1   Length

How to print the length of the string literal 'abcd' ?

How to print the length of the variable x (x is the list [5, 30 ,2]) ?

What would be the length of following dictionary literal {'x': 3, 'y': 3} ?

What would be the length of the tuple literal ('x', 'y') ?

2   Strings

Ask the user for a string and print out whether this string is a palindrome (reads the same forwards and backwards) or not. Your program should output True or False.

3   Lists

3.1   Equal Lists

What is the result of each of the following? explain

[] == []

[] is []

3.2   Adding elements

1. Given the list li = [2, 0, 2, 2]
   a. Append to the end of the list the number 7
   b. Insert to the beginning of the list the number 1
   c. Insert in the second place the number 20
2. What will be the result of running the following line? li.insert(20, 0)

Hint: you will need to understand the .insert() and .append() methods for lists

https://www.programiz.com/python-programming/methods/list/insert

https://www.programiz.com/python-programming/methods/list/append

3.3   Lists - Min Max

How do you get the maximum and minimum values from a list? Write code that outputs these values for the given list: l = [3, 5, 7, 9, 2, 1, 4, 5, 10, 6]

Hint: you may need to use a search engine to find this answer

3.4   Sublists

Write a program that takes a list of numbers (for example, a = [5, 10, 15, 20, 25]) and makes a new list of only the first and last elements of the given list.

3.5   List Overlap

Take two lists, say for example these two:

 a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

 b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

and write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.

Hint: remember when we mentioned sets in class, which are similar to lists but don't have repeated elements? You will need to transform the list into a set using the "set()" function and the ".intersection()" method for sets.

4   While loop

4.1   Stream of Numbers

Use a while loop to

1. print the numbers between 0 and 10
2. print the number between 0 and -10 (in descending order)
3. print the numbers between 10 and 0 (in descending order)

4.2   Adding

Write a while loop that adds all the numbers up to 100 (inclusive).

4.3   Expanding list

Create an empty list and then use a while loop to populate it until it becomes the following list: [0, 1, 2, 3, 4]

4.4   Reverse

Write a while loop that will keep asking user for input, reverses and displays it until he enters the letter 'q' to quit.

4.5   Finding multiples

Ask the user for a number (integer x) and use a while loop to print its multiples (1 * x, 2* x, 3 * x, and so on, until 10 * x).

5   For loop

5.1   Every Character

Given a variable of a string, print character of the string on its own line.

5.2   Stream of Numbers

Use a for loop to

4. print the numbers between 0 and 10
5. print the number between 0 and -10 (in descending order)
6. print the numbers between 10 and 0 (in descending order)

5.3   Cumulative Sum

Given a list of integers, return the running sum of each element, so that if the input is:

[1, 5, 1, 3]

the output is:

[1, 6, 7, 10]

5.4 Fibonnaci

Write a program that asks the user how many Fibonnaci numbers to generate (*n*) and then generates them. The program should print a list with length *n*. The Fibonnaci seqence is a sequence of numbers where the next number in the sequence is the sum of the previous two numbers in the sequence. The sequence in a list with n = 7 looks like this:

[0, 1, 1, 2, 3, 5, 8, 13]

5.5 Right triangle

Write a for loop that makes seven calls to print(), so we get on the output the following triangle:

```
 #
 ##
 ###
 ####
 #####
 ######
 #######
```

6 Extras

6.1 Square

Use nested loops (loop within a loop) to create the following:

```
# # # # # # # #
# # # # # # # #
# # # # # # # #
# # # # # # # #
# # # # # # # #
# # # # # # # #
# # # # # # # #
# # # # # # # #
```

6.2 Even and odd numbers

  a) Use for loop to iterate from 0 to 100 and print only even numbers
  b) Use for loop to iterate from 0 to 100 and print only odd numbers

6.3 Reverse

This is a fruit list, ['banana', 'orange', 'mango', 'lemon'] reverse the order using a for loop.

Hint: you may need to use *range* and the *len* function