producing the reports on the transactional database are quite useful when a system has some pre-cooked version of programs (in SAS we can say macros) with pass on parameter style then one can easily publish on internet without altering the HTML document editors. The main objective of this paper is to demonstrate some of the techniques how to generate the HTML pages (PDF and CSV) pages with customized names and window titles.

Sample data

Before introducing the sample data let us define a SAS macro constants. As were shown in the *BOX-1* usually these were denoted by a symbol (%) represents the SAS macro variable. **%let** creates a SAS macros constants. For more explanation about the SAS macro variables and its definitions please refer the SAS manuals.

We begin with a sample data as shown in the following *BOX-2*. The variables *DHA*, *CANCER* and *SEX* were read as a character variables and rest were taken as a numeric variables. The description of the variables is as follows:

DHA	- District Health Authority
CR	- Crude incidence rate per 100,000 population for the year 2001
COUNT	- Number of cancer incidence cases for the year 2001

CANCER - Type of cancer
SEX - Sex

The categories of the each variable can be seen in the format statement presented in the *BOX-3*.

BOX-1

```
%let year = 2000;  
%let save = c:\output;
```

BOX-2

```
data cancer&year;  
input DHA$ CR COUNT CANCER$ SEX$;  
datalines;  
EK 65.98 260 L M  
EK 62.74 193 L F  
ES 24.17 137 L M  
ES 35.66 77 L F  
WK 25.85 313 L M  
WK 48.04 186 L F  
WS 67.01 358 L M  
WS 74.27 212 L F  
EK 31.01 121 C M  
EK 42.82 112 C F  
WK 41.25 109 C M  
WK 53.59 85 C F  
ES 30.03 167 C M  
ES 47.44 168 C F  
WS 27.31 146 C M  
WS 31.24 157 C F  
RUN;
```

BOX-3

```
PROC FORMAT;  
VALUE $DHAF  
EK = "East Kent"  
ES = "East Surrey"  
WK = "West Kent"  
WS = "West Surrey"  
;  
VALUE $SEXF  
M = "Male"  
F = "Female"  
;  
VALUE $CANCERF  
L = "Lung"  
C = "Colon"  
;  
run;
```

Report Macro

The %macro and %mend are the statements defines the SAS macro by the name specified after the statement %macro. As it were shown in the *BOX-4*. In our demonstration the report() is the macro name can be called to executive elsewhere in the program for performing the code defined. The SAS macros are very useful when a programmer wants to executive the same code for many times in different stages of the program. Later we can see how we call a defined macro.

Let us look carefully the report macro what it does? The proc report is embedded with in the macro started with a passing parameter *data*. The second line of the macro is the standard SAS report procedure. In recent version (from version 7) of SAS supports the HTML syntax like style borderwidth etc. The statement style(report)... defines the properties and its attributes can be seen in the square brackets. Similarly the next three lines define the properties and its attributes of header, column and summary. The header is for the header of the each column and the title if any. The column is for the columns defined in the report COL statement. The summary are the values printed in the cell. So this gives the programmer has a control in defining the colours, fonts and sizes of the each cell (summary) value.

BOX-4

```
%MACRO report(data);
PROC REPORT DATA=&DATA LS=100 NOWD SPLIT = '*' headline headskip
style(report)=[cellspacing=1 borderwidth=2 bordercolor=CXE3FFFF
background=CX000088]
style(header)=[foreground = CX003366 background = CXBADDFC font_face =
complex font_size = 2.5]
style(column)=[foreground=CX003366 background=CXBADDFC font_face=complex
font_size=2]
style(lines)=[foreground=CX003366 background=CXBADDFC font_face=simplex
font_weight = bold font_size=2.25]
style(summary)=[foreground=CX003366 background=CXBADDFC
font_face=complex font_size=3]
;
COL CANCER sex dha cr count;
  DEFINE CANCER / ORDER 'Cancer'*FORMAT = $CANCERF.;
  DEFINE sex / ORDER 'Gender'*FORMAT = $SEXF.;
  DEFINE dha / 'Health *Authority*' FORMAT = $DHAF. FLOW ;
    DEFINE COUNT / 'Number*' FORMAT = 7. center;
    DEFINE CR / 'Crude *rate*' FORMAT = 7.2 center;
BREAK AFTER SEX / SKIP;
COMPUTE BEFORE _PAGE_ / LEFT;
  LINE "Cancer registrations, &year, by area of residence.";
  LINE "Number of new cases and crude rate per 100000.";
  LINE " ";
ENDCOMP;
RUN;
%MEND;
```

The LINE statements between the COMPUTE BEFORE and ENDCOMP statements work as a title of the report. In normal SAS report editor TITLE and FOOTNOTE statements works better. But when one wants to send the output using the ODS CSV output TITLE and FOOTNOTE statements were ignored. So the alternative way of creating the titles are using the line statement with in the Proc Report.

BOX-5

```
%MACRO final(DATA, cancer);
data _null_;
  set &data;
  if cancer eq upcase("&cancer") then
    CALL SYMPUT("label", TRIM(LEFT(PUT(CANCER, $CANCERF.))));
run;
data _null_;
  CALL SYMPUT("h", TRIM(LEFT(PUT(".htm", $4.))));
  CALL SYMPUT("c", TRIM(LEFT(PUT(".csv", $4.))));
  CALL SYMPUT("p", TRIM(LEFT(PUT(".pdf", $4.))));
  CALL SYMPUT("s", TRIM(LEFT(PUT("_", $1.))));
run;
data final;
  set &data;
  if cancer = upcase("&cancer") then output;
run;

/* ----- html ----- */
FILENAME REP "&save";
ods html body="&label&s&YEAR&h" (title="Incidence and counts of &label,
&YEAR")
path=REP nogtitle style=customdefault nogtitle;
%report(final);
ods html close;

/* ----- pdf ----- */
OPTIONS ORIENTATION=portrait NODATE NONUMBER;
GOPTIONS RESET = ALL FTEXT="helvetica" device = pdfc;
ods PDF FILE="&save\&label&s&YEAR&p" NOTOC ;
ods listing close;
%report(final);
ods pdf close;

/* ----- csv ----- */
ods csv file="&label&s&YEAR&c"
path=REP nogtitle style=customdefault;
%report(final);
ods csv close;
quit;
%mend;
```

The final macro %final is defined in the *Box-5*. At the beginning of the macro we initialized some macro constants using the SAS supplied *CALL SYMPUT()*. This command allows us to define a macro constant which are dynamically changes from the format values. Like I am interested to check on the bases of cancer type (LUNG or COLON) and assign it to a constant macro name "label". Now this assigned "label-name will be called with &label in the report output – titles or filenames etc. This is what we called in every ODS statement filename creation (HTML, PDF, and CSV) along with other macros constant variables. That creates flexibility in an individually output and later one can link with JAVA or ASP script. For real example the reader is advised to view www.tcr.org.uk/data. The full program is listed in the following appendix along with the output snapshots. User are free to share the program and have a fun.

Appendix

```
/* views conference paper example */
```

```
/* example data set */
```

```
%let year = 2000;
%let save = c:\output;
data cancer&year;
    input DHA$ CR COUNT cancer$ sex$;
    datalines;
EK 65.98 260 L M
EK 62.74 193 L F
ES 24.17 137 L M
ES 35.66 77 L F
WK 25.85 313 L M
WK 48.04 186 L F
WS 67.01 358 L M
WS 74.27 212 L F
EK 31.01 121 C M
EK 42.82 112 C F
WK 41.25 109 C M
WK 53.59 85 C F
ES 30.03 167 C M
ES 47.44 168 C F
WS 27.31 146 C M
WS 31.24 157 C F
RUN;

PROC FORMAT;
    VALUE $DHAF
        EK = "East Kent"
        ES = "East Surrey"
        WK = "West Kent"
        WS = "West Surrey"
    ;
    VALUE $SEXF
        M = "Male"
        F = "Female"
    ;
```

```

VALUE $CANCERF
    L = "Lung"
    C = "Colon"
;
run;

PROC TEMPLATE;
    define style customdefault;
        parent=styles.default;
        replace color_list /
            'bgA' = CXBADDFC;
    end;
run;

%MACRO report(data);
TITLE;
FOOTNOTE;
PROC REPORT DATA=&DATA LS=100 NOWD SPLIT = '*' headline headskip
style = sasweb
    style(report)=[cellspacing=1 borderwidth=2
bordercolor=CXE3FFFF background=CX000088]
    style(header)=[foreground=CX003366 background=CXBADDFC
font_face=complex font_size=2.5]
    style(column)=[foreground=CX003366 background=CXBADDFC
font_face=complex font_size=2]
    style(lines)=[foreground=CX003366 background=CXBADDFC
font_face=simplex font_weight = bold font_size=2.25]
    style(summary)=[foreground=CX003366 background=CXBADDFC
font_face=complex font_size=3]
;
    COL CANCER sex dha cr count;
    DEFINE CANCER / ORDER 'Cancer*' FORMAT = $CANCERF.;
    DEFINE sex / ORDER 'Gender*' FORMAT = $SEXF.;
    DEFINE dha / 'Health *Authority*' FORMAT = $DHAF. FLOW ;
    DEFINE COUNT / 'Number*' FORMAT = 7. center;
    DEFINE CR / 'Crude *rate*' FORMAT = 7.2 center;
BREAK AFTER SEX / SKIP;
COMPUTE BEFORE _PAGE_ / LEFT;
LINE "Cancer registrations, &year, by area of residence.";
LINE "Number of new cases and crude rate per 100000.";
LINE " ";
ENDCOMP;
RUN;
%MEND;
%MACRO final(DATA, cancer);
data _null_;
    set &data;
    if cancer eq upcase("&cancer") then
        CALL SYMPUT("label", TRIM(LEFT(PUT(CANCER, $CANCERF.))));
run;
data _null_;
    CALL SYMPUT("h", TRIM(LEFT(PUT(".htm", $4.))));
run;
data _null_;
    CALL SYMPUT("c", TRIM(LEFT(PUT(".csv", $4.))));
run;
data _null_;
    CALL SYMPUT("p", TRIM(LEFT(PUT(".pdf", $4.))));
run;
data _null_;
    CALL SYMPUT("s", TRIM(LEFT(PUT("_", $1.))));

```

```

run;

data final;
  set &data;
  if cancer = upcase("&cancer") then output;
run;

/* ----- html ----- */
FILENAME REP "&save";
ods html body="&label&s&YEAR&h" (title="Incidence and counts of
&label, &YEAR")
path=REP nogtitle style=customdefault nogtitle;
%report(final);
ods html close;

/* ----- pdf ----- */
OPTIONS ORIENTATION=portrait NODATE NONUMBER;
GOPTIONS RESET = ALL FTEXT="helvetica" device = pdfc;
ods PDF FILE="&save\&label&s&YEAR&p" NOTOC ;
ods listing close;
%report(final);
ods pdf close;

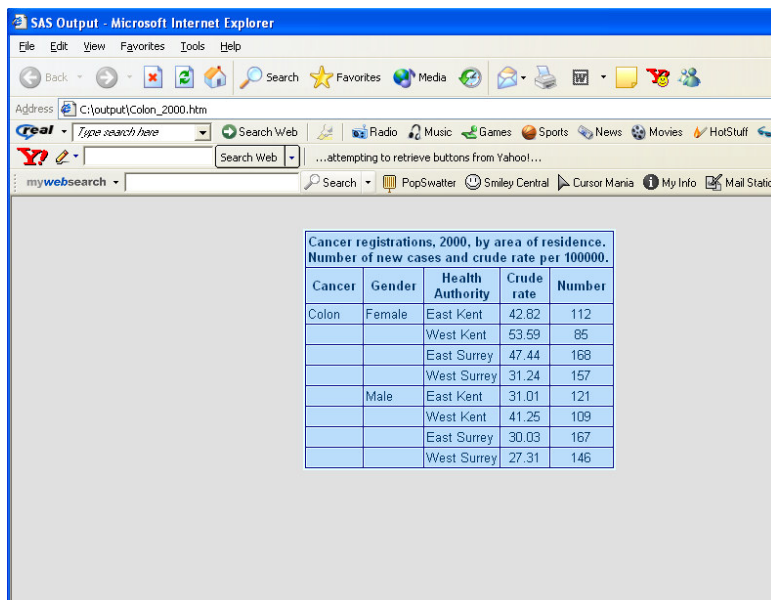
/* ----- csv ----- */
ods csv file="&label&s&YEAR&c"
path=REP nogtitle style=customdefault;
%report(final);
ods csv close;
quit;
%mend;

%final(cancer2000, 1);
%final(cancer2000, C);
quit;

```

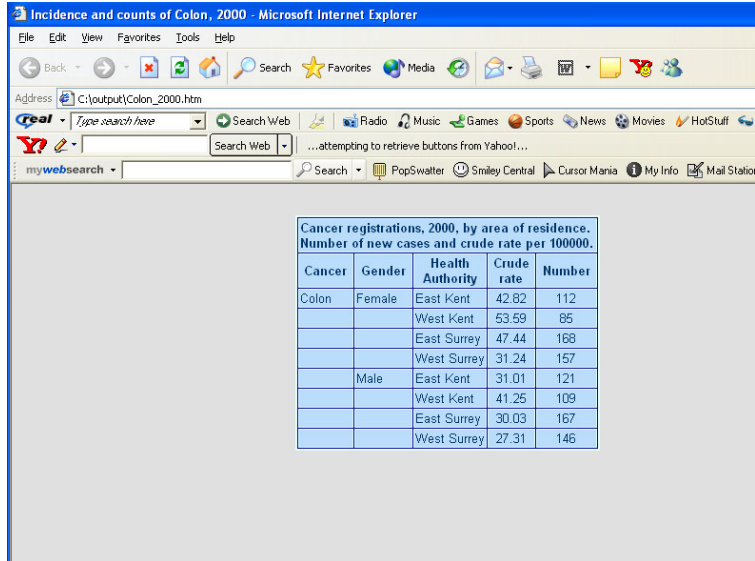
***** OUTPUTS *****

HTML OUTPUT



Cancer	Gender	Health Authority	Crude rate	Number
Colon	Female	East Kent	42.82	112
		West Kent	53.59	85
		East Surrey	47.44	168
		West Surrey	31.24	157
	Male	East Kent	31.01	121
		West Kent	41.25	109
		East Surrey	30.03	167
		West Surrey	27.31	146

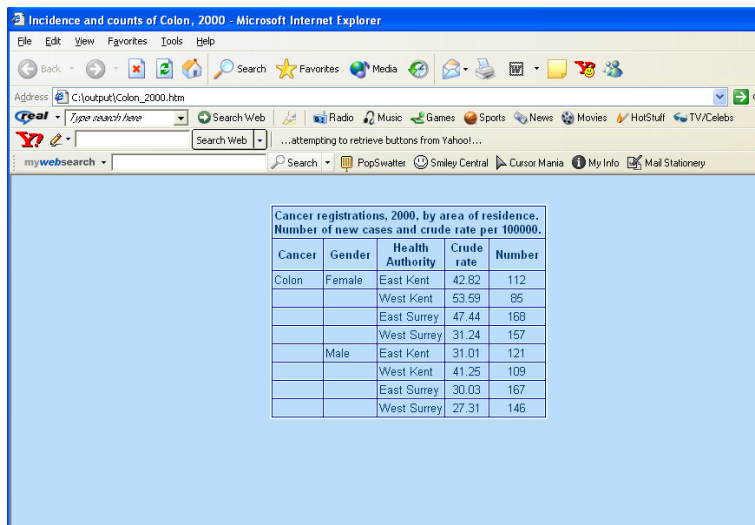
HTML OUTPUT with title in the window



The screenshot shows a Microsoft Internet Explorer window titled "Incidence and counts of Colon, 2000 - Microsoft Internet Explorer". The address bar displays "C:\output\Colon_2000.htm". The page content features a table with the following data:

Cancer	Gender	Health Authority	Crude rate	Number
Colon	Female	East Kent	42.82	112
		West Kent	53.59	85
		East Surrey	47.44	168
		West Surrey	31.24	157
	Male	East Kent	31.01	121
		West Kent	41.25	109
		East Surrey	30.03	167
		West Surrey	27.31	146

HTML OUTPUT with title and background color



The screenshot shows a Microsoft Internet Explorer window titled "Incidence and counts of Colon, 2000 - Microsoft Internet Explorer". The address bar displays "C:\output\Colon_2000.htm". The page content features a table with the following data, set against a light blue background:

Cancer	Gender	Health Authority	Crude rate	Number
Colon	Female	East Kent	42.82	112
		West Kent	53.59	85
		East Surrey	47.44	168
		West Surrey	31.24	157
	Male	East Kent	31.01	121
		West Kent	41.25	109
		East Surrey	30.03	167
		West Surrey	27.31	146

PDF output

Cancer registrations, 2000, by area of residence.
Number of new cases and crude rate per 100000.

Cancer	Gender	Health Authority	Crude rate	Number
Colon	Female	East Kent	42.82	112
		West Kent	53.59	85
		East Surrey	47.44	168
		West Surrey	31.24	157
	Male	East Kent	31.01	121
		West Kent	41.25	109
		East Surrey	30.03	167
		West Surrey	27.31	146

CSV output with titles using the Line statements

Microsoft Excel - Colon_2000

	A	B	C	D	E	F	G
1							
2							
3	Cancer registrations, 2000, by area of residence.						
4	Number of new cases and crude rate per 100000.						
5							
6	Cancer	Gender	Health Authority	Crude rate	Number		
7	Colon	Female	East Kent	42.82	112		
8			West Kent	53.59	85		
9			East Surrey	47.44	168		
10			West Surrey	31.24	157		
11		Male	East Kent	31.01	121		
12			West Kent	41.25	109		
13			East Surrey	30.03	167		
14			West Surrey	27.31	146		
15							
16							
17							

Contact Information

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

Venkata Putcha
Thames Cancer Registry
Kings College London
United Kingdom, SE1 3QD
Email Venkata.putcha@kcl.ac.uk