CS 115 - Introduction to Programming in Python

Lab 02

Lab Objectives: Strings, Loops, Nested Loops.

Q1: Write a program `Lab02_Q1.py` to input a positive integer ***n*** and displays the sum of the first ***n*** terms of the sequence: $\sum_{i=1}^{n} \frac{1}{i}$ In other words, the program should generate the following sequence: 1 + (1/2) + (1/3) + (1/4) + (1/5) + ... + (1/*n*)

Assume that the input ***n*** is positive. The sum of the fractions should be displayed in 3 digits after the decimal point.

Sample Run:
`Enter an integer (-1 to stop): 1`
`the sum of 1 is 1.000`
	Sample Run:
	`Enter an integer (-1 to stop): 2`
	`the sum of 1 + 1/2 is 1.500`

Sample Run:
`Enter an integer (-1 to stop): 6`
`the sum of 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 is 2.450`
	Sample Run:
	`Enter an integer (-1 to stop): 10`
	`the sum of 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 + 1/7 + 1/8 + 1/9 + 1/10 is 2.929`

Q2: Write a program `Lab02_Q2.py` to input two integers and print the sequence of integers between the two input integers, separated by commas and enclosed in square brackets. Display an increasing sequence if the first argument is smaller than the second; otherwise, display a decreasing sequence. If the two numbers are the same, that number should be displayed by itself.

Sample Run:
`Enter the first integer: 5`
 	
`Enter the second integer: 12`
`[5, 6, 7, 8, 9, 10, 11, 12]`
	Sample Run:
	`Enter the first integer: 17`
	
	`Enter the second integer: 10`
	`[17, 16, 15, 14, 13, 12, 11, 10]`

Sample Run:
`Enter the first integer: 115`
 	
`Enter the second integer: 115`
`[115 ]`	

Q3: Write a program, `Lab02_Q3.py`, which prompts the user to enter a string until the user enters the word 'exit' (not case-sensitive). For each string entered, the program outputs a modified version of the given phrase by removing adjacent duplicates. The outputs should appear as in the sample run.

Sample Run:

```
Enter a string (exit to stop): abccdddd
adjacent duplicates removed:  abcd

Enter a string (exit to stop): aa b cccccd eef
adjacent duplicates removed:  a b cd ef

Enter a string (exit to stop): good day all!
adjacent duplicates removed:  god day al!

Enter a string (exit to stop): aaaaa greattt.
adjacent duplicates removed:  a great.

Enter a string (exit to stop): exit
Bye!
```