Scalable Data Analytics **Homework 1**
Spring 2019 **February 11, 2019**

Deadlines Homework 1 is due on Mar.3rd 23:55pm. 50% late penalty will be applied within the first week of due date and no submission is accepted thereafter.

How to submit: Please submit a zip file to the *Assignment/Homework_1* folder in the iCollege. The zip file name should be 'Yourname-Pantherid.zip'. In the zipped folder it should contain the two separate ipython notebook files '1-generator.ipynb', '2-HOF.ipynb', and '3-HOF.ipynb' for the first, second and third problem respectively.

Data Set: We're going to use the Citibike dataset posted in the iCollege.

---

1. (5 points) Python's Generators and Streaming.

   Your task is to compute the median age of the Citibike's subscribed customers. You are required to read data line by line and are not allowed to store the entire data set in memory. Indeed, you should not have any containers (e.g. list, dictionary, DataFrame, etc.) with more than 100 elements in memory.

   What to submit:

   You will turn in an ipython notebook with the plot of the histogram of customers age and print out a single number showing the median age of the subscribed customers in the Citibike dataset.

2. (7.5 points) Python's Higher Order Functions

   This is how you can read the file and transform it to a list of lists.

   ```
   import pandas as pd
   df = pd.read_csv("citibike.csv")
   rows = df.values.tolist()
   ```

   (a).

   First we would like to know how many trips were from gender 1, and how many trips were from gender 2. We can do this by just counting the number of occurrences of "1" and "2" in the gender column:

   ```
   <Read file>
   <YOUR HOF EXPRESSION>

   # After this, you should get something like
   # (37805, 7848)
   ```

(b).

Second, we would like to count the number of trips per birth_year using higher order functions:

```
<Read file>
<YOUR HOF EXPRESSION>

# After this, you should get something like
# {"1900.0": 22, "1901.0": 1, "1910.0": 2, "1922.0": 4, ... "1995.0": 256,
"1996.0": 124, "1997.0": 94, "1998.0": 59, "1999.0": 17}
```

Hint: math.isnan() is able to remove all the nan values.

What to submit:

You will turn in an ipython notebook print out the results for question (a) and (b).

3. (2.5 points) Your task is to extract the first ride of the day from a Citibike data stream. The first ride of the day is interpreted as the ride with the earliest starting time of a day. For the sample data, which is a week worth of citibike records, your program should only generate 7 items (one for each day). However, instead of iterating through the stream using generators, you are asked to complete the task using higher order functions map(), filter() and/or reduce().

```
<ANY FUNCTION TO BE USED IN YOUR HOF>

with open('citibike.csv','r') as fi:
    reader = csv.DictReader(fi)
    first_birth_years = <YOUR HOF EXPRESSION>

# After this, your first_birth_years should be
# [1978, 1992, 1982, 1969, 1971, 1989, 1963]
```