Lab #1: Basic Python + NumPy

This lab is the first step to get farmilliar with Python and a common Machine Learning library, named **NumPy**.

==========

Task 1. Python Programming

==========

Task 1.1. Develop a Python program to perform the following tasks:

– Input an integer number **n**.

– Input **n** integer numbers $\mathbf{x_1}$, $\mathbf{x_2}$,..., $\mathbf{x_n}$ and append them into a list **L**.

– Find the max/min element in **L**.

– Compute the sum of elements in L.

– Sort the list **L** in ascending order.

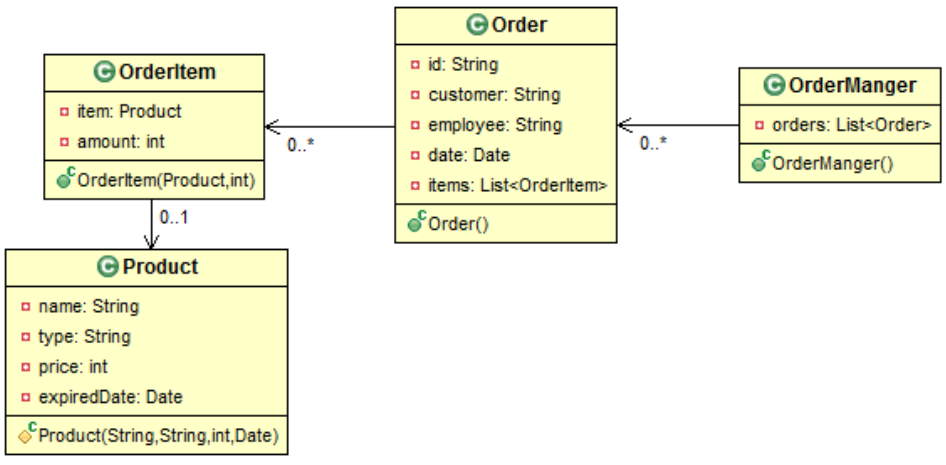
– Show how many positive and negative numbers are in the list **L**.

Hint: Use **min**(), **max**(), **sum**() functions to find max, min, and sum of the list L; **print**() to display values to the console.

Task 1.2. Develop a Python program to compute $\mathbf{S_n}$ as follows:

$$S_n = 1 + \frac{1}{3} + \frac{1}{5} + \cdots + \frac{1}{2n+1}$$

Task 1.3. For a given class diagram as follows:



In class OrderManager, implement the following methods:

1) Determine the product having the highest price.
2) Perform statistical task to show the number of products bought by each type of product (using dictionary)
3) Expand the problem by taking into account two kinds of products, called **ImportedProduct** and **DomesticProduct**. **ImportedProduct** has an additional attributed named **fromCountry**. Then, modify the implemented code and test it.

==========

Task 2. NumPy API

==========

Task 2.1. Develop a NumPy program to create an array with values ranging from 10 to 25, then reverse the array (the first element becomes last).

Task 2.2. Develop a NumPy program that constructs an array by repeating.

Input: [1, 2, 3, 4]

Expected Output:

Original array: [1, 2, 3, 4]

Repeating 2 times: [1 2 3 4 1 2 3 4]

Repeating 3 times: [1 2 3 4 1 2 3 4 1 2 3 4]

Task 2.3. Develop a NumPy program to replace all elements of NumPy array that are greater than a specified array.

Input:

[[0.42 0.48 0.32]
[0.74 0.58 0.38]
[0.51 0.34 0.15]]

Expected Output:

Original array:
[[0.42 0.48 0.32]
[**0.74 0.58** 0.38]
[**0.51** 0.34 0.15]]
Replace all elements of the original array with .5 which are greater than .5
[[0.42 0.48 0.32]
[0.5 0.5 0.38]
[0.5 0.34 0.15]]