Analyze data set:

```
%macro analyzeSASdsn(dsn,numobs,varlist);

%let numvars=; /* Initialize the number of numeric Variables */

%let charvars=; /* Initialize the number of character Variables */

%if %upcase(&varlist) eq _ALL_ %then /* If the user request for all the
variables then

assign appropriately */

%do;

%let numvars=_numeric_;

%let charvars=_character_;

%end;

%else /* then find if the vars requested are numeric type or character type
*/

%do;

/* split the varlist into individual macro var names vars1 vars2 etc*/

%let num=1;

%let vars&num=%scan(&varlist,&num,' ');

%do %while(&&vars&num ne );

%let num=%eval(&num + 1);

%let vars&num=%upcase(%scan(&varlist,&num,' '));

%end;

/* Get the List of variables in the &dsn dataset and put

All char variables in charvarlist macro variables

All Num variables in numvarlist macro variable

*/

%let dsid=%sysfunc(open(&dsn,i));

%let numvarlist=;
```

```
%let charvarlist=;

%do i=1 %to %sysfunc(attrn(&dsid,nvars));

%if (%sysfunc(vartype(&dsid,&i)) = N) %then %let numvarlist=&numvarlist

%upcase(%sysfunc(varname(&dsid,&i)));

%if (%sysfunc(vartype(&dsid,&i)) = C) %then %let charvarlist=&charvarlist

%upcase(%sysfunc(varname(&dsid,&i)));

%end;

%let rc=%sysfunc(close(&dsid));

%put numvarlist=&numvarlist charvarlist=&charvarlist;

/* Now check the variables required to report in the above list and assign
them to

the right macro variables...

All char variables in charvarlist macro variables

All Num variables in numvarlist macro variable

*/

%do i=1 %to %eval(&num - 1);

%if %index(&numvarlist,&&vars&i) %then %let numvars=&&vars&i &numvars;

%if %index(&charvarlist,&&vars&i) %then %let charvars=&&vars&i &charvars;

%end;

%put numvars=&numvars charvars=&charvars;

%end;

ods listing close;

ods html body="&htmlfilepath";

/* Now analyze the dataset with the Specified variables */

proc contents data=&dsn;run; /* Put a Contents procedure */

%if &numvars ne %then
```

```
%do;

/* Get Summary statistics of All the Numeric Variables with means procedure
*/

proc means data=&dsn(obs=&numobs) n mean max min range;

var &numvars;

title 'Summary Statistics of all Numeric variables in the dataset';

run;

%end;

%if &charvars ne %then

%do;

/* Get Summary statistics of All the Character Variables with Freq procedure
*/

proc freq data=&dsn(obs=&numobs);

tables &charvars;

title1 'Summary Statistics of all Character variables in the dataset';

run;

%end;

ods html close;

%mend analyzeSASdsn;



%let htmlfilepath=C:\Users\veera\Desktop\SAS2;



options nodate pageno=1 linesize=80 pagesize=60 mprint symbolgen;
```

```
*-------------------------------------------------------------*;



* MACRO PARAMETERS:;
* metadatafile = the MS Excel file containing the dataset metadata;
* dataset = the dataset or domain name you want to extract;

*-------------------------------------------------------------*;
%macro smake_empty_dataset(metadatafile=, dataset=);

/*
   proc import
      datafile="&metadatafile"
      out=_temp
      dbms=excelcs
      replace;
      sheet="VARIABLE_METADATA";
   run;
*/

   ** sort the dataset by expected specified variable order;
   proc sort
    data=svariable_metadata out= _temp;
                                                    where domain =
                                                    "&dataset";
      by varnum;
   run;

   ** create keepstring macro variable and load metadata
   ** information into macro variables;
   %global &dataset.KEEPSTRING;

   data _null_;
    set _temp nobs=nobs end=eof;

      if _n_=1 then
       call symput("vars", compress(put(nobs,3.)));

      call symputx('var'    || compress(put(_n_, 3.)), variable);
      call symputx('label'  || compress(put(_n_, 3.)), label);
      call symputx('length' || compress(put(_n_, 3.)),
                               put(length, 3.));
```

```
    ** valid ODM types include TEXT, INTEGER, FLOAT, DATETIME,
    ** DATE, TIME and map to SAS numeric or character;
    if upcase(type) in ("INTEGER", "FLOAT") then
     call symputx('type' || compress(put(_n_, 3.)), "");
    else if upcase(type) in ("TEXT", "DATE", "DATETIME",
                         "DATE", "TIME") then
     call symputx('type' || compress(put(_n_, 3.)), "$");
    else
     put "ERR" "OR: not using a valid ODM type.  " type=;


    ** create **KEEPSTRING macro variable;
    length keepstring $ 32767;
    retain keepstring;
    keepstring = compress(keepstring) || "|" || left(variable);
    if eof then
     call symputx(upcase(compress("&dataset" || 'KEEPSTRING')),
               left(trim(translate(keepstring," ","|"))));
  run;


  ** create a 0-observation template data set used for assigning
  ** variable attributes to the actual data sets;
  data EMPTY_&dataset;
    %do i=1 %to &vars;
      attrib &&var&i label="&&label&i" length=&&type&i.&&length&i...
      ;
      %if &&type&i=$ %then
       retain &&var&i '';
      %else
       retain &&var&i .;
      ;
    %end;
    if 0;
  run;

%mend smake_empty_dataset;
```