CS 115 - Introduction to Programming in Python

Lab Guide 06

Lab Objectives: 2D Lists, Classes

1. a) Write a function **`formMatrix`** which gets an integer number **`n`** and creates and returns a square 2-dimensional matrix of size `n x n` according to the below rule:

   - The elements in the **even row and even column** elements of the matrix will be assigned to row value
   - The elements in the **odd row and odd column** elements of the matrix will be assigned to column value
   - Other elements will be assigned to 0

   (Assume n is 5 in the sample below).

0	0	0	0	0
0	1	0	3	0
2	0	2	0	2
0	1	0	3	0
4	0	4	0	4

b) Write a main to input the size of the matrix and display the matrix in matrix form after calling the above function.

Sample Run:

```
Enter size of the matrix: 6
0    0    0    0    0    0
0    1    0    3    0    5
2    0    2    0    2    0
0    1    0    3    0    5
4    0    4    0    4    0
0    1    0    3    0    5
```

2. Write a class called `Stock` (**Stock.py**) that represents a typical Stock object.

   The class will store the following attributes:

   - name
   - count
   - price
   - minimum stock level

   Note that all data attributes should be private (__).

   a. Your class should have **an init() method** that takes the values of all four attributes as parameters.
   b. Your class should have the following **methods**:
      - get_name: returns the name
      - get_count: returns the count
      - get_price: returns the price
      - get_min_stock: returns the minimum stock level

   c. You should also write the set_count() method for updating the stock's count.
   d. In addition to the above methods, your class should define a __repr__ method which returns the information about the stock as a String object with the format shown below, where "`hayley`" is the name, "`13`" is the count, and "`5.25`" is the price and 10 is the minimum stock level:

```
Name: hayley
Count: 13
Price: 5.25
Minimum stock level: 10
```

3. Create a Python script that does the following:
   a. Defines a method `getOrder()`: takes a list of stocks and returns the list of Stocks in the database which have quantity less than their minimum stock level.
   b. Defines a method `getByPrice ()`: returns the Stock name with the least price.
   c. Read the file `'input.txt'` and create a list of stocks using the data from the file.
   d. Print the stocks whose quantity is less than their minimum stock level.
   e. Print the name of the stock with the least price.
   f. Use the above methods where necessary.

Sample Run:

Stocks which should be Ordered:
Name: hayley
Count: 5
Price: 3.0
Minimum Stock Level: 12

Name: twix
Count: 13
Price: 3.75
Minimum Stock Level: 15

Name: damak
Count: 11
Price: 6.75
Minimum Stock Level: 15

Name: biskrem
Count: 15
Price: 4.0
Minimum Stock Level: 21

Name: metro
Count: 7
Price: 2.75
Minimum Stock Level: 10

Name: toblerone
Count: 7
Price: 16.75
Minimum Stock Level: 15

Name: hayley
Count: 4
Price: 3.0
Minimum Stock Level: 5

Name: biskrem
Count: 5
Price: 5.25
Minimum Stock Level: 10

The cheapest stock is “albeni”