Lecture Set 5 – All About Combining SAS Data Sets

By Group Processing

- When you use a BY statement while setting a SAS data set certain temporary automatic variables are created
  - FIRST.variable is a 1 for the first observation in a by group, and 0 otherwise
  - LAST.variable is a 1 for the last observation in a by group, and 0 otherwise
  - Note: For BY statements with more than one variable, a change in the value of a primary by variable forces last.variable to equal 1 for the secondary by variables

LIBNAME sasdata 'X:/rottesen/stat440/sasdata/';

PROC SORT DATA=sasdata.height OUT=height;
  BY gender height;
RUN;

DATA SortHeight; SET height;
  BY gender height;
** cant see FIRST. and LAST. b/c they are temp, unless we assign variables;
  IF FIRST.gender THEN FGender=1; ELSE FGender=0;
  IF LAST.gender THEN LGender=1; ELSE LGender=0;
  IF FIRST.height THEN FHeight=1; ELSE FHeight=0;
  IF LAST.height THEN LHeight=1; ELSE LHeight=0;
RUN;

** note: sally is an example of a forced LAST.;
PROC PRINT; RUN;

LOG not included...
### Direct Access

- We are used to reading data observation by observation, in order
- However we can go straight to any given observation by using the POINT= option in the SET statement

```sas
DATA point;
  num=2;
  SET height POINT=num;
  ** to use POINT= we need output to create the observation;
  ** and stop to keep SAS out of an endless loop;
  OUTPUT;
  STOP;
RUN;
PROC PRINT; RUN;
```

**LOG**

```
57 DATA point;
58     num=2;
59     SET height point=num;
60     ** to use POINT= we need output to create the observation;
61     ** and stop to keep SAS out of an endless loop;
62     OUTPUT;
63     STOP;
64     RUN;

NOTE: The data set WORK.POINT has 1 observations and 3 variables.
NOTE: DATA statement used (Total process time):
    real time 0.00 seconds
cpu time 0.00 seconds
```

---

<table>
<thead>
<tr>
<th>Obs</th>
<th>gender</th>
<th>height</th>
<th>name</th>
<th>FGender</th>
<th>LGender</th>
<th>FHeight</th>
<th>LHeight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>F</td>
<td>62</td>
<td>Lisa</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>65</td>
<td>Betty</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>F</td>
<td>65</td>
<td>Bonnie</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>F</td>
<td>65</td>
<td>Sally</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>M</td>
<td>65</td>
<td>Allan</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>M</td>
<td>69</td>
<td>Jose</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>M</td>
<td>70</td>
<td>Bobby</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>M</td>
<td>70</td>
<td>Jim</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Detecting the End of A Data Set

- To get the last observation without directly specifying the obs number we can use the END= option in the SET statement.

```plaintext
LIBNAME sasdata 'X:/rottesen/stat440/sasdata/';
** don't have to create an output data set, but then;
** sasdata.StatGrades would be sorted, depends on what you want;
PROC SORT DATA=sasdata.StatGrades OUT=StatGrades;
   BY percentage;
RUN;
PROC PRINT; RUN;

DATA HighGrade;
   SET StatGrades END=lastrec;
   IF lastrec;
RUN;
PROC PRINT; RUN;

... portion of log removed ...
76   DATA HighGrade;
77     SET StatGrades END=lastrec;
78     IF lastrec;
79   RUN;
NOTE: There were 26 observations read from the data set WORK.STATGRADES.
NOTE: The data set WORK.HIGHGRADE has 1 observations and 2 variables.
NOTE: DATA statement used (Total process time):
      real time          0.00 seconds
      cpu time           0.00 seconds

... portion of log removed ...
```
### Other Points

- Chapter 11 also includes info on:
  - Summary table of manipulating data actions (page 336-337)
  - Where to use drop/keep
  - More on Compilation/Execution when setting SAS data sets, rather than reading in raw data
    - No input buffer
    - Important: SAS does not set variable values to missing as it did when reading raw data. Only newly created variables are set to missing (with the exception of RETAIN and SUM statement variables). See page 346-353

<table>
<thead>
<tr>
<th>Obs</th>
<th>name</th>
<th>percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ulric</td>
<td>0.922</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Obs</th>
<th>name</th>
<th>percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hans</td>
<td>0.073</td>
</tr>
<tr>
<td>2</td>
<td>Thomas</td>
<td>0.668</td>
</tr>
<tr>
<td>3</td>
<td>Lisa</td>
<td>0.679</td>
</tr>
<tr>
<td>4</td>
<td>Quail</td>
<td>0.698</td>
</tr>
<tr>
<td>5</td>
<td>Fred</td>
<td>0.742</td>
</tr>
<tr>
<td>6</td>
<td>Elmer</td>
<td>0.744</td>
</tr>
<tr>
<td>7</td>
<td>Gregg</td>
<td>0.749</td>
</tr>
<tr>
<td>8</td>
<td>Otto</td>
<td>0.792</td>
</tr>
<tr>
<td>9</td>
<td>Betty</td>
<td>0.800</td>
</tr>
<tr>
<td>10</td>
<td>John</td>
<td>0.802</td>
</tr>
<tr>
<td>11</td>
<td>Isa</td>
<td>0.812</td>
</tr>
<tr>
<td>12</td>
<td>Ron</td>
<td>0.821</td>
</tr>
<tr>
<td>13</td>
<td>Zilly</td>
<td>0.834</td>
</tr>
<tr>
<td>14</td>
<td>Carlie</td>
<td>0.855</td>
</tr>
<tr>
<td>15</td>
<td>Phillip</td>
<td>0.864</td>
</tr>
<tr>
<td>16</td>
<td>Siri</td>
<td>0.864</td>
</tr>
<tr>
<td>17</td>
<td>Kerry</td>
<td>0.870</td>
</tr>
<tr>
<td>18</td>
<td>Xuxa</td>
<td>0.873</td>
</tr>
<tr>
<td>19</td>
<td>William</td>
<td>0.876</td>
</tr>
<tr>
<td>20</td>
<td>Arthur</td>
<td>0.879</td>
</tr>
<tr>
<td>21</td>
<td>Neil</td>
<td>0.882</td>
</tr>
<tr>
<td>22</td>
<td>Dan</td>
<td>0.894</td>
</tr>
<tr>
<td>23</td>
<td>Maire</td>
<td>0.899</td>
</tr>
<tr>
<td>24</td>
<td>Yee</td>
<td>0.911</td>
</tr>
<tr>
<td>25</td>
<td>Van</td>
<td>0.917</td>
</tr>
<tr>
<td>26</td>
<td>Ulric</td>
<td>0.922</td>
</tr>
</tbody>
</table>
Methods of Combining

- There are several ways to combine data sets
  - One-to-one reading (SET)
  - Concatenating (SET) and PROC APPEND
  - Interleaving (SET, BY)
  - Match Merging (MERGE,BY)
  - FYI: PROC SQL (not on test)
- Don't forget if you have same named variables the variable in the last data set overwrites the others, however the first occurrence of the variable (i.e. the first data set) is what determines it's length
- Also for methods that use BY statements the data sets to be combined must be sorted by those variables

PROC APPEND

- Similar to concatenating by stacking data sets
- The major differences are:
  - Using a PROC instead of a DATA step
  - Doesn't create a new data set, just appends data to an existing base data set
  - Only two data sets can be used
  - Observations in the base data set are not read and the descriptor portion of the base data set does not change
- Use the FORCE option when you have:
  - variables that are in the append data set but not in the base data set
  - different type variables (e.g. character or numeric)
  - different length of variables

```sas
LIBNAME sasdata 'X:/rottesen/stat440/sasdata/';
PROC PRINT DATA=sasdata.FacultyUsers; RUN;
PROC PRINT DATA=sasdata.OtherUsers; RUN;
** create temp data sets to not overwrite originals;
DATA Faculty; SET sasdata.FacultyUsers;
  RUN;
DATA Other; SET sasdata.OtherUsers;
  RUN;
PROC APPEND BASE=Faculty DATA=Other;
  RUN;
PROC PRINT; RUN;
```
2415  PROC APPEND BASE=Faculty DATA=Other;
2416  RUN;

NOTE: Appending WORK.OTHER to WORK.FACULTY.
NOTE: There were 395 observations read from the data set WORK.OTHER.
NOTE: 395 observations added.
NOTE: The data set WORK.FACULTY has 1797 observations and 7 variables.
NOTE: PROCEDURE APPEND used (Total process time):
  real time           0.13 seconds
  cpu time            0.06 seconds

<table>
<thead>
<tr>
<th>Obs</th>
<th>Fname</th>
<th>Lname</th>
<th>email</th>
<th>phone</th>
<th>department</th>
<th>class</th>
<th>userid</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHARLES</td>
<td>PORTER</td>
<td><a href="mailto:charles.porter@calpoly.edu">charles.porter@calpoly.edu</a></td>
<td>756-8511</td>
<td>AERO</td>
<td>Faculty</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>DONALD</td>
<td>STEPHENS</td>
<td><a href="mailto:donald.stephens@calpoly.edu">donald.stephens@calpoly.edu</a></td>
<td>756-6884</td>
<td>CHEM</td>
<td>Faculty</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>CRYSTAL</td>
<td>FLORES</td>
<td><a href="mailto:crystal.flores@calpoly.edu">crystal.flores@calpoly.edu</a></td>
<td>756-7053</td>
<td>AERO</td>
<td>Faculty</td>
<td>3</td>
</tr>
<tr>
<td>1400</td>
<td>TAMMY</td>
<td>RUSSELL</td>
<td><a href="mailto:tammy.russell@calpoly.edu">tammy.russell@calpoly.edu</a></td>
<td>756-8791</td>
<td>CSC</td>
<td>Faculty</td>
<td>1792</td>
</tr>
<tr>
<td>1401</td>
<td>PAUL</td>
<td>HAMILTON</td>
<td><a href="mailto:paul.hamilton@calpoly.edu">paul.hamilton@calpoly.edu</a></td>
<td>756-9859</td>
<td>AERO</td>
<td>Faculty</td>
<td>1795</td>
</tr>
<tr>
<td>1402</td>
<td>TINA</td>
<td>STEPHENS</td>
<td><a href="mailto:tina.stephens@calpoly.edu">tina.stephens@calpoly.edu</a></td>
<td>756-2983</td>
<td>BIO</td>
<td>Faculty</td>
<td>1796</td>
</tr>
<tr>
<td>1</td>
<td>DAWN</td>
<td>CUNNINGHAM</td>
<td><a href="mailto:dawn.cunningham@calpoly.edu">dawn.cunningham@calpoly.edu</a></td>
<td>756-8747</td>
<td>MATH</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>THOMAS</td>
<td>PERKINS</td>
<td><a href="mailto:thomas.perkins@calpoly.edu">thomas.perkins@calpoly.edu</a></td>
<td>756-6241</td>
<td>MATH</td>
<td>254</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>PAMELA</td>
<td>COX</td>
<td><a href="mailto:pamela.cox@calpoly.edu">pamela.cox@calpoly.edu</a></td>
<td>756-7837</td>
<td>BMED</td>
<td>453</td>
<td></td>
</tr>
<tr>
<td>393</td>
<td>TIFFANY</td>
<td>WOOD</td>
<td><a href="mailto:tiffany.wood@calpoly.edu">tiffany.wood@calpoly.edu</a></td>
<td>756-3983</td>
<td>PHYS</td>
<td>Grad Student</td>
<td>1793</td>
</tr>
<tr>
<td>394</td>
<td>PATRICK</td>
<td>BENNETT</td>
<td><a href="mailto:patrick.bennett@calpoly.edu">patrick.bennett@calpoly.edu</a></td>
<td>756-1835</td>
<td>MATE</td>
<td>Staff</td>
<td>1794</td>
</tr>
<tr>
<td>395</td>
<td>EDWARD</td>
<td>BELL</td>
<td><a href="mailto:edward.bell@calpoly.edu">edward.bell@calpoly.edu</a></td>
<td>756-7201</td>
<td>MATE</td>
<td>Staff</td>
<td>1797</td>
</tr>
<tr>
<td>Obs</td>
<td>Fname</td>
<td>Lname</td>
<td>email</td>
<td>phone</td>
<td>department</td>
<td>class</td>
<td>userid</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>------------</td>
<td>---------------------------------</td>
<td>-----------</td>
<td>------------</td>
<td>------------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>CHARLES</td>
<td>PORTER</td>
<td><a href="mailto:charles.porter@calpoly.edu">charles.porter@calpoly.edu</a></td>
<td>756-8511</td>
<td>AERO</td>
<td>Faculty</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>DONALD</td>
<td>STEPHENS</td>
<td><a href="mailto:donald.stephens@calpoly.edu">donald.stephens@calpoly.edu</a></td>
<td>756-6884</td>
<td>CHEM</td>
<td>Faculty</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>CRYSTAL</td>
<td>FLORES</td>
<td><a href="mailto:crystal.flores@calpoly.edu">crystal.flores@calpoly.edu</a></td>
<td>756-7053</td>
<td>AERO</td>
<td>Faculty</td>
<td>3</td>
</tr>
</tbody>
</table>

... portion of output removed ...

| 1401 | PAUL    | HAMILTON  | paul.hamilton@calpoly.edu       | 756-9859  | AERO       | Faculty    | 1795   |
| 1402 | TINA    | STEPHENS  | tina.stephens@calpoly.edu       | 756-2983  | BIO        | Faculty    | 1796   |
| 1403 | DAWN    | CUNNINGHAM| dawn.cunningham@calpoly.edu     | 756-8747  | MATH       | 36         |
| 1404 | THOMAS  | PERKINS   | thomas.perkins@calpoly.edu      | 756-6241  | MATH       | 254        |
| 1405 | PAMELA  | COX       | pamela.cox@calpoly.edu         | 756-7837  | BMED       | 453        |
| 1406 | LORI    | PHILLIPS  | lori.phillips@calpoly.edu       | 756-8299  | IME        | Grad Student| 801    |

... portion of output removed ...
Merging

LIBNAME sasdata 'X:/rottesen/stat440/sasdata/';
PROC SORT DATA=sasdata.StatGrades OUT=Grades; BY name; RUN;
PROC PRINT; RUN;
PROC SORT DATA=sasdata.OfficeHours OUT=OfficeHours; BY name; RUN;
PROC PRINT; RUN;
DATA OH NoOH;
   MERGE Grades (IN=all)
      OfficeHours (IN=OH);
   BY Name;
   ** FIRST. and LAST. can also be created with a MERGE, with BY;
   IF FIRST.Name THEN count=0;  ** start count at 0 for each new name;
      count+1;  ** run counter for each office hour attended;
   IF FIRST.Name AND date=. THEN count=0;  ** those with no OH get 0;
   IF all=1 AND OH=1 /*AND LAST.name*/ THEN OUTPUT OH;
   ELSE IF all=1 AND OH=0 /*AND LAST.name*/ THEN OUTPUT NoOH;
RUN;
PROC PRINT DATA=OH; FORMAT date MMDDYY8.; RUN;
PROC PRINT DATA=NoOH; FORMAT date MMDDYY8.; RUN;
Lecture Set 5

DATA OH NoOH;
MERGE Grades (IN=all)
    OfficeHours (IN=OH);
BY Name;
** FIRST. and LAST. can also be created with a MERGE, with BY;
FIRST.Name THEN count=0; ** start count at 0 for each new name;
count+1; ** run counter for each office hour attended;
IF FIRST.Name AND date=. THEN count=0; ** those with no OH get 0;
IF all=1 AND OH=1 /*AND LAST.name*/ THEN OUTPUT OH;
ELSE IF all=1 AND OH=-1 /*AND LAST.name*/ THEN OUTPUT NoOH;
RUN;

NOTE: There were 26 observations read from the data set WORK.GRADES.
NOTE: There were 208 observations read from the data set WORK.OFFICEHOURS.
NOTE: The data set WORK.OH has 208 observations and 4 variables.
NOTE: The data set WORK.NOOH has 2 observations and 4 variables.
NOTE: DATA statement used (Total process time):
    real time           0.01 seconds
    cpu time            0.01 seconds

Obs    name      percentage        Date    count
1    Arthur       0.879      10/02/04       1
2    Arthur       0.879      10/03/04       2
3    Arthur       0.879      10/05/04       3
4    Arthur       0.879      10/09/04       4
5    Arthur       0.879      10/11/04       5
6    Arthur       0.879      10/13/04       6
7    Arthur       0.879      11/04/04       7
8    Arthur       0.879      11/06/04       8
9    Arthur       0.879      11/09/04       9
10   Arthur       0.879      11/25/04      10
11   Arthur       0.879      12/07/04      11
12   Arthur       0.879      12/11/04      12
13   Arthur       0.879      12/15/04      13
14   Betty        0.800      10/06/04       1
15   Betty        0.800      10/14/04       2
16   Betty        0.800      10/25/04       3
17   Betty        0.800      11/19/04       4
18   Betty        0.800      12/07/04       5

Obs    name      percentage        Date    count
1    Gregg       0.749             .      0
2    Van         0.917             .      0
Other Points

- Chapter 12 also includes info on:
  - Specifics and examples for each method
  - A walk through compilation/execution for match merge processing pages 379-385, very specific details
  - RENAME data set option
  - Specifics on where to place DROP/KEEP options