

15/10/2025

CSA-0805-PYTHON PROGRAM

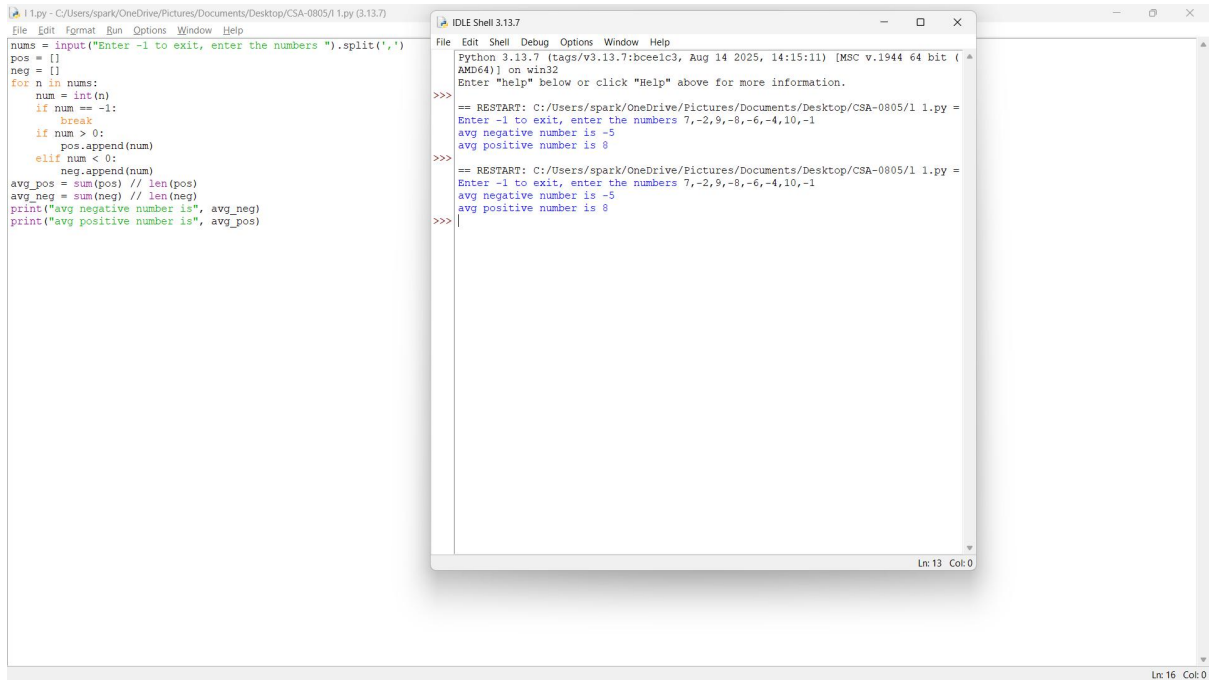
LEVEL- 1

1. Read the number until -1 is encounter. find the avg of positive numbers and negative numbers entered by user

Sample Input:

Enter -1 to exit, enter the numbers 7,-2,9,-8,-6,-4,10,-1

Output: avg negative number is -5, avg positive number is 8



The screenshot shows a Python IDE with two windows. The left window displays the source code for a program that calculates the average of positive and negative numbers entered by the user until -1 is entered. The right window shows the execution output, which matches the sample input and output provided in the problem statement.

```
l1.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1.py (3.13.7)
File Edit Format Run Options Window Help
nums = input("Enter -1 to exit, enter the numbers ").split(',')
pos = []
neg = []
for n in nums:
    num = int(n)
    if num == -1:
        break
    if num > 0:
        pos.append(num)
    elif num < 0:
        neg.append(num)
avg_pos = sum(pos) // len(pos)
avg_neg = sum(neg) // len(neg)
print("avg negative number is", avg_neg)
print("avg positive number is", avg_pos)
```

```
IDLE Shell 3.13.7
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1.py =
Enter -1 to exit, enter the numbers 7,-2,9,-8,-6,-4,10,-1
avg negative number is -5
avg positive number is 8
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1.py =
Enter -1 to exit, enter the numbers 7,-2,9,-8,-6,-4,10,-1
avg negative number is -5
avg positive number is 8
>>>
```

1. Write a python program to find the square, cube of the given decimal number. Sample Input:

Given Number: 0.6

Output:

Square Number: 0.36

Cube

Number:0.216

```
l2.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l2.py (3.13.7)
File Edit Format Run Options Window Help
num = float(input("Given Number: "))
square = num ** 2
cube = num ** 3
print("Square Number:", round(square, 3))
print("Cube Number:", round(cube, 3))

IDLE Shell 3.13.7
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)]
on win32
Enter "help" below or click "Help" above for more information.

>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 1.py =
Enter -1 to exit, enter the numbers 7,-2,9,-8,-6,-4,10,-1
avg negative number is -5
avg positive number is 8
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 1.py =
Enter -1 to exit, enter the numbers 7,-2,9,-8,-6,-4,10,-1
avg negative number is -5
avg positive number is 8
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 2.py =
Given Number: 0.6
Square Number: 0.36
Cube Number: 0.216
>>>
===== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 2.py =====
Given Number:
===== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 2.py =====
Given Number: 0.6
Square Number: 0.36
Cube Number: 0.216
>>>
```

2. Write a python program to print the following pattern.

Sample Input:

Enter the Character to be printed:+

Number of rows.: 5

Output:

```
+
++
+++
++++
+++++
```

```
l3.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l3.py (3.13.7)
File Edit Format Run Options Window Help
char = input("Enter the Character to be printed: ")
rows = int(input("Number of rows.: "))
for i in range(1, rows + 1):
    print((char + ' ') * i)

IDLE Shell 3.13.7
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l1 3.py =
Enter the Character to be printed: +
Number of rows.: 5
+
++
+++
++++
+++++
>>>
```

3. Python Program to Display the Multiplication Table

Sample Input:

A=7

B=5

Output:

7 x 1 = 7

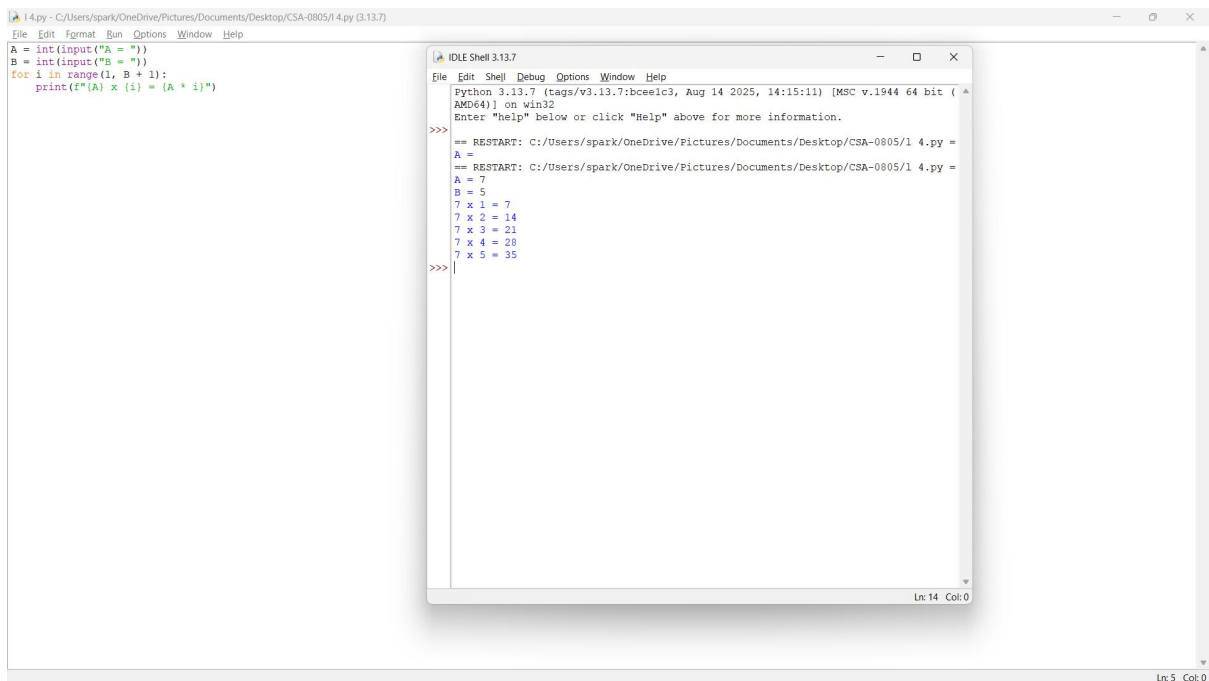
7 x 2 = 14

7 x 3 = 21

7 x 4 = 28

7 x 5 =

35



The screenshot shows a Python IDE window titled 'Python 3.13.7'. The main editor displays the following code:

```
A = int(input("A = "))
B = int(input("B = "))
for i in range(1, B + 1):
    print(f"{A} x {i} = {A * i}")
```

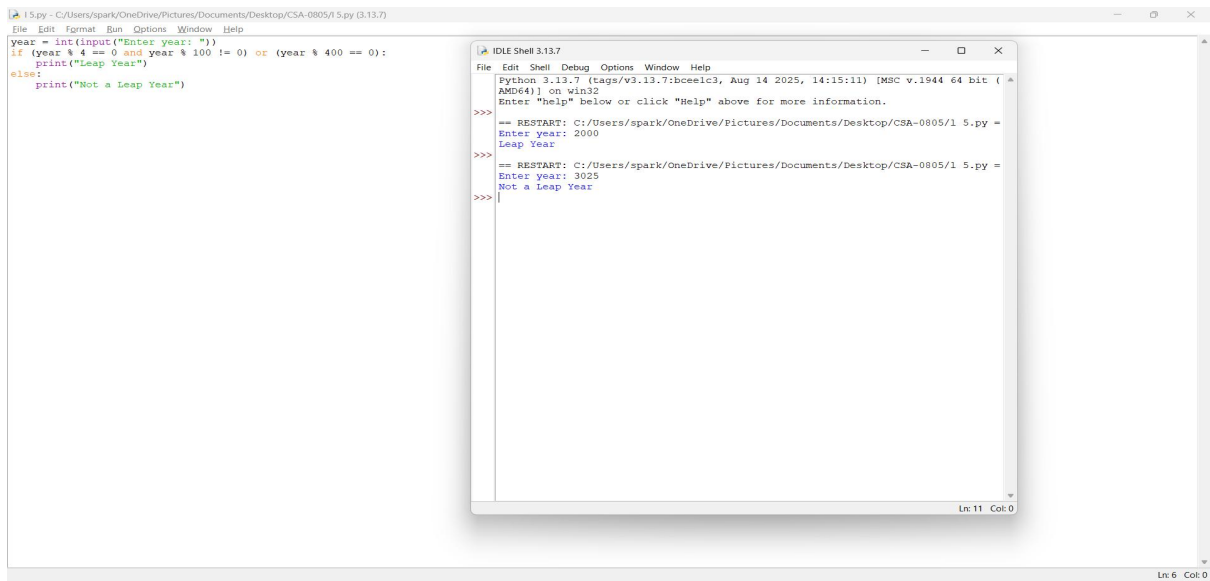
The output window shows the execution of the program. It displays the prompts and inputs for A and B, followed by the multiplication table output:

```
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/1 4.py ==
A =
7
B =
5
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
>>>
```

4. Write a program to find whether it is leap year or not?

Sample Input: 2000

Output: Leap Year



The screenshot shows a Python IDE with a file named `l5.py` containing a leap year program. The code checks if a year is a leap year based on the rules: divisible by 4 but not by 100, or divisible by 400. The IDE Shell shows the program being executed twice: first with input 2000, which is a leap year, and then with input 3025, which is not a leap year.

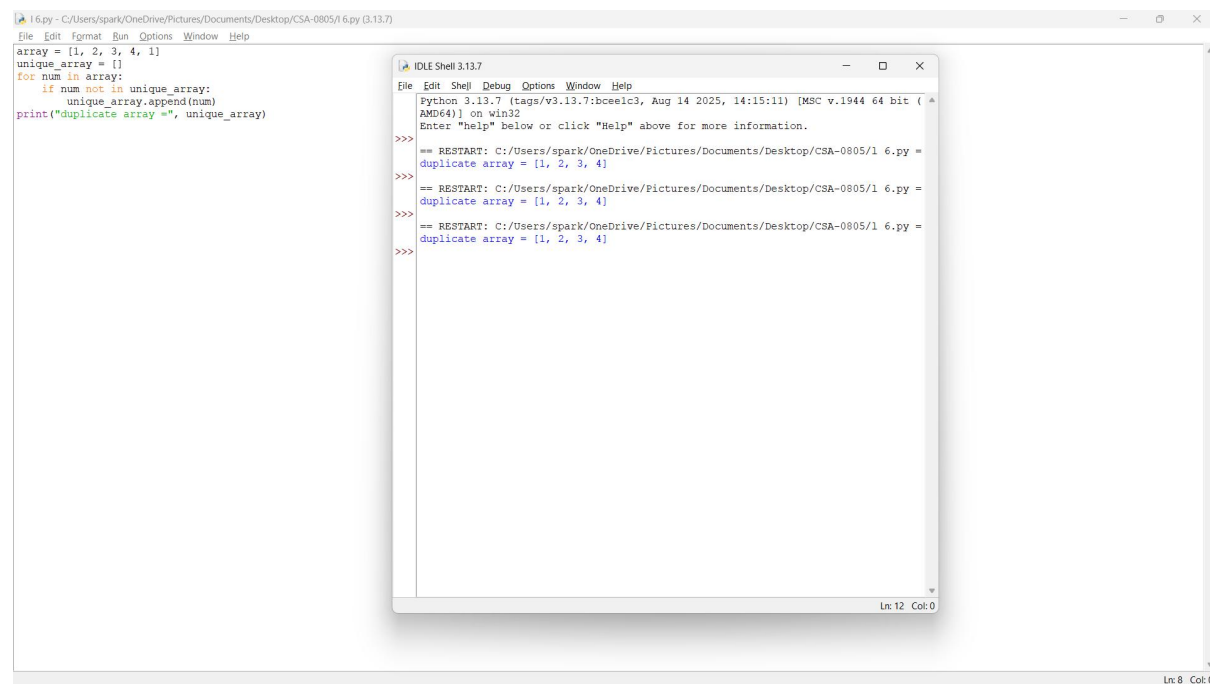
```
l5.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l5.py (3.13.7)
File Edit Format Run Options Window Help
year = int(input("Enter year: "))
if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
    print("Leap Year")
else:
    print("Not a Leap Year")

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> == RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l5.py =
Enter year: 2000
Leap Year
>>> == RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l5.py =
Enter year: 3025
Not a Leap Year
>>>
```

5. Write a program to find out the duplicate array

Sample Input: array={1,2,3,4,1}

1. Output: duplicate array={1,2,3,4}



The screenshot shows a Python IDE with a file named `l6.py` containing a program to find duplicate elements in an array. The program uses a list to track unique elements and prints the duplicate array. The IDE Shell shows the program being executed three times, each time with the same input array [1, 2, 3, 4, 1], resulting in the output [1, 2, 3, 4].

```
l6.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l6.py (3.13.7)
File Edit Format Run Options Window Help
array = [1, 2, 3, 4, 1]
unique_array = []
for num in array:
    if num not in unique_array:
        unique_array.append(num)
print("duplicate array =", unique_array)

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> == RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l6.py =
duplicate array = [1, 2, 3, 4]
>>> == RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l6.py =
duplicate array = [1, 2, 3, 4]
>>> == RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l6.py =
duplicate array = [1, 2, 3, 4]
>>>
```

6. Check whether the number is positive or negative

Sample Input:23

Output: positive

```
l7.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l7.py (3.13.7)
File Edit Format Run Options Window Help
num = int(input("Enter a number: "))
if num > 0:
    print("positive")
elif num < 0:
    print("negative")
else:
    print("zero")

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l7.py ==
Enter a number: 32
positive
>>>
```

7. Write a python program to find the average of mean median mode

Sample Input: [12,45,83,52]/4

1. Output: utput:48

```
l8.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l8.py (3.13.7)
File Edit Format Run Options Window Help
data = [12, 45, 83, 52]
mean = sum(data) / len(data)
data.sort()
n = len(data)
if n % 2 == 0:
    median = (data[n//2 - 1] + data[n//2]) / 2
else:
    median = data[n//2]
mode = None
for i in data:
    if data.count(i) > 1:
        mode = i
        break
if mode is None:
    mode = data[0]
average = (mean + median + mode) / 3
print("Output:", int(average))

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/l8.py ==
Output: 36
>>>
```

8. Write a python program to store the arrays in non-increasing order

Sample Input:[1,8,3,4,0]

1. Output:[8,4,3,1,0]

```
9.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/9.py (3.13.7)
File Edit Format Run Options Window Help
input_str = input("Enter array : ")
input_str = input_str.strip(" ")
arr = list(map(int, input_str.split(',')))
arr.sort(reverse=True)
print("Output:", arr)

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/9.py =
Enter array : [1,8,3,4,0]
Output: [8, 4, 3, 1, 0]
>>>
```

9. Write a Python Program to Intersecting an elements

Sample Input:

(2,3,4,5)

(3,4,8,6)

Output:

1. (3,4)

```
10.py - C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/10.py (3.13.7)
File Edit Format Run Options Window Help
a = (2, 3, 4, 5)
b = (3, 4, 8, 6)
result = tuple(set(a) & set(b))
print("Output:", result)

IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceec13, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
== RESTART: C:/Users/spark/OneDrive/Pictures/Documents/Desktop/CSA-0805/10.py =
Output: (3, 4)
>>>
```