You have some data that look as follows....

```
data study;
input
id    :   $3.
visit :   mmddyy.
chol
;
format visit mmddyy10.;
datalines;
001    10/15/2004      200
002    10/15/2004      200
003    10/15/2004      300
004    10/15/2004      275
005    10/15/2004      250
002    11/10/2004      175
002    11/10/2004      175
002    11/10/2004      175
002    11/10/2004      195
004    11/13/2004      275
003    11/14/2004      280
004    12/14/2004      275
;
run;
```

Each record in the data file has: an ID number; a date of visit; a cholesterol measurement. Some subjects have a single occurrence (IDs 001 and 005) while others have multiple occurrences. The following are some tasks that can be accomplished using FIRST. and LAST. variables. Use of these variables requires a SET statement + a BY statement, in this case...

```
set study;
by id;
```

Whenever you use a BY statement, SAS requires that the data set being used be sorted according to all the variables in the BY statement. Also, since you will be looking for first time you saw a subject, last time, etc., you also sort the data by VISIT within each ID...

```
proc sort data=study;
by id visit;
run;

proc print data=study;
run;
```

The data set now looks as follows...

```
Obs    id          visit    chol
  1    001    10/15/2004     200
  2    002    10/15/2004     200
  3    002    11/10/2004     175
  4    002    11/10/2004     175
  5    002    11/10/2004     175
  6    002    11/10/2004     195
  7    003    10/15/2004     300
  8    003    11/14/2004     280
  9    004    10/15/2004     275
 10    004    11/13/2004     275
 11    004    12/14/2004     275
 12    005    10/15/2004     250
```

One way to learn about FIRST. and LAST. variables is to print the data set showing the their values within each observation. Since FIRST. and LAST. variables ONLY EXIST FOR THE DURATION OF THE DATA STEP and are NOT ADDED TO THE DATA SET, you must create new variables that contain the values if the FIRST. and LAST. variables...

```
data fl;
set study;
by id;
first_id = first.id;
last_id  = last.id;
label
first_id = 'first.id'
last_id  = 'last.id'
;
run;

proc print data=fl label;
run;
```

```
Obs    id          visit    chol    first.id    last.id
  1    001    10/15/2004     200        1          1

  2    002    10/15/2004     200        1          0
  3    002    11/10/2004     175        0          0
  4    002    11/10/2004     175        0          0
  5    002    11/10/2004     175        0          0
  6    002    11/10/2004     195        0          1

  7    003    10/15/2004     300        1          0
  8    003    11/14/2004     280        0          1

  9    004    10/15/2004     275        1          0
 10    004    11/13/2004     275        0          0
 11    004    12/14/2004     275        0          1

 12    005    10/15/2004     250        1          1
```

The values of the FIRST. and LAST. variables are helpful in answering the questions about your data. For example, if you are looking for observations in which the variable FIRST.ID has a value of 1 (the first observation within each ID), you can use either...

```
if first.id;
```

or...

```
if first.id eq 1;
```

or...

```
if first.id ne 0;
```

#1 Create a new data set that contains one observation per ID --- *the FIRST time each ID participated in your study*.

look for observations where FIRST.ID has a value of 1

```
data study_f;
set study;
by id;
if first.id;
run;

FIRST VISIT
Obs    id          visit    chol
 1     001    10/15/2004     200
 2     002    10/15/2004     200
 3     003    10/15/2004     300
 4     004    10/15/2004     275
 5     005    10/15/2004     250
```

#2 Create a new data set that contains one observation per ID --- *the LAST time each ID participated in your study*.

look for observations where LAST.ID has a value of 1

```
data study_l;
set study;
by id;
if last.id;
run;

LAST VISIT
Obs    id          visit    chol
 1     001    10/15/2004     200
 2     002    11/10/2004     195
 3     003    11/14/2004     280
 4     004    12/14/2004     275
 5     005    10/15/2004     250
```

#3 Create a new data set that contains two observations per ID --- *the FIRST and LAST time each ID participated in your study*.

look for observations where FIRST.ID or LAST.ID has a value of 1

```
* first and last time you saw each ID;
data study_fl;
set study;
by id;
if first.id or last.id;
run;

FIRST AND LAST VISIT
Obs    id          visit    chol
 1     001    10/15/2004     200
 2     002    10/15/2004     200
 3     002    11/10/2004     195
 4     003    10/15/2004     300
 5     003    11/14/2004     280
 6     004    10/15/2004     275
 7     004    12/14/2004     275
 8     005    10/15/2004     250
```

#4 Create two data sets --- *one with all subjects who only have ONE observation in the data set, one with subjects who have MULTIPLE observations in the data set*.

identify ONE observation subjects as those with both FIRST.ID and LAST.ID having the value 1
all others are MULTIPLE observation subjects

```
data single multiple;
set study;
by id;
if first.id and last.id then output single;
else output multiple;
run;
```

```
SINGLE VISIT
Obs    id          visit    chol
 1     001    10/15/2004     200
 2     005    10/15/2004     250
```

```
MULTIPLE VISITS
Obs    id          visit    chol
  1    002    10/15/2004     200
  2    002    11/10/2004     175
  3    002    11/10/2004     175
  4    002    11/10/2004     175
  5    002    11/10/2004     195
  6    003    10/15/2004     300
  7    003    11/14/2004     280
  8    004    10/15/2004     275
  9    004    11/13/2004     275
 10    004    12/14/2004     275
```

#5 Create one data set from the original data set STUDY --- *the FIRST time each ID participated in your study for only those subjects with multiple visits.*

```
data study_fm;
set study;
by id;
if first.id and not last.id;
run;
```

```
FIRST VISIT OF MULTIPLE VISITS
Obs    id          visit    chol
 1     002    10/15/2004     200
 2     003    10/15/2004     300
 3     004    10/15/2004     275
```

NOTE: You could have used the new data set MULTIPLE created in #4 and just specify...

```
if first.id;
```

#6 There should not be any repeated dates with any ID --- *create a data set with repeated dates within any of the IDs.*

Once again, it is helpful to know the values of the FIRST. and LAST. variables in the data step...

```
data fl;
set study;
by id visit;
first_id = first.id;
last_id  = last.id;
first_visit = first.visit;
last_visit  = last.visit;
label
first_id = 'first.id'
last_id  = 'last.id'
first_visit = 'first.visit'
last_visit  = 'last.visit'
;
run;

proc print data=fl label;
run;
```

```
                                                          first.   last.
Obs    id          visit    chol    first.id    last.id    visit    visit
  1    001    10/15/2004     200        1          1         1        1

  2    002    10/15/2004     200        1          0         1        1
  3    002    11/10/2004     175        0          0         1        0
  4    002    11/10/2004     175        0          0         0        0
  5    002    11/10/2004     175        0          0         0        0
  6    002    11/10/2004     195        0          1         0        1

  7    003    10/15/2004     300        1          0         1        1
  8    003    11/14/2004     280        0          1         1        1

  9    004    10/15/2004     275        1          0         1        1
 10    004    11/13/2004     275        0          0         1        1
 11    004    12/14/2004     275        0          1         1        1

 12    005    10/15/2004     250        1          1         1        1
```

How can you identify repeated dates within each ID...

```
data repeats;
set study;
by id visit;
if not (first.visit and last.visit);
run;
```

```
REPEATED VISITS WITHIN AN ID
Obs    id          visit    chol
 1     002    11/10/2004     175
 2     002    11/10/2004     175
 3     002    11/10/2004     175
 4     002    11/10/2004     195
```

#6 As in #5, one common use of FIRST. and LAST. variables is to identify duplicate observations within a data set. For example, if you are working with the vital statistics death file, each observation contains a social security number (SSN). There should only be one observation per SSN --- no repeated 'deaths' or individuals with identical SSNs --- *create one data set with observations with duplicate SSNs and another with unique SSNs*.

```
* assume the data set with social security numbers is named DEATHS;
* assume the variable with the social security number is named SSN;

proc sort data=deaths;
by ssn;
run;

data duplicates unique;
set deaths;
by ssn;
if not (first.ssn and last.ssn) then output duplicates;
else output unique;
run;
```