

1 ICS 104 - Introduction to Programming in Python and C

1.1 Loops - Lab 2

- SOLUTIONS

2 Lab Objectives

- To understand nested loops
- To process strings

3 Exercises

3.1 Exercise # 1:

- Write a python nested loops code that prints the table in Ex1 below.
- Then, with minimum change to the original code, generate the remaining outputs of Ex2 to Ex6.

#Ex1

```

1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5

```

#Ex2

```

1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5

```

#Ex3

```

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

```

#Ex4

```

1 2 3 4 5
2 3 4 5
3 4 5
4 5
5

```

#Ex5

```

*****
****
***
**
*

```

#Ex6

```

1 *****
2 ****
3 ***
4 **
5 *

```

In [1]:

```
1  %%writefile lab07Ex1_YourID.py
2  ## Uncomment the above line after you finish your code and want to save it in a file.
3
4  # YOUR CODE HERE
5
6  for i in range(1, 6):
7      for j in range (1, 6):
8          print(i, end="\t")
9      print("")
10
11 print(" ")
12
13 for i in range(1, 6):
14     for j in range (1, 6):
15         print(j, end="\t")
16     print("")
17
18 print(" ")
19
20 for i in range(1, 6):
21     for j in range (1, i + 1):
22         print(j, end="\t")
23     print("")
24
25 print(" ")
26
27 for i in range(1, 6):
28     i=i-1
29     for j in range (i+1,6,1):
30         print(j, end="\t")
31     print("")
32
33 print(" ")
34
35 for i in range(1, 6):
36     i=i-1
37     for j in range (i+1,6,1):
38         j= "*"
39         print(j, end="\t")
40     print("")
41
42 print(" ")
```

```

43
44 for i in range(1, 6):
45     print(i, end="\t")
46     i=i-1
47     for j in range (i+1,6,1):
48         j= "*"
49         print(j, end="\t")
50     print("")

```

```

1      1      1      1      1
2      2      2      2      2
3      3      3      3      3
4      4      4      4      4
5      5      5      5      5

```

```

1      2      3      4      5
1      2      3      4      5
1      2      3      4      5
1      2      3      4      5
1      2      3      4      5

```

```

1
1      2
1      2      3
1      2      3      4
1      2      3      4      5

```

```

1      2      3      4      5
2      3      4      5
3      4      5
4      5
5

```

```

*      *      *      *      *
*      *      *      *
*      *      *
*      *
*

```

```

1      *      *      *      *      *
2      *      *      *      *
3      *      *      *
4      *      *
5      *

```

3.2 Exercise # 2:

Write a program that asks the user for a collection of N numbers. Then, it will find the smallest value and its frequency.

- Get the value of N from the user.
- If $N \leq 0$, display an appropriate error message and terminate the program; otherwise
- Read the values as entered from the user. (If $N = 5$, then there are 5 values the user is going to enter).
- Find the smallest, its frequency (how many times it is repeated).
- **Hint:** To find the smallest value, read the first value **before the loop** then assume that this first value entered by the user is the smallest value. **Within the loop** compare the assumed smallest value with other values **read inside the loop** and change the smallest value accordingly.

3.2.1 Two sample runs

```
Enter how many values to consider: 7
Enter value 1: -1
Enter value 2: 2
Enter value 3: 8
Enter value 4: 9
Enter value 5: -1
Enter value 6: 6
Enter value 7: 5
*****
Minimum value= -1
It's frequency is 2
```

```
Enter how many values to consider: -2
Wrong input. Input must be > 0
```

In [9]:

```
1  %%writefile lab07Ex2.py
2  ## Uncomment the above line after you finish your code and want to save it in a file.
3
4  # YOUR CODE HERE
5
6  numOfValues = int(input("Enter how many values to consider: "))
7
8  if numOfValues > 0:
9      for i in range(1, numOfValues + 1):
10         value = int(input("Enter value %d: " % i))
11         if i == 1:
12             minimum = value
13             frequency = 1
14         elif value < minimum:
15             minimum = value
16             frequency = 1
17         elif minimum == value:
18             frequency = frequency + 1
19     print(""*15)
20     print("Minimum Values= ", minimum)
21     print("It's frequency is ", frequency)
22 else:
23     print("Wrong Input. Input must be > 0")
24
```

Enter how many values to consider: -2

Wrong Input. Input must be > 0

3.3 Exercise # 3:

Write a program that reads a string then generates and prints a new string that has same characters of the original string but with alternating capital and small letters. For example, check the sample RUN below.

3.3.1 Sample run

Enter a statement: Easy Summer
you mean:

EaSy sUmMeR

In [10]:

```
1  %%writefile lab07Ex3.py
2  ## Uncomment the above line after you finish your code and want to save it in a file.
3
4  # YOUR CODE HERE
5
6  counter = 0
7  statement = input("Enter a statement: ")
8  print("you mean: ")
9  for i in statement:
10     if counter%2 == 0:
11         i = i.upper()
12     else:
13         i = i.lower()
14     counter = counter + 1
15     print(i, end = "")
```

Enter a statement: Easy Summer
you mean:
EaSy sUmMeR

3.4 Exercise # 4:

Write a program that reads a word and prints the number of vowels in the word. For this exercise, assume that **a e i o u y** are vowels. For example, if the user provides the input "Harry", the program prints 2 vowels.

Sample Runs

Enter a word: Accounting
There are 4 vowels in Accounting

Enter a word: LG
There are 0 vowels in LG

```
In [1]: 1  ###writefile lab07Ex4_YourID.py
2  ## Uncomment the above line after you finish your code and want to save it in a file.
3
4  # YOUR CODE HERE
5
6  word = input("Enter a word: ")
7  counter = 0
8  for i in range(len(word)):
9      if word[i] == "A" or word[i] == "a" or word[i] == "E" or word[i] == "e" or word[i] == "I" or word[i] ==
10         counter = counter + 1
11  print("There are %d vowels in %s" %(counter, word))
```

Enter a word: LG

There are 0 vowels in LG

4 End of the Loops - Lab 2

Good luck...