ANA-522-OL1 Spring 2022
Mod01 Week01 Lab: Python Part One
Due: Friday January 14th at midnight

To complete each exercise you should expect to use some or all of these Python features:

- Generate output with print statements
- Read input, including casting that input to the appropriate type
- Perform calculations involving integers and floating point numbers using Python operators like +, -, *, /, //, %, and **
- Control how output is displayed using format specifiers

0.1 Mod01_Lab 1: Mailing Address

Create a program that displays your name and complete mailing address formatted in the manner that you would usually see it on the outside of an envelope. Your program does not need to read any input from the user.
>Hiint: Prepare each string line to be output individually, or multipline all together, to be printed with print statements.

0.2 Mod01_Lab 2: Area of a Room

Write a program that asks the user to enter the width and length of a room. Once the values have been read, your program should compute and display the area of the room. The length and the width will be entered as floating point numbers. Include units in your prompt and output message; either feet or meters, depending on which unit you are more comfortable working with.
>Hint: Use input statement to prompt users and receive input. Also see [Get User Input Interactive] from Mod00 lab example.

0.3 Mod01_Lab 3: Arithmetic Operations using Binary Operator

Create a program that reads two integers, a and b, from the user. Your program should compute and display:

- The sum of a and b
- The difference when b is subtracted from a
- The product of a and b
- The quotient when a is divided by b
- The remainder when a is divided by b
- The result of a^b

0.4 Mod01_Lab 4: Even or Odd?

Write a program that reads an integer from the user. Then your program should display a message indicating whether the integer is even or odd.
>Hint: See example on p.49 about how to construct the boolean condition whether a number if a multiple of 3 or 5. Then think about a multiple of 2.

0.5 Mod01_Lab 5:Vowel or Consonant

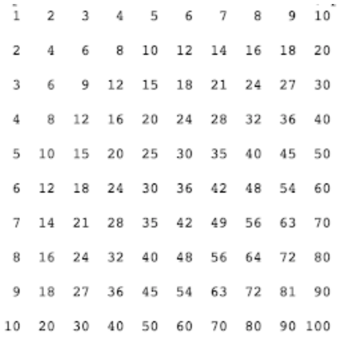
In this exercise you will create a program that reads a letter of the alphabet from the user. If the user enters a, e, i, o or u then your program should display a message indicating that the entered letter is a vowel. If the user enters y then your program should display a message indicating that sometimes y is a vowel, and sometimes y is a consonant. Otherwise your program should display a message indicating that the letter is a consonant.
>Hint: See control flow examples in p.46.

0.6 Mod01_Lab 6: Average

In this exercise you will create a program that computes the average of a collection of values entered by the user. The user will enter 0 as a sentinel value to indicate that no further values will be provided. Your program should display an appropriate error message if the first value entered by the user is 0.
>Hint: Because the 0 marks the end of the input it should not be counted and included in the average.

0.7 Mod01_Lab 7: Multiplication Table

In this exercise you will create a program that displays a multiplication table that shows the products of all combinations of integers from 1 times 1 up to and including 10 times 10. Your multiplication table should include a row of labels across the top of it containing the numbers 1 through 10. It should also include labels down the left side consisting of the numbers 1 through 10. The expected output from the program is shown below:



```
  1   2   3   4   5   6   7   8   9  10

  2   4   6   8  10  12  14  16  18  20

  3   6   9  12  15  18  21  24  27  30

  4   8  12  16  20  24  28  32  36  40

  5  10  15  20  25  30  35  40  45  50

  6  12  18  24  30  36  42  48  54  60

  7  14  21  28  35  42  49  56  63  70

  8  16  24  32  40  48  56  64  72  80

  9  18  27  36  45  54  63  72  81  90

 10  20  30  40  50  60  70  80  90 100
```

When completing this exercise you will probably find it helpful to be able to print out a value without moving down to the next line. This can be accomplished by added end="" as the last parameter to your print statement. For example, print(“A”) will display the letter A and then move down to the next line. The statement print(“A”, end="") will display the letter A without moving down to the next line, causing the next print statement to display its result on the same line as the letter A.

Hint: Use range() function to easily generate a list of numbers from 1 to 10 in order.