DSC 200 – Data Wrangling
Lab 2: Python Basics - 2

Objectives:

1. Demonstrate ability to create and use methods/functions in Python
2. Demonstrate ability to implement a simple class in Python
3. Demonstrate ability to add methods/functions to python classes and use these methods

Instructions:

I. Write a script that stores and displays information about customers of a bank. The relevant information of a customer to be captured and/or maintained includes ***name, opening balance, closing balance, and customer type***. The opening balance is the balance on the account as at the time the account is accessed (we assume that amount is the balance at the end of the year prior to interest accrued). Each customer earns an interest of 12.5% on the opening balance. Once the interest is calculated, the closing balance is updated to reflect the interest (i.e., closing balance is the sum of opening balance and interest earned). The customer type is determined based on the closing balance. (See the next task for additional details about how to determine the customer type)

II. Create a customer class with a constructor and the following methods:
   - **calculateClosingBalance** – This method calculates the closing balance of the customer and stores the calculated value to be used later to determine a customer's status
   - **determineCustomerType** – This method determines the customer type based on the closing balance.
     The customer type is determined based on the closing balance as follows:

Closing balance greater than 150,000	customer type = Diamond
Closing balance between 100,000 and 150,000	customer type = Gold
Closing balance 90,000 or more and less than 100,000	customer type = Silver
Closing balance below 90,000	customer type = Bronze

   - **displayTabularInfo** – This method prints the name, opening and closing balances, interest and customer type of each customer. This should be a single row in the output table for the given customer object.
   - Note that the constructor should accept the customer's name and opening balance as formal parameters.

III. Input Validation: Per the requirements of the bank, each customer must have a minimum opening balance of $50 and cannot exceed $2,000,000.00.

IV. Your script should allow a user to enter the number of customers of the bank, and, for each customer, the required information, i.e. the customer name and their opening balance. The script should then create objects corresponding to the customers and store all the customer objects created in a list or dictionary.

V. Using the list/dictionary of customer objects created, your script should print a table that displays the list of customers and the related information as shown in the following sample run:

```
Enter the number of customers you have int his bank: 6
Enter customer 1's name: Victoria Adams
Enter customer 1's opening balance: 56798.67
Enter customer 2's name: Sam Jonah
Enter customer 2's opening balance: 89763.69
Enter customer 3's name: Kingsley James
Enter customer 3's opening balance: 897631.45
Enter customer 4's name: Samuel Adams
Enter customer 4's opening balance: 7809.09
Enter customer 5's name: Tom Hayes
Enter customer 5's opening balance: 99000
Enter customer 6's name: Tony Max
Enter customer 6's opening balance: 120000
Customer Name          Customer Type   Opening($)  Interest($) Closing($)
*********************************************************************************
Victoria Adams         Bronze           56798.67     7099.83    63898.50
Sam Jonah              Gold             89763.69    11220.46   100984.15
Kingsley James         Diamond         897631.45   112203.93  1009835.38
Samuel Adams           Bronze            7809.09      976.14     8785.23
Tom Hayes              Gold             99000.00    12375.00   111375.00
Tony Max               Gold            120000.00    15000.00   135000.00
*********************************************************************************
```

What to submit:

Submit your script (a single file named using the following format: ***your_nku_user_id_Lab2.py***) via the Canvas from which this assignment is linked.

Rubric:

Comments	6
Input Validation	9
Implementation of class methods	20
Creating and using objects	10
Required Output	5
Total	50