This Assignment is to help you develop your concept of Lists in Python.

Write the code in Python to do the following tasks:

Task 1

Write a Python program that reads 5 numbers from the user into a list. After reading each number, print all the numbers that have been entered so far in the list.

Example:
If the user enters 3, prints "Numbers in the list: [3]"
If the user enters 5 next, prints "Numbers in the list: [3, 5]"
If the user enters 34, prints "Numbers in the list: [3, 5, 34]"
If the user then enters -11, prints "Numbers in the list: [3, 5, 34, -11]"
Finally, if the user enters 0 as the last number, prints "Numbers in the list: [3, 5, 34, -11, 0]"

Task 2

Write a Python program that takes an input from the user, then creates a new list excluding the first and last two elements of the given list and prints the new list. If there are not enough elements in the list to do the task, print "Not possible".

Note: You may use list slicing.

==

Sample Input 1:
10, 20, 24, 25, 26, 35, 70

Sample Output 1:
[24, 25, 26]

==

Sample Input 2:
10, 20, 24, 25, 26

Sample Output 2:
[24]

==

Sample Input 3:
10, 20, 24, 25

Sample Output 3:
[]

==

Sample Input 4:
10, 20, 24

Sample Output 4:
Not possible

Task 3

Write a python program that reads 5 numbers from the user into a list, and then prints them in the reverse order.

Hint: You may create a list to store the input numbers and then use loop to print them in reverse order

==

Sample Input:
5
-5
100
1
0

Sample Output:
Printing values from the list in reverse order:
0
1
100
-5
5

Task 4

Write a Python program that turns every item of a list into its square. [Your program should work for any lists; make changes to the lists below and check whether your program works correctly]

==

Given list 1:
[1, 2, 3, 4, 5, 6, 7]
Sample Output 1:
[1, 4, 9, 16, 25, 36, 49]

==

Given list 2:
[3, 5, 1, 6]

Sample Output 2:
[9, 25, 1, 36]

Task 5

Write a Python program that removes all Empty strings from a **given list** and prints the modified list. [Your program should work for any given list; make changes to the list below and check whether your program works correctly]

Given List:

["hey", "there", "", "what's", "", "up", "", "?"]

Sample Output:

Original List: ['hey', 'there', '', "what's", '', 'up', '', '?']

Modified List: ['hey', 'there', "what's", 'up', '?']

Task 6

Write a Python program that reads a string containing 7 numbers separated by commas, then makes a list of those numbers and prints the **largest number** and its location or index position in the list. [You are not allowed to use the max(), sort(), sorted() function here]

[Your program should work for any given list; make changes to the list below and check whether your program works correctly]

Hint: You may assume the first input to be the largest value initially and the largest value's location to be 0.

Note: You may need to be careful while printing the output. Depending on your code, you might need data conversion.

==

Sample Input:
7, 13, 2, 10, 6, -11, 0

Sample Output:
My list: [7, 13, 2, 10, 6, -11, 0]
Largest number in the list is 13 which was found at index 1.

Task 7

Suppose you have been given two lists. Write a Python program that replaces the last element of the first list with the second list. [Your program should work for any two given lists; make changes to the lists below and check whether your program works correctly]

==

Sample given lists 1:
List_one : [1, 4, 7, 5]
List_two : [6, 1, 3, 9]

Sample Output 1:
[1, 4, 7, 6, 1, 3, 9]

==

Sample given lists 2:
List_one : [1, 3, 5, 7, 9, 10]
List_two : [2, 4, 6, 8]

Sample Output 2:
[1, 3, 5, 7, 9, 2, 4, 6, 8]

Task 8

Assume, you have been given two lists. [Your program should work for any two given lists; change the following lists and check whether your program works correctly for the code you have written]

list_one = [1, 2, 3, 4, 5, 6, 7, 8, 9]
list_two = [10, 11, 12, -13, -14, -15, -16]

Write a Python program that creates a new list with all the **even elements** of both of the given lists and prints the new list.

Hint: You may create a third list to store the even elements of the given lists.

==

Output for the above lists: [2, 4, 6, 8, 10, 12, -14, -16]

Task 9

Write a Python program that reads a string as an input from the user where multiple numbers are separated by spaces. Then, make a list of numbers from the input string without using the split() function and print the list. Finally, remove all the occurrences of **even numbers** from the same input list and print the modified list.

==

Sample Input:

7 12 4 55 96 2 11 61 33 42

Sample Output:

Original list: [7, 12, 4, 55, 96, 2, 11, 61, 33, 42]
Modified list: [7, 55, 11, 61, 33]

Task 10

Write a Python program that reads a string as an input from the user where multiple numbers are separated by commas. Then, make a list of numbers from the input string and print the list. Finally, remove multiple occurrences of any numbers from the input list and print the modified list **without duplicate values**.

Hint: You may create a third list to store the results but you may try doing it in the same input list. You can use membership operators (in, not in) to make sure no duplicates are added.

==

Sample Input 1:
0, 0, 1, 2, 3, 4, 4, 5, 6, 6, 6, 7, 8, 9, 4, 4

Sample Output 1:
Input list: [0, 0, 1, 2, 3, 4, 4, 5, 6, 6, 6, 7, 8, 9, 4, 4]
Modified list: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

==

Sample Input 2:
7, 7, 7, 1, 0, 3, 3, 55, 9

Sample Output 2:
Input list: [7, 7, 7, 1, 0, 3, 3, 55, 9]
Modified list: [7, 1, 0, 3, 55, 9]

Task 11

Assume, you have been given two lists: List_one and List_two. [Your program should work for any two given lists; change the following lists and check whether your program works correctly for the code you have written]

Write a Python program that prints "True", if the given two lists have at least one common member. Otherwise, print "False".

Note: Please use concepts of flag and break to solve this task.

==

Given lists 1:
List_one : [1, 4, 3, 2, 6]
List_two : [5, 6, 9, 8, 7]

Sample Output 1:
True

==

Given lists 2:
List_one : [1, 4, 3, 2, 5]
List_two : [8, 7, 6, 9]

Sample Output 2:
False

Task 12

1	`myList = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]`
2	`index1 = 0`
3	`index2 = 0`
4	`index1 = 1`
5	`while(index1<10):`
6	`    myList[index1] = index1+4`
7	`    index2 = 1`
8	`    while(index2<index1):`
9	`        myList[index1] = myList[index1] + myList[index2]-index1`
10	`        index2 = index2+1`
11	`    print(myList[index1])`
12	`    index1 = index1+1`

Output

Task 13

1	`myList = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]`
2	`index1 = 0`
3	`index2 = 0`
4	`index1 = 1`
5	`while (index1 < 10):`
6	`  myList[index1] = index1 + 4`
7	`  index2 = 1`
8	`  while (index2 < index1):`
9	`    myList[index1] = myList[index1-1] - myList[index2-1] - index1`
10	`    index2 = index2 + 1`
11	`  print(myList[index1])`
12	`  index1 = index1 + 1`

Output
5
3
-5
-12
-12

Task 14

1	`myList = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]`
2	`b = []`
3	`index1 = 0`
4	`index2 = 0`
5	`index1 = 1`
6	`b = myList`
7	`while(index1<10):`
8	`   myList[index1] = index1+2`
9	`   index2 = 1`
10	`   while(index2<index1):`
11	`      myList[index1] = b[index1]+myList[index2]-index1`
12	`      index2 = index2+1`
13	`   print(myList[index1])`
14	`   index1 = index1+1`

Output

Task 15

1	`myList = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]`
2	`b = []`
3	`index1 = 0`
4	`index2 = 0`
5	`index1 = 1`
6	`b = myList`
7	`while (index1 < 10):`
8	`    myList[index1] = index1 + 1`
9	`    index2 = 1`
10	`    while (index2 < index1):`
11	`        myList[index1] = b[index2-1] + myList[index2] - index1`
12	`        index2 = index2 + 1`
13	`    print(myList[index1])`
14	`    index1 = index1 + 1`

Output

Optional Tasks (16-21) [Ungraded]

Task 16

Write a Python program that reads 7 numbers into a list and prints the **second largest number** and its location or index position on the list. [You are not allowed to use the max(), sort(), sorted() function here]

==

Sample Input:
7, 13, 2, 10, 6, -11, 0

Sample Output:
My list: [7, 13, 2, 10, 6, -11, 0]
Second largest number in the list is 10 which was found at index 3.

Task 17

Write a Python program that reads 5 numbers into a list and prints the smallest and largest number and their locations in the list. [You are not allowed to use the max(), min(), sort(), sorted() functions here]

Hint: You may assume the first input to be the largest value initially and the largest value's location to be 0. Similarly, you can assume the first input to be the smallest value initially and the smallest value's location to be 0.

Note: You may need to be careful while printing the output. Depending on your code, you might need data conversion.

==

Sample Input:
7, 13, -5, 10, 6

Sample Output:
My list: [7, 13, -5, 10, 6]
Smallest number in the list is -5 which was found at index 2
Largest number in the list is 13 which was found at index 1

Task 18

Write a Python program that takes two lists as an input from the user. Then print a new list with the **common elements** of both the input lists.

Hint: You may need to create a third list to store the results. You can use membership operators (in, not in) to make sure similar elements are added.

==

Sample Input 1:
A, B, C, D

C, E, F, B

Sample Output 1:
['C', 'B']

==

Sample Input 2:
1, 3, A, H, P

A, G, 1, P, O

Sample Output 2:
['1', 'A', 'P']

Task 19

Assume, you have been given two lists. [Your program should work for any two given lists; make changes to the lists below and check whether your program works correctly]

list_one = [1, 2, 2, 4, 5, 5, 7, 99, 200, 303, 70]
list_two = [1, 1, 2, 3, 3, 3, 4, 5, 200, 500, -5]

Write a Python program that creates a new list with all the **unique elements** of both the given lists. **You need to make sure that there are no duplicates in the resulting list.** Finally, print the updated list.

Hint: You may create a third list to store the results. You can use membership operators (in, not in) to make sure no duplicates are added.

==

Output for the above two lists: [1, 2, 4, 5, 7, 99, 200, 303, 70, 3, 500, -5]

Task 20

Write a python program to take a list as a string input from the user and print it back to the user as a list. Please look at the examples below for clarification.

[Must use string split() and strip() functions]

==

Sample Input 1:
'1,   2 ,   3, 50, 4'

Sample Output 1:
Original data: 1,   2 ,   3, 50, 4
After removing square brackets: 1,   2 ,   3, 50, 4
Numbers in string format with extra white spaces: ['1', '   2 ', '   3', '50', '4']
Final data (numbers in list format): [1, 2, 3, 50, 4]

==

Sample Input 2: '[1,   2 ,   3, 50, 4]'

Sample Output 2:
Original data: [1,   2 ,   3, 50, 4]
After removing square brackets: 1,   2 ,   3, 50, 4
Numbers in string format with extra white spaces: ['1', '   2 ', '   3', '50', '4']
Final data (numbers in list format): [1, 2, 3, 50, 4]

==

Sample Input 3:
"   [1, 2 ,   3, 50, 4]   "

Sample Output 3:
Original data: [1, 2 ,   3, 50, 4]
After removing square brackets: 1, 2 ,   3, 50, 4
Numbers in string format with extra white spaces: [' 1', ' 2 ', '   3', ' 50', ' 4']
Final data (numbers in list format): [1, 2, 3, 50, 4]

Task 21

Write a Python program that takes a single string as an input from the user where few numbers are separated by commas. Now, make a list with the numbers of the given string. Then your task is to remove multiple occurrences of any number and then finally print the list **without any duplicate values**.

Hint (1): For obtaining the numbers from the string, use split(). For cleaning the data, use strip().

Hint (2): You may create a third list to store the results. You can use membership operators (in, not in) to make sure no duplicates are added.

==

Sample Input 1:
0, 0, 1, 2, 3, 4,   4, 5, 6, 6, 6,   7, 8, 9, 4,   4

Sample Output 1:
Given numbers in list: [0, 0, 1, 2, 3, 4, 4, 5, 6, 6, 6, 7, 8, 9, 4, 4]
List without any dupliacte values: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

==

Sample Input 2:
7, 7, 7, 1,   0, 3, 3, 55, 9

Sample Output 2:
Given numbers in list: [7, 7, 7, 1, 0, 3, 3, 55, 9]
List without any dupliacte values: [7, 1, 0, 3, 55, 9]