

Lab 8 *Some of these problems may be more challenging than others. Please feel free to work with others, attend office hours, or post on the course discussion forum if you need help. While collaboration with other students is encouraged, each student is responsible for submitting his or her own work. This assignment should be submitted in one well-commented SAS program. For any questions that require a written answer, do so in the SAS comments. Be sure to re-name the uploaded SAS scripts according to the naming convention `LastnameFirstinitial_Lab#.sas` (e.g., `PileggiS_Lab8.sas`).*

Project Euler (<https://projecteuler.net/>) presents a set of challenging problems that requires programming to solve. This project is gaining in popularity and promise - job candidates are even listing on their resume how many Project Euler problems they have solved. For this exercise, we will examine **Problem #19**: how many Sundays fell on the first of the month during the twentieth century (January 1, 1901 to December 31, 2000).

1. Discuss with your neighboring students a *game plan* for solving this problem. After a few minutes, students will be randomly selected to present their game plan to the class. Include your game plan as a comment in your SAS code. (*Hint: you will need to use SAS date functions!*)
2. Write SAS code to execute your game plan. In your SAS code, create a new observation for each instance in which a Sunday fell on the first of the month such that you can see the date on which it occurs. Print your results, and be sure to format your date values so that the dates are readable. (*So that you can verify your result: I got 171 Sundays, with the last instance being Oct 1, 2000.*)

A local gym ran a New Year's promotional membership offer available to anyone who joined the gym on New Year's day. As a result 245 people signed up for this offer. Members were tracked automatically via the computer check-in as to when they arrived and left the gym. This data is contained in the SAS data set called `NewYears.sas7bdat`, which has the following variables:

- `id` - member id
 - `inday1-inday119` - check-in time, recorded in seconds since 12:00am, for the first 119 days of the year
 - `outday1-outday119` - check-out time, recorded in seconds since 12:00am, for the first 119 days of the year
3. Create a library reference called `mylib` to access the `NewYears.sas7bdat` data set.
 4. Create a temporary data set that copies the data from `NewYears.sas7bdat`. Do the remaining items in this data set.
 5. Calculate the daily time (in minutes) each member spent at the gym, for each day.
 6. Calculate the average daily time (in minutes) each member spent at the gym.

7. Apply PROC MEANS to you average daily time variable. Your results should match those below.

Analysis Variable : ave_workout				
N	Mean	Std Dev	Minimum	Maximum
245	65.9666775	36.7278211	15.3734559	125.3552726