**Workshop3-B**

**Book: Learning SAS by Example: A programmer's Guide**

**Section 9.11: Problems: 6,8,10**

**Section 11.13: Problems: 8,10,12**

**Section 12.17: Problems: 8,10,12**

**Question 1**

\* 9.11-6 Program to compute frequencies by supplying a format. ;  
  
/\*Using the Medical data set, compute frequencies for the days of the week for the date of the visit  
(VisitDate). Supply a format for the days of the week and months of the year.  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
  
data days;  
 set pract.medical(keep=VisitDate);  
 Day = weekday(VisitDate);  
 Month = month(VisitDate);  
run;  
  
proc format;  
 value days 1='Sunday' 2='Monday' 3='Tuesday' 4='Wednesday'  
 5='Thursday' 6='Friday' 7='Saturday';  
   
 value months 1='Jan' 2='Feb' 3='Mar'  
 4='Apr' 5='May' 6='Jun'  
 7='Jul' 8='Aug' 9='Sep'  
 10='Oct' 11='Nov' 12='Dec';  
run;  
  
title "Frequencies for days and months of VisitDate :";  
proc freq data=days;  
 tables Day Month / nocum nopercent;  
 format Day days. Month months.;  
run;

**Question 2**

\* 9.11-8 Program to create dataset according to given format.;  
  
/\*Using the values for Day, Month, and Year in the raw data below, create a temporary SAS data set  
containing a SAS date based on these values (call it Date) and format this value using the  
MMDDYY10. format. Here are the Day, Month, and Year values:  
  
25 12 2005  
1 1 1960  
21 10 1946  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
data pract.date;  
input Day Month Year;  
datalines;  
25 12 2005  
1 1 1960  
21 10 1946  
;  
  
data pract.dates;  
set pract.date;  
Date = mdy(month,day,year);  
run;  
  
title "Formatted date :";  
proc print data=pract.dates;  
format Date mmddyy10.;  
run;

**Question 3**

\* 9.11-10 Program to compute no. of months as per given criteria. ;

/\*Using the Hosp data set, compute the number of months from the admission date (AdmitDate) and

December 31, 2007 (call it MonthsDec). Also, compute the number of months from the admission

date to today's date (call it MonthsToday). Use a date interval function to solve this problem. List

the first 20 observations for your solution.

\*/

libname pract '/home/u58712040/Programming\_Workshops';

data hospitals;

set pract.hosp;

MonthsDec = intck ('month',AdmitDate,'31dec2007'd);

MonthsToday = intck ('month',AdmitDate,today());

run;

title "Months using data interval function :";

proc print data=hospitals(obs=20);

run;

**Question 4**

\* 11.13-8 Program to create a random data set as per given conditions. ;  
  
/\*Create a temporary SAS data set (Random) consisting of 1,000 observations, each with a random  
integer from 1 to 5. Make sure that all integers in the range are equally likely. Run PROC FREQ  
to test this assumption.  
\*/  
libname pract '/home/u58712040/Programming\_Workshops';  
  
data pract.random;  
do i=1 to 1000;  
x=int(rand('uniform')\*5)+1;  
output;  
end;  
run;  
  
title 'Random numbers from 1 to 5';  
proc print data=pract.random(obs=10);  
run;  
  
title 'Testing details';  
proc freq data=pract.random;  
table x;  
run;

**Question 5**

\* 11.13-10 Program to create data set that has numeric to character values and vice versa. ;  
  
/\*Data set Char\_Num contains character variables Age and Weight and numeric variables SS and  
Zip. Create a new, temporary SAS data set called Convert with new variables NumAge and  
NumWeight that are numeric values of Age and Weight, respectively, and CharSS and CharZip  
that are character variables created from SS and Zip. CharSS should contain leading 0s and dashes  
in the appropriate places for Social Security numbers and CharZip should contain leading 0s.  
Hint: The Z5. format includes leading 0s for the ZIP code.  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
data pract.Convert;  
 set pract.char\_num;  
 NumAge = input(Age,5.);  
 NumWeight = input(Weight,5.);  
 CharSS = put(SS,ssn11.);  
 CharZip = put(Zip,z5.);  
run;  
   
title "Properly Formatted values of variables";  
proc print data=pract.Convert;   
run;

**Question 6**

\* 11.13-12 Program to compute daily changes of stocks and plot them. ;

/\*Using the Stocks data set (containing variables Date and Price), compute daily changes in the

prices. Use the statements here to create the plot.

title "Plot of Daily Price Differences";

proc sgplot data=Difference;

series x=Date y=Date;

run;

\*/

libname pract '/home/u58712040/Programming\_Workshops';

data pract.stock\_diff;

set pract.stocks;

Diff\_Stock\_Price\_by\_lag = Price - lag(Price);

Diff\_Stock\_Price\_by\_dif = Dif(Price);

run;

\*both of the two variables computed above are correct, just done using 2 different ways.;

title "Plot of Daily Price Differences";

proc sgplot data=pract.stock\_diff;

series x=Date y=Diff\_Stock\_Price\_by\_lag;

run;

**Question 7**

\* 12.17-8 Program to round values. ;  
  
/\*Notice in the listing of data set Study in Problem 6 that the variable called Weight contains units  
(either lbs or kgs). These units are not always consistent in case and may or may not contain a  
period. Assume an upper- or lowercase LB indicates pounds and an upper- or lowercase KG  
indicates kilograms. Create a new, temporary SAS data set (Study) with a numeric variable also  
called Weight (careful here) that represents weight in pounds, rounded to the nearest 10th of a  
pound.  
  
Note: 1 kilogram = 2.2 pounds.  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
data pract.Study1;  
 set pract.study (keep=Weight rename=(Weight=Weight\_with\_units));  
 Weight=input(compress(Weight\_with\_units,,'kd'),8.);  
 if find(Weight\_with\_units,'KG','i') then Weight=round(2.2\*Weight,.1);  
 else if find(Weight\_with\_units,'LB','i') then Weight=round(Weight,.1);  
run;  
  
\*In func compress(weight\_with\_units,,'kd') kd is used to   
keep only digits from string.  
Then find func is used in lines 18 and 19 giving 'i' as an argument to ignore  
case and round func is used to round the values of weight.;  
  
title "Rounded values for weight";   
proc print data=pract.Study1;  
run;

**Question 8**

\* 12.17-10 Program to create dataset as per given conditions for dataset Errors. ;  
  
/\*Data set Errors contains character variables Subj (3 bytes) and PartNumber (8 bytes). (See the partial listing here.) Create a temporary SAS data set (Check1) with any observation in Errors that  
violates either of the following two rules: first, Subj should contain only digits, and second, PartNumber should contain only the uppercase letters L and S and digits.  
  
Here is a partial listing of Errors:  
Listing of Data Set Learn.Errors  
  
 Part  
Subj Number Name  
001 L1232 Nichole Brown  
0a2 L887X Fred Beans  
003 12321 Alfred 2 Nice  
004 abcde Mary Bumpers  
X89 8888S Gill Sandford  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
data pract.Check1;  
 set pract.errors;  
 where notdigit(trim(Subj)) OR verify(trim(PartNumber),'0123456789LS');  
run;  
  
\*Here as we are checking obs that violate rules, so we have given func  
 notdigit which will check for strings that do not have digit and return value.  
 trim() is used inside notdigit() as without it notdigit() will just return the   
 1st position of the string where char value will be found.  
 verify() is used by giving the substring as an argument for which we are checking   
 the 2nd rule. This () returns the 1st char in the source which is not present in given substring.   
   
title"Observations that violate given 2 rules :";  
proc print data=pract.Check1;  
run;

**Question 9**

\* 12.17-12 Program to list subject number from Errors dataset as per given criteria. ;  
  
/\*List the subject number (Subj) for any observations in Errors where PartNumber  
contains an upper- or lowercase X or D.  
\*/  
  
libname pract '/home/u58712040/Programming\_Workshops';  
  
title "Subject no. where PartNumber has x or d in any case:";  
proc print data=pract.errors noobs;  
where findc(PartNumber,'XD','i');  
run;  
  
\*findc() returns the 1st instance of given substring i.e. xd her, from   
the PartNumber. 'i' is used to ignore case.