#1 DATES

The following data are similar to data in example 8.3 in the notes.

```plaintext
data admits;
format admit mmddyy10.;
input admit1 : mmddyy10. @@;
datalines;
11181998 12111998 02281998 02161998 02271998
05071998 05101998 06031998 08021998 08131998
01081997 07251997 02041997 08171997 03071997
11161997 03281997 09301997 12271997 04031997
;
run;
```

A/ Use the data set ADMITS created by the above data step to create a two new data sets: one that contains only those observations from July 1, 1997 through December 31, 1997; another that contains observations from January 1, 1998 through June 30, 1998. (HINT: Write a new data step, use a SET statement to read data set ADMITS, use date constants or functions (e.g. MDY) and OUTPUT to create new data sets.)

B/ Use data set ADMITS to find the most common day of the week for admission. (HINT: Use a function or format.)

C/ Similar to B... find the most common day of the week for admission in 1997 and 1998.

```plaintext
*** ANSWER;

data admits;
format admit mmddyy10.;
input admit : mmddyy10. @@;
datalines;
11181998 12111998 02281998 02161998 02271998
05071998 05101998 06031998 08021998 08131998
01081997 07251997 02041997 08171997 03071997
11161997 03281997 09301997 12271997 04031997
;
run;
```

```plaintext
*** part A;
data before on_after;
set admits;
if admit ge '01jul1997'd and admit le '31dec1997'd then output before;
else
  if admit ge '01jan1998'd and admit le '30jun1998'd then output on_after;
run;

*** part B;
title 'ADMITS BY DAY OF WEEK';
proc freq data=admits order=freq;
  table admit;
  format admit downname.;
run;
```

```plaintext
*** part C;
title 'ADMITS BY DAY OF WEEK - 1997';
proc freq data=admits order=freq;
  table admit;
  format admit downname.;
  where year(admit) eq 1997;
run;
```
title 'ADMITS BY DAY OF WEEK - 1998';
proc freq data=admits order=freq;
table admit;
format admit downame.;
where year(admit) eq 1998;
run;

*** to get answer in one table, add a variable to the data set - YEAR;

data admits;
set admits;
year = year(admit);
run;

title 'ADMITS BY DAY OF WEEK AND YEAR';
proc freq data=admits order=freq;
table admit*year/norow nocol nopercent;
format admit downame.;
run;

title;

#2 DATES

Here some data for people participating in a study....

data study;
input id : $5.  firstvis : mmddyy8.;
datalines;
12345 01092001
23456 10152001
34567 07062001
;
run;

Each observation has a date that represents their first visit to your clinic. Each person in the study is supposed to come back for their next visit in 6 months (assume that means 180 days). Write SAS code that allows you to tell each person the date that they should make their next visit to your clinic.

See if you can add SAS code that does this too...

If the follow up visit is to occur on a Saturday, change the date to the preceding Friday.
If the follow up visit is to occur on a Sunday, change the date to the following Monday.

*** answer;

data study;
input id : $5.  firstvis : mmddyy8.;
datalines:
12345 01092001
23456 10152001
34567 07062001
;
run;

data revisit;
set study;
*** come back in 180 days;
nextvis = firstvis + 180;

*** if NEXTVIS = 1 (Sunday), come back Monday
*** if NEXTVIS = 7 (Saturday), come back Friday;
dow = weekday(nextvis);
if dow eq 1 then reschedule = nextvis + 1;
else
if dow eq 7 then reschedule = nextvis - 1;

format firstvis nextvis reschedule weekdate.;
drop dow;
run;
#3 DATES (FIRST. and LAST. variables)...

(from assignment #2) There is a data file available in the computer lab named... CLINICAL.DAT
It contains the following data...

column | data
1-2    | patient ID
3-10   | date of visit (stored as mm/dd/yy)
11     | drug or placebo group (D or P)
12-14  | cholesterol
15-17  | systolic bp
18-20  | diastolic bp
21-22  | heart rate
23     | routine visit, yes or no (Y or N)

A/ Use FIRST. and LAST. variables to create a data set that contains new variables...the change from first visit to last visit in cholesterol, both blood pressures, and heart rate for each person (as indicated by patient ID), patient ID.

B/ Use the new data set to compute the mean change in cholesterol, both blood pressures, and heart rate in the DRUG and PLACEBO groups. (HINT: PROC MEANS with VAR and CLASS statements).

C/ Can you come up with a way to add the number of visits (per person) to each observation of the data set you created in part A?

*** ANSWERS;

data clinical;
infile 'f:\sasclass\data\clinical.dat';
input
id      $  1-2
visit     mmddyy8.
group    $  11
chol     12-14
sbp      15-17
dbp      18-20
hr       21-22
typevis  $  23
;
bpratio = sbp / dbp;
format visit mmddyy10.;
label
id      = 'PERSONAL IDENTIFIER'
group   = 'STUDY GROUP'
visit   = 'DATE OF VISIT'
chol    = 'CHOLESTEROL'
sbp     = 'SYSTOLIC BLOOD PRESSURE'
dbp     = 'DIASTOLIC BLOOD PRESSURE'
hr      = 'HEART RATE'
typevis = 'TYPE OF VISIT'
bpratio = 'SYSTOLIC/DIASTOLIC'
;
run;

proc sort data=clinical;
by id visit;
run;
*** part A (includes part C - add number of visits);

data changes;
retain hc hs hd hh;
set clinical;
by id;
if first.id then do;
  hc = chol;
  hs = sbp;
  hd = dbp;
  hh = hr;
  visits = 0;
end;
visits+1;
if last.id then do;
  diff_chol = chol - hc;
  diff_sbp = sbp - hs;
  diff_dbp = dbp - hd;
  diff_hr = hr - hh;
  output;
end;
keep id group diff: visits;
run;

title 'DATA SET SHOWING NUMBER OF VISITS PER PERSON';
proc print data=changes;
run;

*** part B;

title 'MEAN CHANGE IN CLINICAL DATA';
proc means data=changes maxdec=1 mean;
var diff: ;
class group;
run;

*** part B+ (change MEANS to TTEST);

title 'COMPARE GROUPS T-TEST';
proc ttest data=changes;
var diff: ;
class group;
run;

title;

You will learn an alternative method for #3 when COMBINING DATA is covered later in the semester...

*** #3 - alternative (coming attractions);

*** assume you read the data set as shown previously and create data set CLINICAL;

proc sort data=clinical;
by id visit;
run;

*** create two data sets, one with FIRST visit data, the other with LAST visit data;

data first last;
set clinical;
by id;
if first.id then output first;
else
  if last.id then output last;
run;
*** create a data set that contains the number of visits for each unique value of ID;

    proc freq data=clinical;
    table id / nprint out=howmany (rename=(count=visits));
    run;

    title 'HOWMANY';
    proc print data=howmany;
    run;

    title 'FIRST';
    proc print data=first;
    run;

    title 'LAST';
    proc print data=last;
    run;

*** combine the data sets, rename variables as you combine, compute differences;

    data both;
    merge
        howmany
        first (rename = (chol=_chol sbp=_sbp dbp=_dbp hr=_hr))
        last;
    by id;
    diff_chol = chol - _chol;
    diff_sbp  = sbp  - _sbp;
    diff_dbp  = dbp  - _dbp;
    diff_hr   = hr   - _hr;
    keep id group visits diff: ;
    run;

    title 'BOTH - WITH DIFFERENCES';
    proc print data=both;
    run;