Hong Kong Baptist University
Department of Computer Science

COMP1007 Introduction to Python and Its Applications
First Semester 2020-2021

Assignment #1

Assigned Date: 9th October 2020 (Friday)
Due Date for Section 1:   Before 1:30pm, 22nd October 2020 (Thursday)
Due Date for Section 2:   Before 1:30pm, 21st October 2020 (Wednesday)

Description:
This is an exercise to design and write a Python program in good programming style for a simulation of stock price over a period of 100 days. In this exercise, you are asked to simulate the stock price starting at $100.00 for 100 days with a daily fluctuation based on the Normal Distribution with mean = 0.0 & sigma = 0.0125. The program will show the daily stock price, the 7-day minimum, the 7-day maximum, the 7-day average, and the 50-day average. The program should also give an indication of "***" when the stock price drops beyond the 50-day average!

I highly recommend you take a look at the sample run of the program to get a better idea of the look and feel of this assignment.

What to submit:
Save all your source codes as a single Python file, name it "<yourSSOid>-assign1.py", and submit it through the course page at BU Moodle. That is, if your SSOid is 20709394, then you should name your python file, 20709394-assign1.py

Notes:
To become a good programmer, one should

1. Make use of the docstring to describe what this program is about in the Program Header
2. Put inline comments to remind you what a particular piece of code is about
3. Add in blank lines and makes the program more readable
4. Use meaningful names for functions and variables

- Happy Programming -

1) Make use of the *random* module to simulate the fluctuation of stock price.
    Here is the function and codes for simulating the changes in the stock prices:

```
import random
random.seed(37) # I like to set the random seed at 37!

def fluctuation():
    return (random.normalvariate(0.0, 0.0125))
…
StockPrice = 100.0 # StockPrice starts at $100.00
…
for i in range(100):
    StockPrice *= (1 + fluctuation()) # Calculate the next StockPrice
    Print(StockPrice)
```

2) Use a list StockHistory to keep track of the stock prices. Starting with an empty list, append to
    the end of the list the stock price of the day. Keep appending the stock price to the list until the
    length of the list is greater than 50. You can then delete the first item from the list and keep the
    list just to contain the last 50 records of the stock prices. And thus, you can calculate the 50-Day
    Average for the stock.

```
StockHistory = [] # Start with an empty list
…
StockHistory.append(StockPrice) # To append StockPrice to the end of the list
…
If (len(StockHistory) > 50): # If the list contains more than 50 records,
    StockHistory.pop(0)      # delete the first record which is indexed by zero
…
# Calculate the 50-Day Average here
```

3) Here is the suggested flow of the program:

- Construct the Program Header with program name, author's name, program description, … etc
- Import the module(s) needed
- Define the function(s) needed
- Set the random seed by simply adding "`random.seed(37)`" into the program
- Setup the global variable(s) with their initial value(s)
- Print the "Table Header"
- For the next 100 days do the followings:
  - Calculate the StockPrice according to the random fluctuation
  - Print the Number of Days and the current StockPrice
  - Append the StockPrice to the StockHistory list
  - The list should contain up to 50 records, delete the first record if necessary
  - Print out the 7-Day Min, Max, and Average if you have 7 or more records
  - Print out the 50-Day Average if you have reached and maintained at 50 records
  - Print out the indication of "***" if the StockPrice drops below the 50-Day Average

Sample Run:

```
 Day  Price   7DayMin 7DayMax 7DayAve  50DayAve
============================================================
    1 100.430   N/A     N/A     N/A    N/A:  1 record only!
    2 102.035   N/A     N/A     N/A    N/A:  2 record only!
    3 103.544   N/A     N/A     N/A    N/A:  3 record only!
    4 104.005   N/A     N/A     N/A    N/A:  4 record only!
    5 104.075   N/A     N/A     N/A    N/A:  5 record only!
    6 105.098   N/A     N/A     N/A    N/A:  6 record only!
    7 104.307 100.430 105.098 103.356  N/A:  7 record only!
    8 105.257 102.035 105.257 104.046  N/A:  8 record only!
    9 103.909 103.544 105.257 104.314  N/A:  9 record only!
   10 102.860 102.860 105.257 104.216  N/A: 10 record only!
   11 104.508 102.860 105.257 104.288  N/A: 11 record only!
   12 105.239 102.860 105.257 104.454  N/A: 12 record only!
   13 103.018 102.860 105.257 104.157  N/A: 13 record only!
   14 101.931 101.931 105.257 103.817  N/A: 14 record only!
   15 102.110 101.931 105.239 103.368  N/A: 15 record only!
   16 101.724 101.724 105.239 103.056  N/A: 16 record only!
   17 104.617 101.724 105.239 103.307  N/A: 17 record only!
   18 105.754 101.724 105.754 103.485  N/A: 18 record only!
   19 107.406 101.724 107.406 103.794  N/A: 19 record only!
   20 107.700 101.724 107.700 104.463  N/A: 20 record only!
   21 108.102 101.724 108.102 105.345  N/A: 21 record only!
   22 109.072 101.724 109.072 106.339  N/A: 22 record only!
   23 107.316 104.617 109.072 107.138  N/A: 23 record only!
   24 105.429 105.429 109.072 107.254  N/A: 24 record only!
   25 105.558 105.429 109.072 107.226  N/A: 25 record only!
   26 105.927 105.429 109.072 107.015  N/A: 26 record only!
   27 106.443 105.429 109.072 106.835  N/A: 27 record only!
   28 106.460 105.429 109.072 106.601  N/A: 28 record only!
   29 106.592 105.429 107.316 106.246  N/A: 29 record only!
   30 106.976 105.429 106.976 106.198  N/A: 30 record only!
   31 106.377 105.558 106.976 106.333  N/A: 31 record only!
   32 107.154 105.927 107.154 106.561  N/A: 32 record only!
   33 107.715 106.377 107.715 106.817  N/A: 33 record only!
   34 107.870 106.377 107.870 107.021  N/A: 34 record only!
   35 108.851 106.377 108.851 107.362  N/A: 35 record only!
   36 108.319 106.377 108.851 107.609  N/A: 36 record only!
   37 110.846 106.377 110.846 108.162  N/A: 37 record only!
   38 110.038 107.154 110.846 108.685  N/A: 38 record only!
   39 112.567 107.715 112.567 109.458  N/A: 39 record only!
   40 111.409 107.870 112.567 109.986  N/A: 40 record only!
   41 111.210 108.319 112.567 110.463  N/A: 41 record only!
   42 112.808 108.319 112.808 111.028  N/A: 42 record only!
   43 114.457 110.038 114.457 111.905  N/A: 43 record only!
   44 112.541 110.038 114.457 112.147  N/A: 44 record only!
   45 113.414 111.210 114.457 112.630  N/A: 45 record only!
   46 110.535 110.535 114.457 112.339  N/A: 46 record only!
   47 113.114 110.535 114.457 112.583  N/A: 47 record only!
   48 110.396 110.396 114.457 112.467  N/A: 48 record only!
   49 109.363 109.363 114.457 111.974  N/A: 49 record only!
   50 108.904 108.904 113.414 111.181  107.106
   51 107.142 107.142 113.414 110.410  107.240 ***
   52 107.967 107.142 113.114 109.632  107.359
   53 109.558 107.142 113.114 109.492  107.479
   54 107.356 107.142 110.396 108.669  107.546 ***
   55 108.697 107.142 109.558 108.427  107.638
```

```
 56 106.906    106.906 109.558 108.076   107.675 ***
 57 108.727    106.906 109.558 108.050   107.763
 58 108.526    106.906 109.558 108.248   107.828
 59 106.926    106.906 109.558 108.099   107.889 ***
 60 106.773    106.773 108.727 107.702   107.967 ***
 61 107.399    106.773 108.727 107.708   108.025 ***
 62 106.325    106.325 108.727 107.369   108.046 ***
 63 105.409    105.409 108.727 107.155   108.094 ***
 64 103.058    103.058 108.526 106.345   108.117 ***
 65 104.850    103.058 107.399 105.820   108.172 ***
 66 105.534    103.058 107.399 105.621   108.248 ***
 67 105.582    103.058 107.399 105.451   108.267 ***
 68 105.656    103.058 106.325 105.202   108.265 ***
 69 105.801    103.058 105.801 105.127   108.233 ***
 70 105.770    103.058 105.801 105.179   108.195 ***
 71 106.881    104.850 106.881 105.725   108.170 ***
 72 108.815    105.534 108.815 106.291   108.165
 73 109.718    105.582 109.718 106.889   108.213
 74 109.935    105.656 109.935 107.511   108.303
 75 111.708    105.770 111.708 108.376   108.426
 76 110.394    105.770 111.708 109.032   108.515
 77 110.657    106.881 111.708 109.730   108.600
 78 108.628    108.628 111.708 109.979   108.643 ***
 79 108.190    108.190 111.708 109.890   108.675 ***
 80 107.816    107.816 111.708 109.618   108.692 ***
 81 108.890    107.816 111.708 109.469   108.742
 82 107.755    107.755 110.657 108.904   108.754 ***
 83 107.756    107.755 110.657 108.528   108.755 ***
 84 107.046    107.046 108.890 108.012   108.739 ***
 85 109.494    107.046 109.494 108.135   108.751
 86 108.003    107.046 109.494 108.109   108.745 ***
 87 107.984    107.046 109.494 108.133   108.688 ***
 88 107.121    107.046 109.494 107.880   108.629 ***
 89 108.025    107.046 109.494 107.918   108.539 ***
 90 107.489    107.046 109.494 107.880   108.460 ***
 91 107.331    107.121 109.494 107.921   108.383 ***
 92 110.167    107.121 110.167 108.017   108.330
 93 108.463    107.121 110.167 108.083   108.210
 94 106.691    106.691 110.167 107.898   108.093 ***
 95 103.771    103.771 110.167 107.420   107.900 ***
 96 104.179    103.771 110.167 106.870   107.773 ***
 97 106.573    103.771 110.167 106.739   107.642 ***
 98 108.709    103.771 110.167 106.936   107.608
 99 106.547    103.771 108.709 106.419   107.552 ***
100 104.034    103.771 108.709 105.786   107.455 ***
```