PHPM 672/677 Assignment #1: Hello World!
Due date: Email by 11:59pm Monday 1/26/2014

Submission. Submit on blackboard by 11:59pm the day before the class they are due.

- Commented code (SAS: lnameN.sas, where N indicates the assignment number and lname is your last name)
- Output from your code (SAS: lnameN.log & lnameN.lst)

Homework Assignments. There will be a homework assignment due approximately every other week. Homework assignments and related materials will be made available on the course website. **I strongly advise students to begin to work on their homework assignments soon after they are assigned, so that if they need help from the instructor they have time to ask and receive assistance.** There will be ***six (6)*** regular HW assignments during the semester. The overall weight of the HW assignments on the computation of the final course grade is ***48%, HALF the grade.***

All assignments are due at 11:59pm the day before the class they are due.

Late Assignments. Each student will be allowed one late assignment, due 7 days from the due date. NO other late assignments or make up will be accepted.

Tentative Assignments and Exam Schedule

Date	Assignment Given	Assignment Due
1/20/2014	Assignment 1	
1/27/2014	Assignment 2	Assignment 1
2/3/2014		
2/10/2014	Assignment 3	Assignment 2
2/17/2014		
2/24/2014	Assignment 4	Assignment 3
3/3/2014		
3/10/2014	Midterm	
3/17/2014	SPRING BREAK	
3/24/2014	Assignment 5	Assignment 4
3/31/2014	Assignment 6	Assignment 5
4/7/2014		
4/14/2014	Final Project (FP)	Assignment 6
4/21/2014		FP MileStone 1
4/28/2014		FP MileStone 2
5/5/2014		Final Project

Collaboration: Collaboration on assignments, in class labs and homeworks, IS encouraged. However, what you hand in must be in your own writing/typing. Good scholarship requires that all collaborations must be acknowledged. Thus, if you collaborate on the solution of the problem set, we expect that you list your collaborators at the top of the page. Collaboration on in-class evaluations (quizzes, mid-terms, and the final exams) is, of course, a violation. This includes a discussion of questions on a quiz, midterm, or final with students from sections that has not yet taken the evaluation.

Plagiarism: If you consult any outside sources when doing your work, you are expect to further document these sources. Give credit where credit is due. Plagiarism will not be tolerated.

<u>All handed in homework should state at the top any assistance with debugging and programming, as well as citations of any program segments copied from a website.</u>

Seeking Help: Language classes are cumulative; don't fall behind. Ask for assistance from your fellow students or from the instructor, especially if you find yourself struggling. But remember in the end, programming classes are like skill building math classes. Either you know how to do them or not. No amount of watching others or the instructor program or debug it will suffice for you to build these skills. Only your hard work to work through them will build your skills. It is a lot of work to build these skills, but data skills are highly valuable in the job market in the modern digital world, so it will be well worth your effort. Furthermore, once you learn to think in this manner, it's not something you forget.

Elegance: There is always more than one way to say something, but some ways will be more "elegant" than others. You will learn to recognize elegant expressions as you become more familiar with a language and use the elegant idioms as you become more skilled.

Recommended readings for this assignment

In the following sections, if I copied and pasted sections from any of these websites, it is italicized.

1. http://www.ats.ucla.edu/stat/sas/

Assignment 1: Hello World!

In this assignment, you will write your first SAS code. It is a very simple assignment, so you should be able to complete this in the computer lab if you do not have access to the software elsewhere.

Setting Up

1. Locate the software (SAS) on the computer and execute it
2. Create a working directory, often referred to as pwd (present working directory), where you will work on this assignment. For example, "assign1".
3. When you are writing programs, it is often a good idea to put your code in a different directory from your datasets, because if you are writing a lot of code, or working with a lot of datasets, you will stay more organized. So, create a subdirectory called "data" in the pwd
4. Now, get a good text editor. For example, portable apps notepad++ or notepad2 are mentioned below. You should be able to unpack this software into your pwd or USB and run from there by double clicking on "Notepad2…" if you cannot install on the lab computer.
5. http://portableapps.com/apps/development/notepadpp_portable
   http://portableapps.com/apps/development/notepad2-mod_portable
6. Now you are READY!

Getting Data : Download Flu Trends Data from Google Flu Trends

1. Go to http://www.google.org/flutrends
2. Choose "United States" from dropdown on the left as "national"
3. Click "download data" link on the left (leave the second drop down)
4. Save as "flu.csv" in your data directory
5. Open this csv file using the text editor
6. Cut and Paste the comments on top (11 lines) into a different text file ("readme.txt"). This is documentation on what the data is for your reference later, in case you have to come back to this. The csv file should start with "Date, United States …"
7. Don't forget to save your files.

What you will do with the data via programming:

1. Import the data into the software
2. Find out how many variables (columns) and observations (rows) are in the data, and what is in the data. Note you have to get the software to output this, even if you know the answer to this question.
3. Print the first 10 observations
4. [EXTRA CREDIT (+1)] If you feel really ambitious, try to find the mean of each state. If you can't figure out how to do this after trying a bit, don't worry. We will learn this later in class.

P1. SAS programing:

1. Open a new file in the text editor. In this file, type:

```
*This program imports, describes, and displays the dataset;

libname data '../data';

proc import out= flu
            datafile= "../data/flu.csv"
            dbms=csv replace;
     getnames=yes;
     datarow=2;

proc contents data=flu;
proc print data=flu(obs=10);
```

   Save your program as “flu.sas” and exit.

2. Let's look at this program line by line.

   The first line of the program is a comment. Comments begin with a "*", and end with a ";" You can type anything you want in a comment, but generally comments give information about your program.

   The next line (libname) tells SAS where to find the dataset you will be using. "../data" means to look in the data subdirectory that you created.

   The next five lines import the csv dataset, and creates a working SAS dataset called flu, which contains the data from flu.csv. Note that this dataset is temporary and will be deleted when you are done running the code.

   The line after that (contents) gives a description of the data.

   The final line prints the first 10 observations in the data.

3. Use SAS to run your program. Note that the command you use to run SAS depends on the system you are using. If you double click on the sas program it should run sas.

4. Now, save out the log and lst.

P2. Try using point and click

Now, try doing the same task you did with point and click mode (without writing the program) in the SAS software.

P3. Questions:

Answer these questions in a text file called readme.txt (convention for documentation of programs).

1. Describe what data (flu.csv) you have in no more than three sentences
2. Write one thing you liked better and one thing you did not like about using programs compared to point and click.