\*-------------------------------------------------------\*

\* PROGRAMMER: Liz Uribe \*

\* \*

\* DATE CREATED: March 25, 2014 \*

\* \*

\* FILE NAME: Print macro example for talk \*

\* \*

\* DESCRIPTION: Uses sashelp.heart to create an example \*

\* demographics table using the prntinit macro \*

\* \*

\* NOTES: \*

\* \*

\* INPUT DATA SETS: \*

\* \*

\* OUTPUT DATA SETS: \*

\*-------------------------------------------------------\*;

%let output\_path = H:\work\print macro example for talk;

data heart;

 set sashelp.heart;

 age = ageatstart;

run;

proc sql noprint;

 select count(status)

 into: total separated by ''

 from heart ;

quit;

proc means data = heart mean std min max nmiss;

var age height weight cholesterol diastolic systolic ;

output out = summary mean = mean\_age mean\_ht mean\_wt mean\_chol mean\_dia mean\_sys

 std = std\_age std\_ht std\_wt std\_chol std\_dia std\_sys

 min = min\_age min\_ht min\_wt min\_chol min\_dia min\_sys

 max = max\_age max\_ht max\_wt max\_chol max\_dia max\_sys

 nmiss = nmiss\_age nmiss\_ht nmiss\_wt nmiss\_chol nmiss\_dia nmiss\_sys ;

run;

data summary2 (drop=mean: std: min\_: max: nmis: \_TYPE\_);

 set summary;

 length avestd\_age avestd\_ht avestd\_wt avestd\_chol avestd\_dia avestd\_sys $12.;

 array mean {\*} mean\_age mean\_ht mean\_wt mean\_chol mean\_dia mean\_sys ;

 array std {\*} std\_age std\_ht std\_wt std\_chol std\_dia std\_sys ;

 array avestd {\*} $ avestd\_age avestd\_ht avestd\_wt avestd\_chol avestd\_dia avestd\_sys ;

 array min {\*} min\_age min\_ht min\_wt min\_chol min\_dia min\_sys ;

 array max {\*} max\_age max\_ht max\_wt max\_chol max\_dia max\_sys ;

 array minmax {\*} $10. minmax\_age minmax\_ht minmax\_wt minmax\_chol minmax\_dia minmax\_sys;

 array nmis {\*} nmiss\_age nmiss\_ht nmiss\_wt nmiss\_chol nmiss\_dia nmiss\_sys ;

 array miss {\*} $ miss\_age miss\_ht miss\_wt miss\_chol miss\_dia miss\_sys ;

 do i=1 to dim(mean);

 avestd[i] = strip(put(mean[i], 5.2)) || " (" || strip(put(std[i], 5.1)) || ")";

 minmax[i] = "(" || strip(put(min[i],5.)) || ", " || strip(put(max[i],5.)) || ")";

 miss[i] = strip(put(nmis[i], 3.));

 end;

run;

proc transpose data=summary2 out=summary3;

var avestd\_age minmax\_age miss\_age

 avestd\_ht minmax\_ht miss\_ht

 avestd\_wt minmax\_wt miss\_wt

 avestd\_chol minmax\_chol miss\_chol

 avestd\_dia minmax\_dia miss\_dia

 avestd\_sys minmax\_sys miss\_sys;

run;

data summary4;

 set summary3;

 if \_N\_ < 13 then i = \_N\_+3;

 else i = \_N\_ + 12;

run;

\*categorical stats;

%macro transpose(var);

 proc freq data=heart;

 tables &var / missing out = freq\_&var;

 run;

 data freq2\_&var (keep= &var COL1);

 set freq\_&var;

 COL1 = strip(put(COUNT,5.)) || " (" || strip(put(PERCENT,5.1)) || "%)";

 if missing(&var) then &var = "Missing";

 run;

%mend;

%transpose(status); %transpose(sex); %transpose(smoking\_status);

data status;

 set freq2\_status;

 i = \_N\_;

run;

data gender;

 set freq2\_sex;

 i = \_N\_ + 15;

run;

data smoke;

 set freq2\_smoking\_status;

 if Smoking\_status = "Non-smoker" then i = 19;

 else if Smoking\_status = "Light (1-5)" then i = 20;

 else if Smoking\_status = "Moderate (6-15)" then i = 21;

 else if Smoking\_status = "Heavy (16-25)" then i = 22;

 else if Smoking\_status = "Very Heavy (> 25)" then i = 23;

 else if Smoking\_status ="Missing" then i = 24;

run;

proc sort data = smoke; by i; run;

data all;

 set summary4 status gender smoke;

 by i;

 newtable = 1;

run;

\*Cause of death for those who died;

proc sql noprint;

 select count(status)

 into: dead separated by ''

 from heart

 where status = "Dead";

quit;

proc freq data = heart(where = (status = "Dead"));

 tables deathcause / out = freq\_deathcause outpct sparse;

run;

data freq2\_deathcause (keep = deathcause n\_per newtable i);

 set freq\_deathcause;

 n\_per = strip(put(COUNT,5.)) || " (" || strip(put(PERCENT,5.1)) || "%)";

 i = \_N\_;

 newtable = 2;

run;

data all2(drop=\_NAME\_ sex smoking\_status status deathcause);

 set all freq2\_deathcause;

 by newtable i;

run;

\*combine everything;

data labels;

input newtable i shortname $42.;

cards;

1 1 Alive

1 2 Dead

1 3 Missing

1 4 Mean (Std)

1 5 (Min, Max)

1 6 Missing

1 7 Mean (Std)

1 8 (Min, Max)

1 9 Missing

1 10 Mean (Std)

1 11 (Min, Max)

1 12 Missing

1 13 Mean (Std)

1 14 (Min, Max)

1 15 Missing

1 16 Female

1 17 Male

1 18 Missing

1 19 Non-Smoker

1 20 Light (1-5)

1 21 Moderate (6-15)

1 22 Heavy (16-25)

1 23 Very Heavy (> 25)

1 24 Missing

1 25 Mean

1 26 (Min, Max)

1 27 Missing

1 28 Mean

1 29 (Min, Max)

1 30 Missing

2 1 Cancer

2 2 Cerebral Vascular Disease

2 3 Coronary Heart Disease

2 4 Other

2 5 Unknown

;

run;

data headers;

input newtable i name $42.;

cards;

1 0.5 Status

1 3.5 Age

1 6.5 Height

1 9.5 Weight

1 12.5 Cholesterol

1 15.5 Gender

1 18.5 Smoking Status

1 24.5 Diatolic Blood Pressure

1 27.5 Systolic Blood Pressure

;

run;

data combine ;

 merge labels headers all2;

 by newtable i;

 if missing(name) then name = ' ' || strip(shortname);

 if not missing(shortname) and newtable = 1 then do;

 if missing(col1) then col1 = "0 (0%)";

 end;

run;

options nocenter nodate;

%INCLUDE 'K:\DMC\SAS LIBRARY\PRINTDRV\PRNTINIT.SAS';

data \_null\_;

set combine end = eof;

by newtable i;

%PRNTINIT(DRIVER=RTF,PRINT="&output\_path\Example demographics table.rtf",

 FONT=ARIAL,FONTSIZE=8, ORIENT=portrait, LS=340,PS=85,TJUST=LEFT,MARGINS=LEFT:1);

if newtable = 1 then do;

 if first.newtable then do;

 %print("Example Demographics Table", just = left, style = bold);

 %print(2, proc = newline);

 %print(proc = table, column = 1 60 90, lines = all:single);

 %print(proc = newrow);

 %print("Variable", column = 1, style = bold, just = center, lines = bottom:none);

 %print("Subjects", column = 2, style = bold, just = center, lines = bottom:none);

 %print(proc = newrow);

 %print( column = 1);

 %print("(N = &total)", column = 2, just = center);

 %print(proc = newrow);

 end;

 if not last.newtable then do;

 if missing(col1) then do;

 %print(name, column = 1, style = bold, lines = top:single);

 %print( column = 2, just = center, lines = top:single);

 %print(proc = newrow);

 end;

 else if not missing(col1) then do;

 %print(name, column = 1, lines = bottom:none);

 %print(col1, column = 2, just = center, lines = bottom:none);

 %print(proc = newrow);

 end;

 end;

 if last.newtable then do;

 %print(name, column = 1);

 %print(col1, column = 2, just = center);

 %print(proc = endtable);

 %print(proc = newline);

 %print("Comments can be added");

 %print(3, proc = newline);

 end;

end;

if newtable = 2 then do;

 if first.newtable then do;

 %print("For Those who Died: Causes of Death", just = left, style = bold);

 %print(2, proc = newline);

 %print(proc = table, column = 1 60 90, lines = all:single);

 %print(proc = newrow);

 %print("Cause of Death", column = 1, style = bold, just = center, lines = bottom:none);

 %print("Count", column = 2, style = bold, just = center, lines = bottom:none);

 %print(proc = newrow);

 %print( column = 1);

 %print("(N = &dead)", column = 2, just = center);

 %print(proc = newrow);

 end;

 if not eof then do;

 %print(name, column = 1, lines = bottom:none);

 %print(n\_per, column = 2, just = center, lines = bottom:none);

 %print(proc = newrow);

 end;

 if eof then do;

 %print(name, column = 1);

 %print(n\_per, column = 2, just = center);

 %print(proc = endtable);

 %print(proc = newline);

 %print("Comments can be added");

 %prntstop;

 end;

end;

run;