

Name: Beshair Khan

Std ID: BIT-24S-006

GitHub Link: <https://github.com/Beshair-Khan/Python-Lab>

Lab 01

Task 1: Make 2-2 programs of each datatype.

- **Integers:**

Program 1: Add and multiply two integers.

```
Lab manuals > lab task1.py > ...
1  a=10
2  b=12
3  sum=a+b
4  print("The sum of a and b is:",sum)
5  product=a*b
6  print("The product of a and b is:",product)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
The sum of a and b is: 22
The product of a and b is: 120
PS D:\Uni programs>
```

Program 2: Difference of two integers

```
a=int(input("Enter number 1: "))
b=int(input("Enter number 2: "))
print("The differene of a and b is :",a-b)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Enter number 1: 20
Enter number 2: 45
The differene of a and b is : -25
PS D:\Uni programs> 
```

- **Float:**

Program 1: Find average and divide two floating point numbers.

```
num1 = 7.5
num2 = 2.5
num3=3.5
division = num1 / num2
average=(num1+num2+num3)/3
print("Division is: ", division)
print("Average is: ",average)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Division is:  3.0
Average is:  4.5
PS D:\Uni programs> 
```

Program 2: Find difference of two floating point numbers.

```
a=float(input("Enter number 1: "))
b=float(input("Enter number 2: "))
print("The differene of a and b is :",a-b)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Enter number 1: 8.5
Enter number 2: 4.7
The differene of a and b is : 3.8
PS D:\Uni programs> 
```

- **String:**

Program 1: Find length of a string.

```
message = "Hello, Python!"
length = len(message)
print("Length of message:", length)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Length of message: 14
PS D:\Uni programs> 
```

Program 2: Concatenate two strings.

```
first_name = "Beshair"
last_name = "Khan"
full_name = first_name + " " + last_name
print("Full Name:", full_name)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS C:\Users\TFS> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Full Name: Beshair Khan
PS C:\Users\TFS>
```

- **Boolean:**

Program 1: Check if number is positive.

```
num = 10
if num > 0:
    print("Number is positive")
else:
    print("Number is negative")
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Number is positive
PS D:\Uni programs\Lab manuals>
```

Program 2: if it's raining, tell the user to take an umbrella.

```
is_raining = True

if is_raining:
    print("Take an umbrella!")
else:
    print("Enjoy the sunshine!")
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Take an umbrella!
PS D:\Uni programs\Lab manuals>
```

- **List:**

Program 1: Create and print a list.

```
fruits = ["apple", "banana", "cherry"]  
print("Fruits:", fruits)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  
  
PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"  
Fruits: ['apple', 'banana', 'cherry']  
PS D:\Uni programs\Lab manuals>
```

Program 2: Add items to list.

```
numbers = [1, 2, 3]  
numbers.append(4)  
print("Updated numbers:", numbers)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  
  
PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"  
Updated numbers: [1, 2, 3, 4]  
PS D:\Uni programs\Lab manuals>
```

- **Tuple:**

Program 1: Create and access tuple elements.

```
colors = ("red", "green", "blue")  
print("First color:", colors[0])  
print("Second color:", colors[1])  
print("Third color:", colors[2])
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
First color: red
Second color: green
Third color: blue
PS D:\Uni programs\Lab manuals>
```

Program 2: Find length of tuple.

```
animals = ("cat", "dog", "rabbit")
print("Number of animals:", len(animals))
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Number of animals: 3
PS D:\Uni programs\Lab manuals>
```

- **Set:**
Program 1: Create and print a set

```
lab task1.py > ...
1  unique_numbers = {1, 2, 3, 4}
2  print("Unique numbers:", unique_numbers)
3
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Unique numbers: {1, 2, 3, 4}
PS D:\Uni programs\Lab manuals>
```

Program 2: Add an element to a set

```
my_set = {5, 6, 7}
my_set.add(8)
print("Updated set:", my_set)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Updated set: {8, 5, 6, 7}
PS D:\Uni programs\Lab manuals>
```

- **Dictionary:**
Program 1: Create and print a dictionary

```
student = {"name": "Beshair", "age": 19}
print("Student info:", student)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Student info: {'name': 'Beshair', 'age': 19}
PS D:\Uni programs\Lab manuals>
```

Program 2: Access value by key

```
car = {"brand": "Toyota", "model": "Corolla"}
print("Car brand:", car["brand"])
print("car model: ", car["model"])
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
Car brand: Toyota
car model: Corolla
PS D:\Uni programs\Lab manuals>
```

Task 2: Make up to 5 Shape programs using *.

1. Square:

```
for i in range(5):  
    print("* " * 5)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  
  
PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"  
* * * * *  
* * * * *  
* * * * *  
* * * * *  
* * * * *  
PS D:\Uni programs\Lab manuals>
```

2. Right Angled Triangle:

```
print (""  
    *  
    **  
    ***  
    ****  
    *****  
    ***** ""
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  
  
PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"  
  
    *  
    **  
    ***  
    ****  
    *****  
    *****  
PS D:\Uni programs\Lab manuals>
```

3. Diamond:

```
print (""  
    *  
    **  
    ***  
    ****  
    ***  
    **  
    * ""
```


Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"

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

PS D:\Uni programs\Lab manuals>
```

4. Pyramid:

```
print ("""
*
**
***
****
*****
***** """)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"

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

PS D:\Uni programs\Lab manuals>
```

5. Half Diamond:

```
for i in range(1, 6):
    print("* " * i)
for i in range(4, 0, -1):
    print("* " * i)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

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

PS D:\Uni programs\Lab manuals>
```

Task 3: Make same shapes you have made in task 2, using * mutiple by number.

1. Square:

```
print ("*" * 5)
print ("*" * 5)
print ("*" * 5)
print ("*" * 5)
print ("*" * 5)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
*****
*****
*****
*****
*****

PS D:\Uni programs\Lab manuals>
```

2. Right Angled Triangle:

```
print ("*" * 1)
print ("*" * 2)
print ("*" * 3)
print ("*" * 4)
print ("*" * 5)
```

Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
*
**
***
****
*****

PS D:\Uni programs\Lab manuals>
```

3. Diamond:

```
print (" " * 6 + " " * "")
print (" " * 5 + " " * " * 3)
print (" " * 4 + " " * " * 4)
print (" " * 3 + " " * " * 5)
print (" " * 2 + " " * " * 6)
print (" " * 3 + " " * " * 5)
print (" " * 4 + " " * " * 4)
print (" " * 5 + " " * " * 3)
print (" " * 6 + " " * "")
```

Output

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    COMMENTS

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

PS D:\Uni programs\Lab manuals>
```

4. Pyramid:

```
print (" " * 6 + " " * 11)
print (" " * 5 + " " * 13 * 3)
print (" " * 4 + " " * 15 * 4)
print (" " * 3 + " " * 17 * 5)
print (" " * 2 + " " * 19 * 6)
print (" " * 1 + " " * 21 * 7)
print (" " * 0 + " " * 23 * 8)
```

Output

```
PS D:\Uni programs\Lab manuals> & D:/Python/python.exe "d:/Uni programs/Lab manuals/lab task1.py"
```

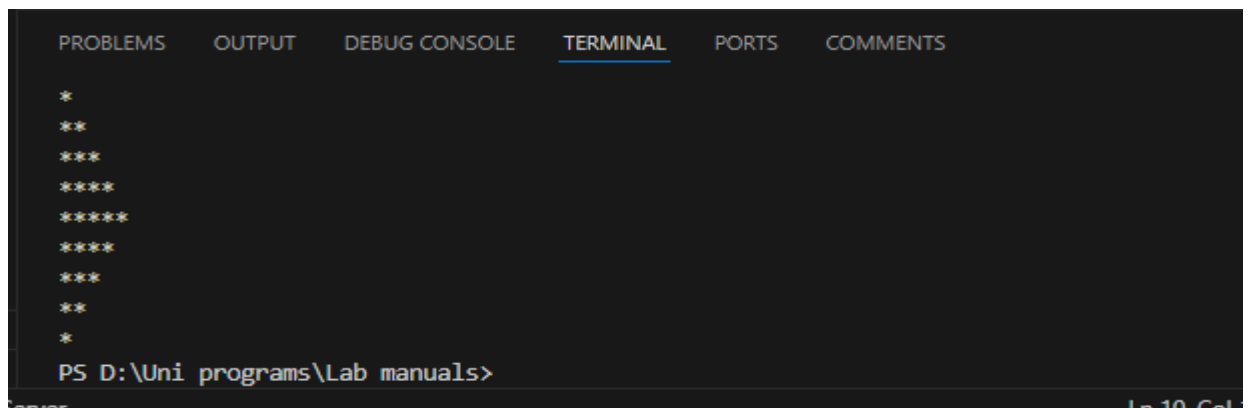
```
      *  
    * * *  
  * * * *  
* * * * *  
* * * * *  
* * * * *  
* * * * *
```

```
PS D:\Uni programs\Lab manuals>
```

5. Half Diamond:

```
print ("*")  
print ("*" * 2)  
print ("*" * 3)  
print ("*" * 4)  
print ("*" * 5)  
print ("*" * 4)  
print ("*" * 3)  
print ("*" * 2)  
print ("*")
```

Output



The screenshot shows a terminal window with a dark background. At the top, there are tabs labeled "PROBLEMS", "OUTPUT", "DEBUG CONSOLE", "TERMINAL" (which is selected and underlined), "PORTS", and "COMMENTS". Below the tabs, the output of the program is displayed as a half diamond pattern of asterisks: a single asterisk on the first line, followed by two, three, four, five, four, three, two, and finally a single asterisk on the ninth line. At the bottom of the terminal, the command prompt shows the path "PS D:\Uni programs\Lab manuals>".

```
*  
**  
***  
****  
*****  
****  
***  
**  
*  
  
PS D:\Uni programs\Lab manuals>
```