

Assignment 1



**Department of Electrical and Computer Engineering
North South University**

CSE445

Sec: 04

Submitted by

Anik Roy Pranto

1921219042

Assignment 1

Task 1:

Code:

```
import random
import turtle
def draw_circle():
    t = turtle.Turtle()
    s = turtle.Screen()
    t.goto(random.randint(-200,0),random.randint(0,200))
    t.color("violet")
    t.begin_fill()
    t.circle(100)
    t.end_fill()

    s.exitonclick()

def draw_rectangle():
    t = turtle.Turtle()
    s = turtle.Screen()
    t.goto(random.randint(-200, 0), random.randint(0, 200))
    t.color("violet")
    t.begin_fill()

    t.forward(100)
    t.left(90)

    t.forward(80)
    t.left(90)

    # drawing third side
    t.forward(100)
    t.left(90)

    # drawing fourth side
    t.forward(80)
    t.left(90)
    t.end_fill()

    s.exitonclick()

def draw_triangle():
    t = turtle.Turtle()
    s = turtle.Screen()
    t.goto(random.randint(-200, 0), random.randint(0, 200))
    t.color("violet")
    t.begin_fill()

    for i in range(3):
        t.forward(150)
```

```

        t.left(120)
    t.end_fill()

    s.exitonclick()

print("1) Enter 'circle' or 'Circle' or 'CIRCLE' to draw a circle at a random location")
print("2) Enter 'rectangle' or 'Rectangle' or 'RECTANGLE' to draw a rectangle at a random location")
print("3) Enter 'triangle' or 'Triangle' or 'TRIANGLE' to draw a triangle at a random location")

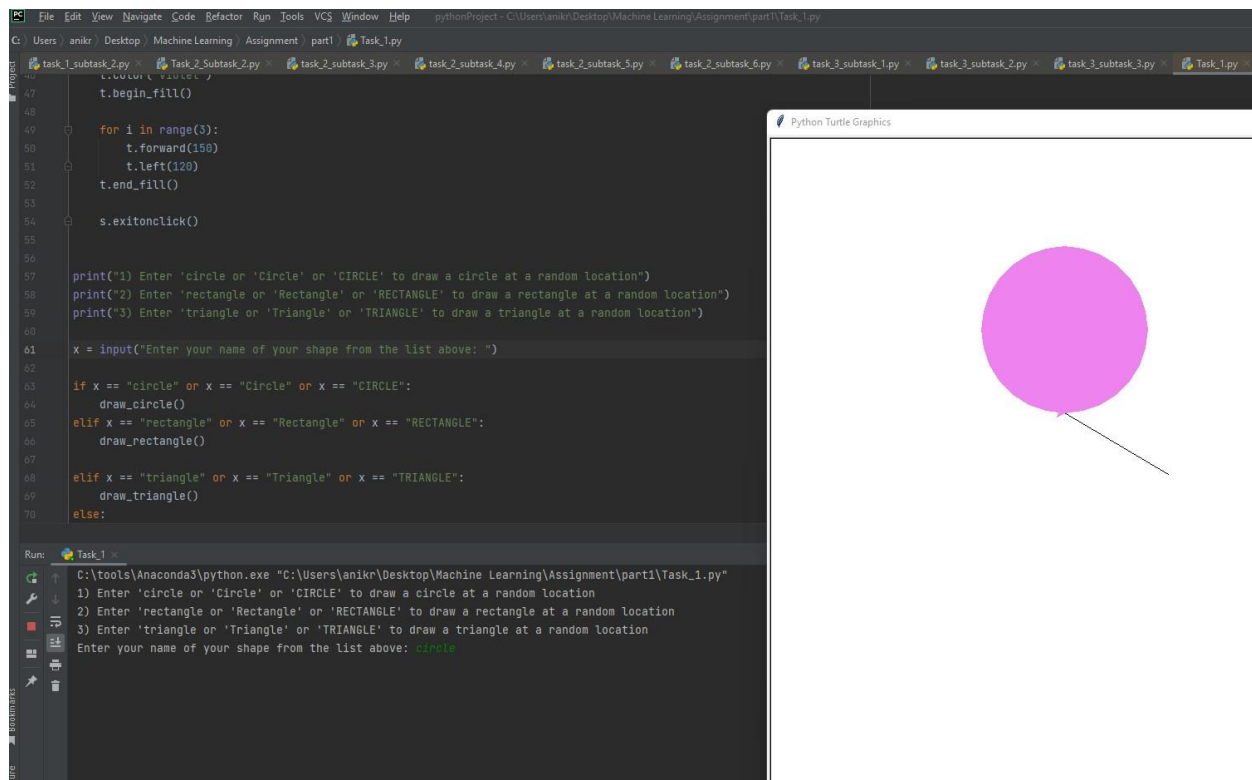
x = input("Enter your name of your shape from the list above: ")

if x == "circle" or x == "Circle" or x == "CIRCLE":
    draw_circle()
elif x == "rectangle" or x == "Rectangle" or x == "RECTANGLE":
    draw_rectangle()

elif x == "triangle" or x == "Triangle" or x == "TRIANGLE":
    draw_triangle()
else:
    print("Choice not listed above")

```

Output:



Task 2:

Code:

```
def add_numbers(x1,x2):
    print("Result of addition operation:",x1+x2)

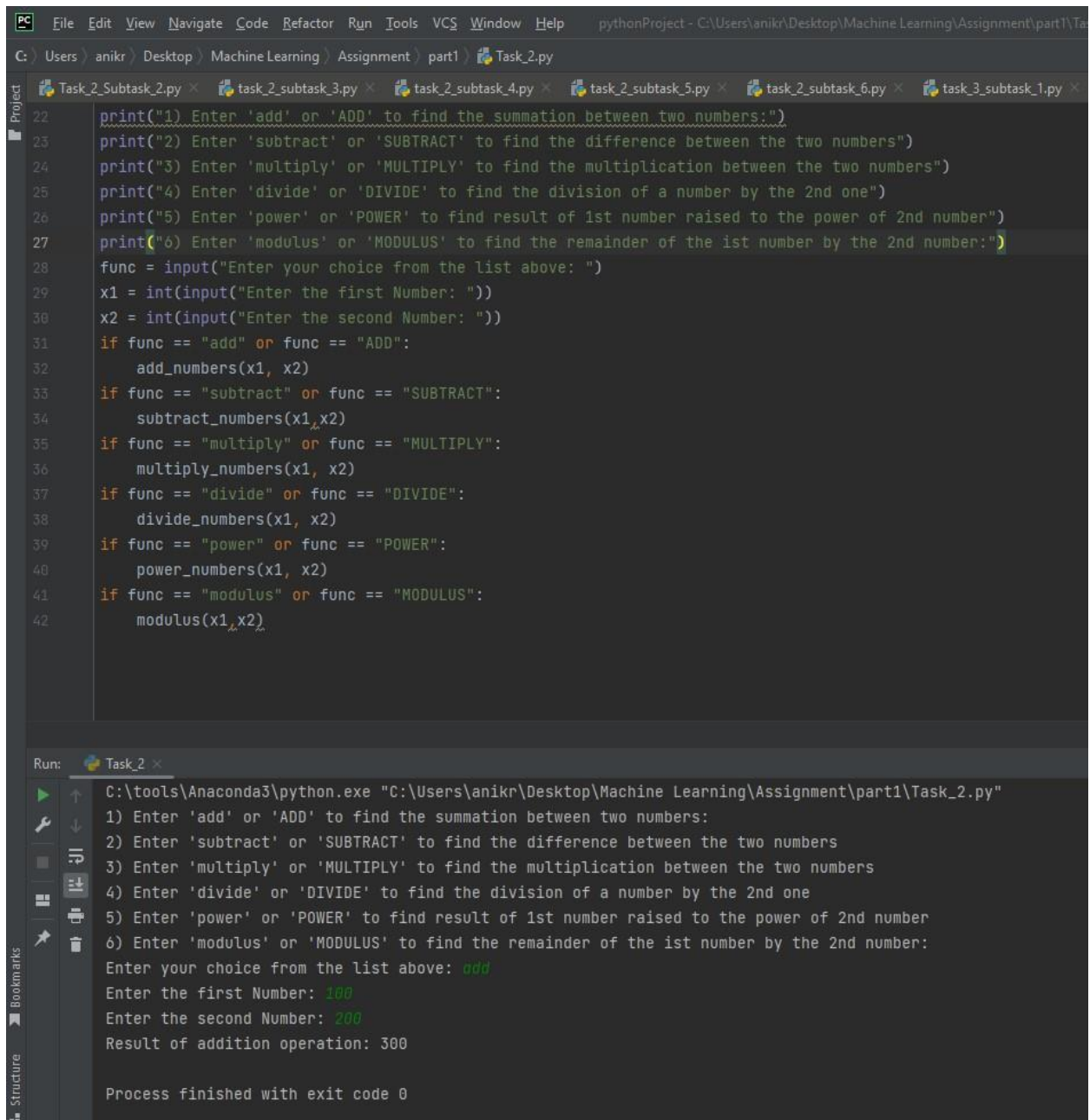
def subtract_numbers(x1,x2):
    print("Result of subtraction operation",x2-x1)

def multiply_numbers(x1, x2):
    print("Result of multiplication operation",x1*x2)

def divide_numbers(x1,x2):
    if x2==0:
        print("ERROR: divide by zero.")
    else:
        print("Result of division operation:",x1/x2)
def power_numbers(x1,x2):
    print("Result of power operation:",pow(x1,2))
def modulus(x1,x2):
    print("Result of modulus operation:",x1 % x2)

print("1) Enter 'add' or 'ADD' to find the summation between two numbers:")
print("2) Enter 'subtract' or 'SUBTRACT' to find the difference between the two numbers")
print("3) Enter 'multiply' or 'MULTIPLY' to find the multiplication between the two numbers")
print("4) Enter 'divide' or 'DIVIDE' to find the division of a number by the 2nd one")
print("5) Enter 'power' or 'POWER' to find result of 1st number raised to the power of 2nd number")
print("6) Enter 'modulus' or 'MODULUS' to find the remainder of the 1st number by the 2nd number:")
func = input("Enter your choice from the list above: ")
x1 = int(input("Enter the first Number: "))
x2 = int(input("Enter the second Number: "))
if func == "add" or func == "ADD":
    add_numbers(x1, x2)
if func == "subtract" or func == "SUBTRACT":
    subtract_numbers(x1,x2)
if func == "multiply" or func == "MULTIPLY":
    multiply_numbers(x1, x2)
if func == "divide" or func == "DIVIDE":
    divide_numbers(x1, x2)
if func == "power" or func == "POWER":
    power_numbers(x1, x2)
if func == "modulus" or func == "MODULUS":
    modulus(x1,x2)
```

Output:



The screenshot displays a Python IDE with a dark theme. The top toolbar includes icons for File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The file explorer on the left shows the project structure: C:\Users\anikr\Desktop\Machine Learning\Assignment\part1\Task_2.py. The editor window contains the following Python code:

```
22 print("1) Enter 'add' or 'ADD' to find the summation between two numbers:")
23 print("2) Enter 'subtract' or 'SUBTRACT' to find the difference between the two numbers")
24 print("3) Enter 'multiply' or 'MULTIPLY' to find the multiplication between the two numbers")
25 print("4) Enter 'divide' or 'DIVIDE' to find the division of a number by the 2nd one")
26 print("5) Enter 'power' or 'POWER' to find result of 1st number raised to the power of 2nd number")
27 print("6) Enter 'modulus' or 'MODULUS' to find the remainder of the 1st number by the 2nd number:")
28 func = input("Enter your choice from the list above: ")
29 x1 = int(input("Enter the first Number: "))
30 x2 = int(input("Enter the second Number: "))
31 if func == "add" or func == "ADD":
32     add_numbers(x1, x2)
33 if func == "subtract" or func == "SUBTRACT":
34     subtract_numbers(x1,x2)
35 if func == "multiply" or func == "MULTIPLY":
36     multiply_numbers(x1, x2)
37 if func == "divide" or func == "DIVIDE":
38     divide_numbers(x1, x2)
39 if func == "power" or func == "POWER":
40     power_numbers(x1, x2)
41 if func == "modulus" or func == "MODULUS":
42     modulus(x1,x2)
```

The bottom panel shows the Run output for Task_2.py:

```
Run: Task_2
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part1\Task_2.py"
1) Enter 'add' or 'ADD' to find the summation between two numbers:
2) Enter 'subtract' or 'SUBTRACT' to find the difference between the two numbers
3) Enter 'multiply' or 'MULTIPLY' to find the multiplication between the two numbers
4) Enter 'divide' or 'DIVIDE' to find the division of a number by the 2nd one
5) Enter 'power' or 'POWER' to find result of 1st number raised to the power of 2nd number
6) Enter 'modulus' or 'MODULUS' to find the remainder of the 1st number by the 2nd number:
Enter your choice from the list above: add
Enter the first Number: 100
Enter the second Number: 200
Result of addition operation: 300

Process finished with exit code 0
```

Assignment 2:

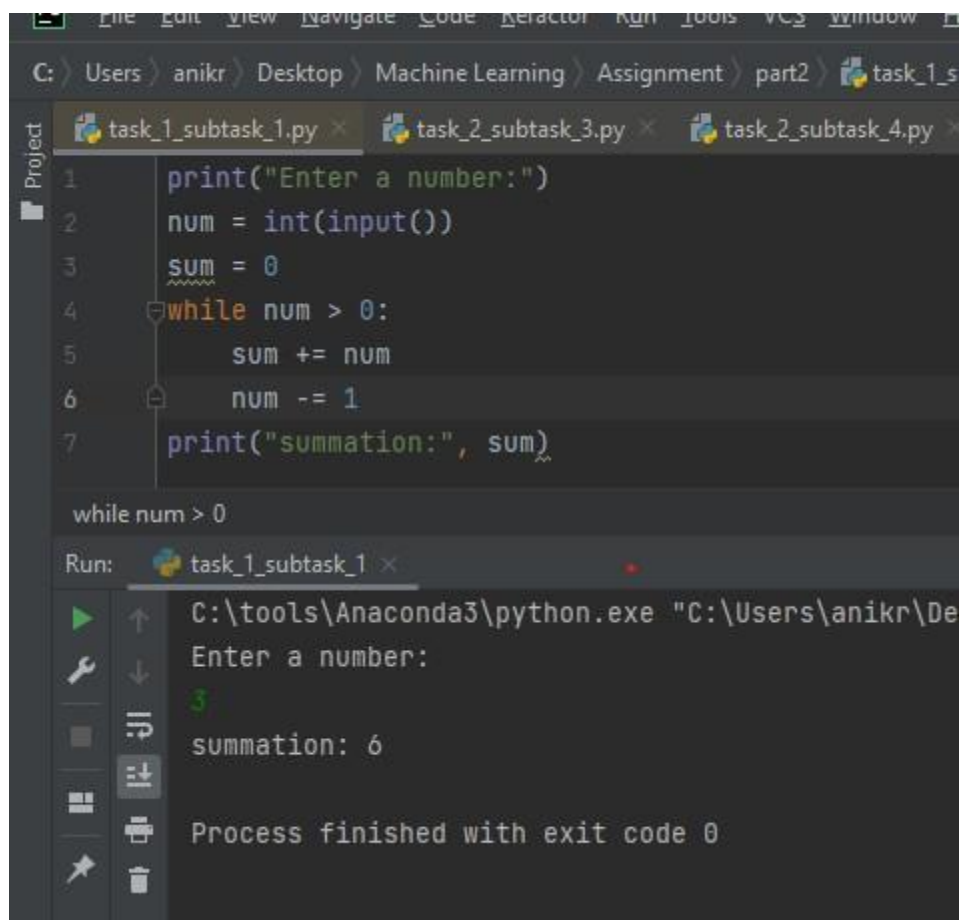
Task 1

Subtask_1:

Code:

```
print("Enter a number:")
num = int(input())
sum = 0
while num > 0:
    sum += num
    num -= 1
print("summation:", sum)
```

Output:



The screenshot shows a code editor with the following code:

```
1 print("Enter a number:")
2 num = int(input())
3 sum = 0
4 while num > 0:
5     sum += num
6     num -= 1
7 print("summation:", sum)
```

The code is executed, and the output is shown in the Run console:

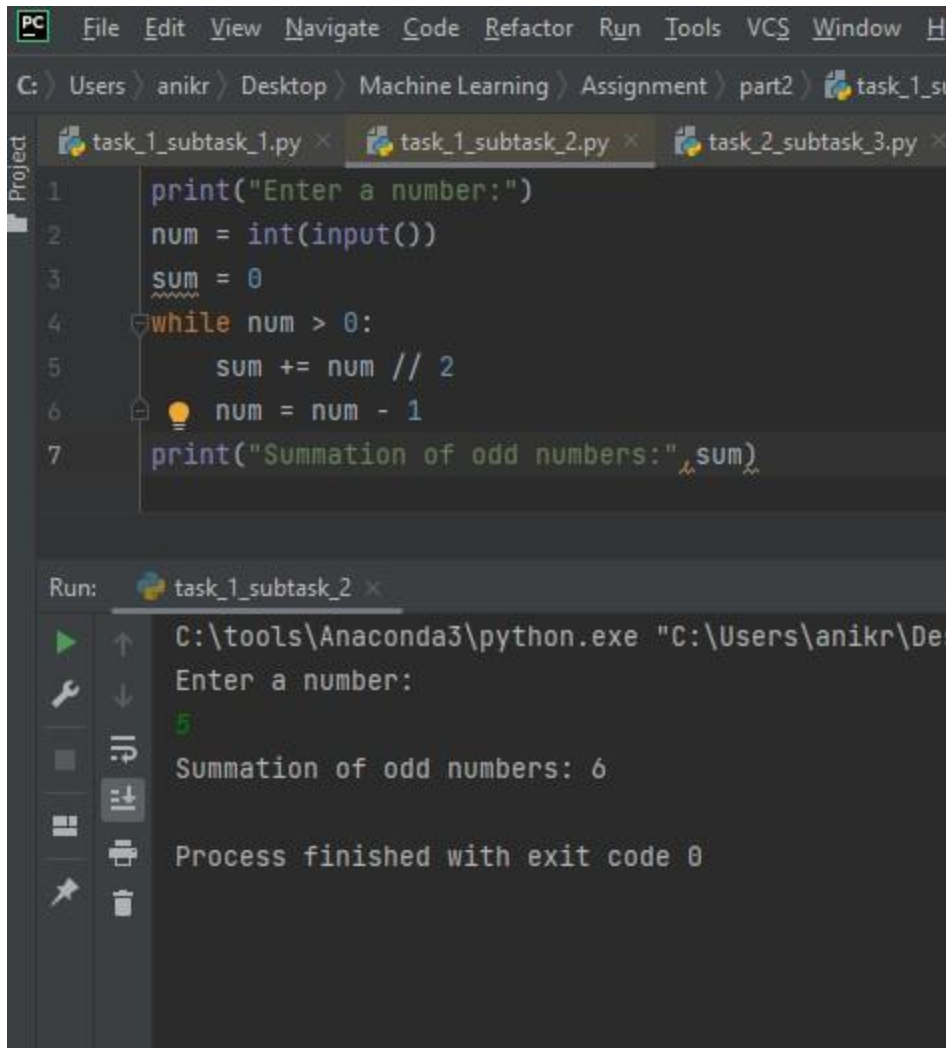
```
Run: task_1_subtask_1 x
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_1_subtask_1.py"
Enter a number:
3
summation: 6
Process finished with exit code 0
```

Subtask_2:

Code:

```
print("Enter a number:")
num = int(input())
sum = 0
while num > 0:
    sum += num // 2
    num = num - 1
print("Summation of odd numbers:", sum)
```

Output:



The screenshot shows an IDE window with the following components:

- File Explorer:** Shows the project path: C:\Users\anikr\Desktop\Machine Learning\Assignment\part2.
- Code Editor:** Displays the Python code for task_1_subtask_2.py. The code is as follows:

```
1 print("Enter a number:")
2 num = int(input())
3 sum = 0
4 while num > 0:
5     sum += num // 2
6     num = num - 1
7 print("Summation of odd numbers:", sum)
```
- Run Console:** Shows the execution of the code. The output is:

```
Run: task_1_subtask_2 x
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_1_subtask_2.py"
Enter a number:
5
Summation of odd numbers: 6
Process finished with exit code 0
```

Task 2:

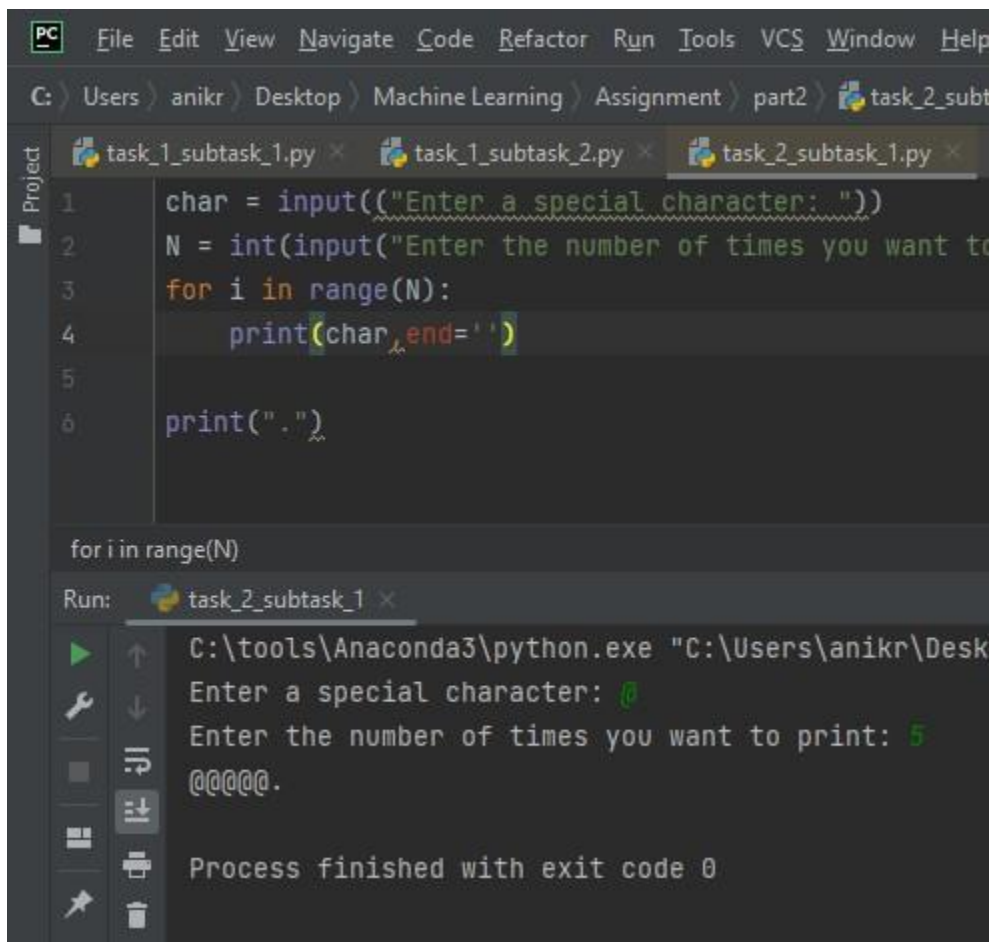
Subtask 1:

Code:

```
char = input("Enter a special character: ")
N = int(input("Enter the number of times you want to print: "))
for i in range(N):
    print(char,end='')

print(".")
```

Output:



The screenshot shows an IDE window with the following components:

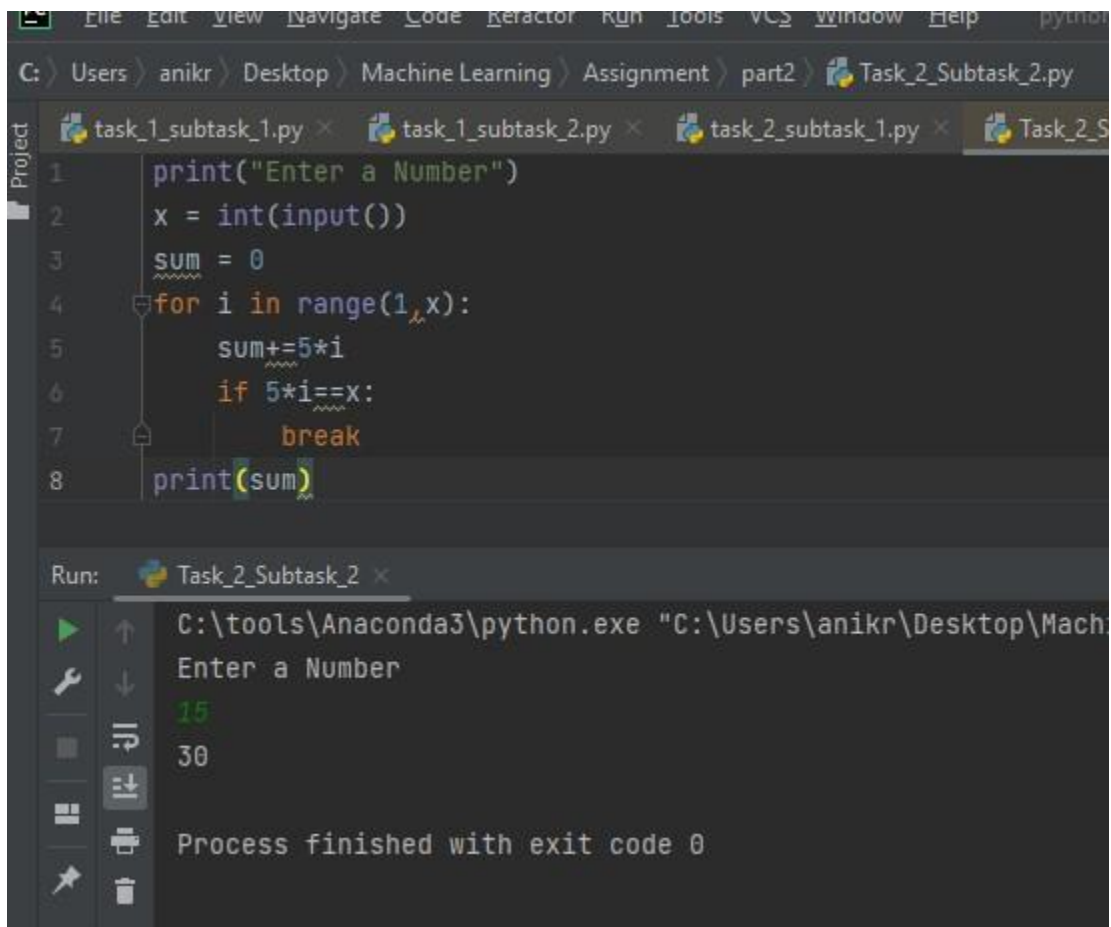
- File Explorer:** Shows the path `C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_1.py`.
- Code Editor:** Contains the Python code from the previous block, with line numbers 1 through 6 on the left margin.
- Run Console:** Shows the execution of the script. It starts with the command `C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_1.py"`. The input `@` is shown for the first prompt, and `5` for the second. The output is `@@@@@.`. The console ends with `Process finished with exit code 0`.

Subtask 2:

Code:

```
print("Enter a Number")
x = int(input())
sum = 0
for i in range(1,x):
    sum+=5*i
    if 5*i==x:
        break
print(sum)
```

Output:



The screenshot shows an IDE window with the file path `C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\Task_2_Subtask_2.py`. The code editor displays the same Python code as in the previous block. Below the editor, the 'Run' panel shows the command `C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Mach` and the output `Enter a Number`, `15`, and `30`. The process finished with exit code 0.

```
1 print("Enter a Number")
2 x = int(input())
3 sum = 0
4 for i in range(1,x):
5     sum+=5*i
6     if 5*i==x:
7         break
8 print(sum)
```

Run: Task_2_Subtask_2

C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Mach

Enter a Number

15

30

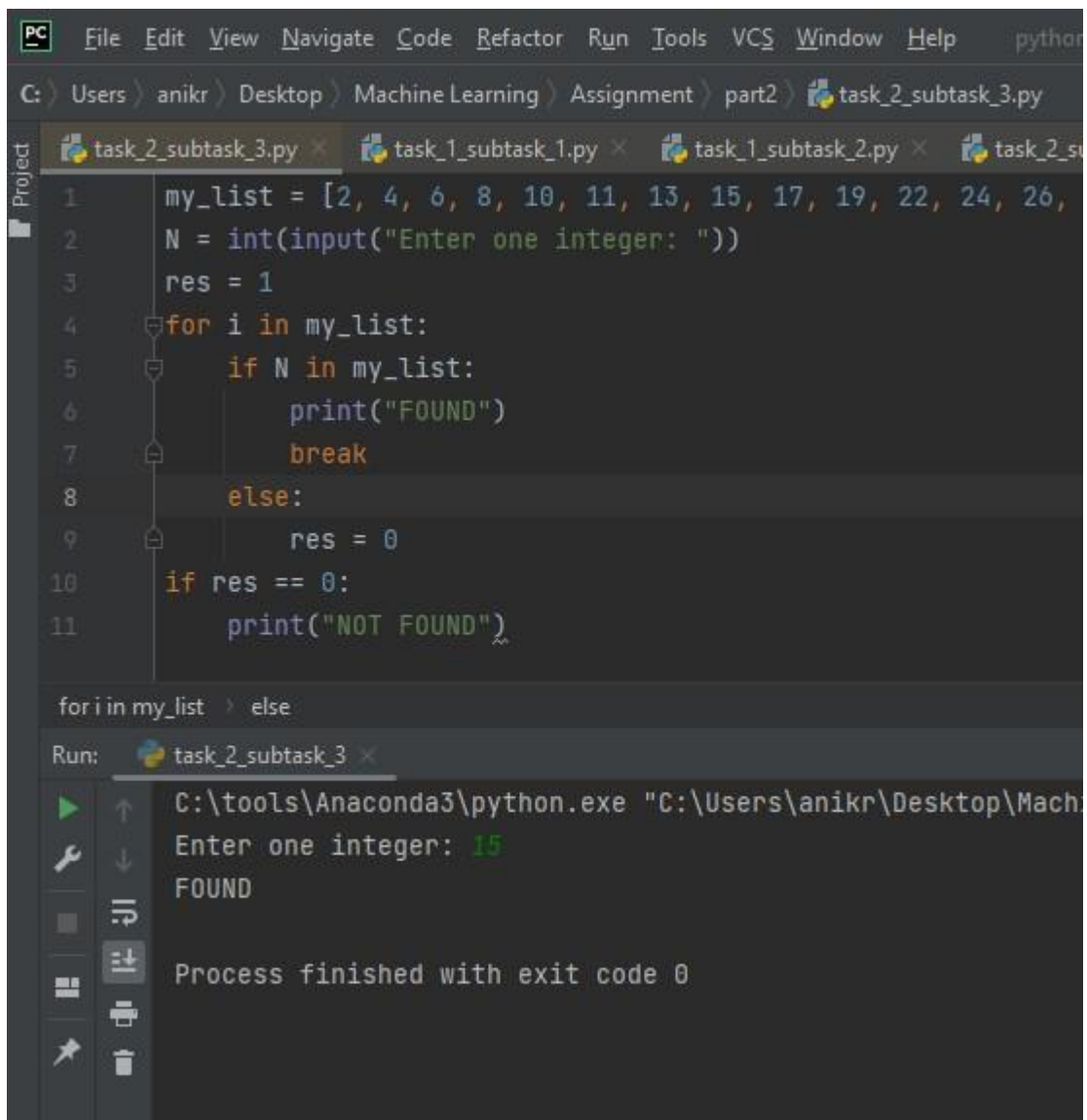
Process finished with exit code 0

Subtask 3:

Code:

```
my_list = [2, 4, 6, 8, 10, 11, 13, 15, 17, 19, 22, 24, 26, 26, 28]
N = int(input("Enter one integer: "))
res = 1
for i in my_list:
    if N in my_list:
        print("FOUND")
        break
    else:
        res = 0
if res == 0:
    print("NOT FOUND")
```

Output:



The screenshot shows an IDE window with the following content:

File Edit View Navigate Code Refactor Run Tools VCS Window Help python

C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_3.py

task_2_subtask_3.py x task_1_subtask_1.py x task_1_subtask_2.py x task_2_subtask_3.py x

```
1 my_list = [2, 4, 6, 8, 10, 11, 13, 15, 17, 19, 22, 24, 26, 26, 28]
2 N = int(input("Enter one integer: "))
3 res = 1
4 for i in my_list:
5     if N in my_list:
6         print("FOUND")
7         break
8     else:
9         res = 0
10 if res == 0:
11     print("NOT FOUND")
```

for i in my_list > else

Run: task_2_subtask_3 x

C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Mach

Enter one integer: 15

FOUND

Process finished with exit code 0

Subtask 4:

Code:

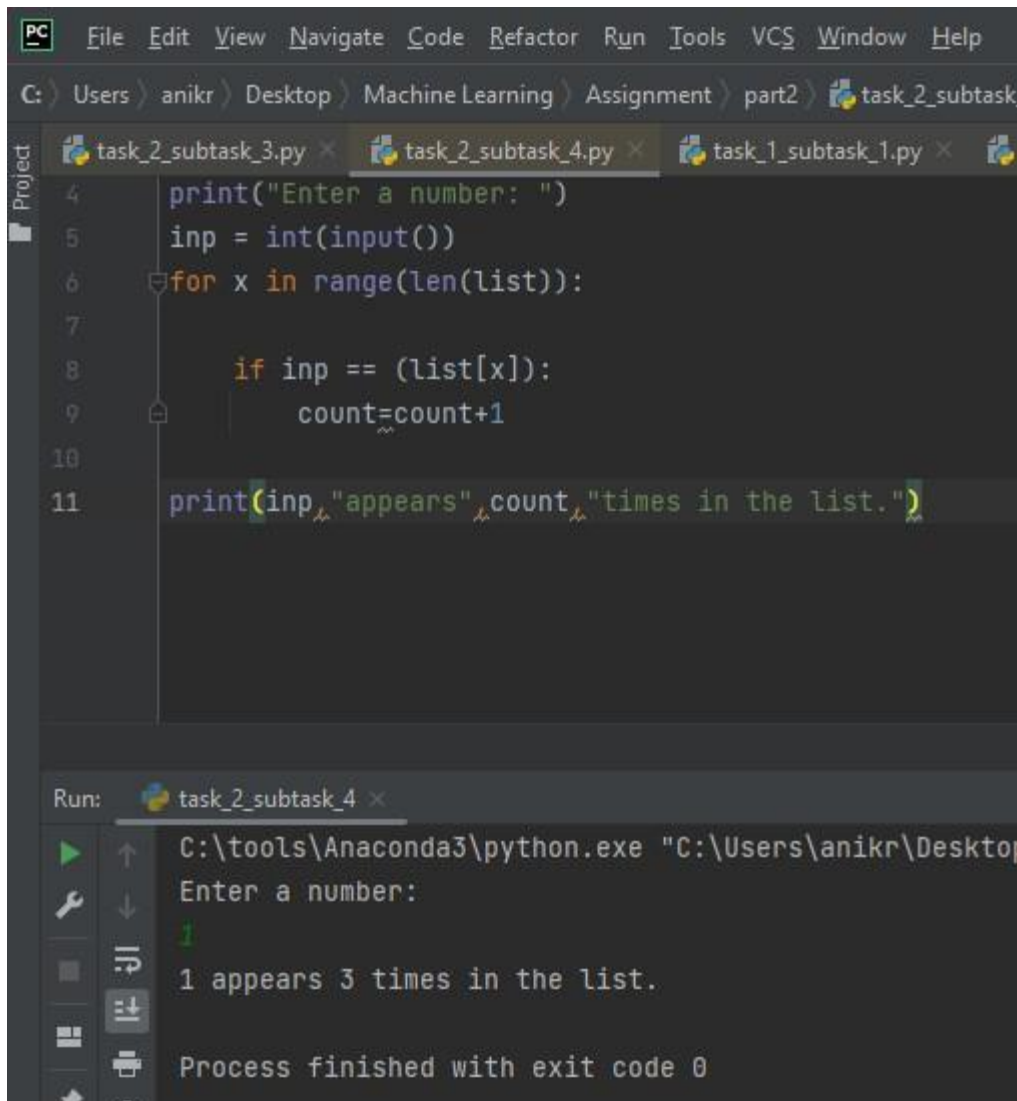
```
list = [1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7, 8, 8, 8, 8, 8, 8, 8, 8, 9, 9, 9, 9, 9, 9, 9, 9]

count = 0
print("Enter a number: ")
inp = int(input())
for x in range(len(list)):

    if inp == (list[x]):
        count=count+1

print(inp,"appears",count,"times in the list.")
```

Output:



The screenshot shows an IDE window with the following components:

- Menu Bar:** PC, File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help.
- Path Bar:** C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask4.py
- Project Explorer:** Shows a project structure with files task_2_subtask_3.py, task_2_subtask_4.py (selected), and task_1_subtask_1.py.
- Code Editor:** Displays the Python code from the previous block. Line 11 is highlighted.
- Run Console:** Shows the execution of task_2_subtask_4.py. The output is:
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\...
Enter a number:
1
1 appears 3 times in the list.
Process finished with exit code 0

Subtask 5:

Code:

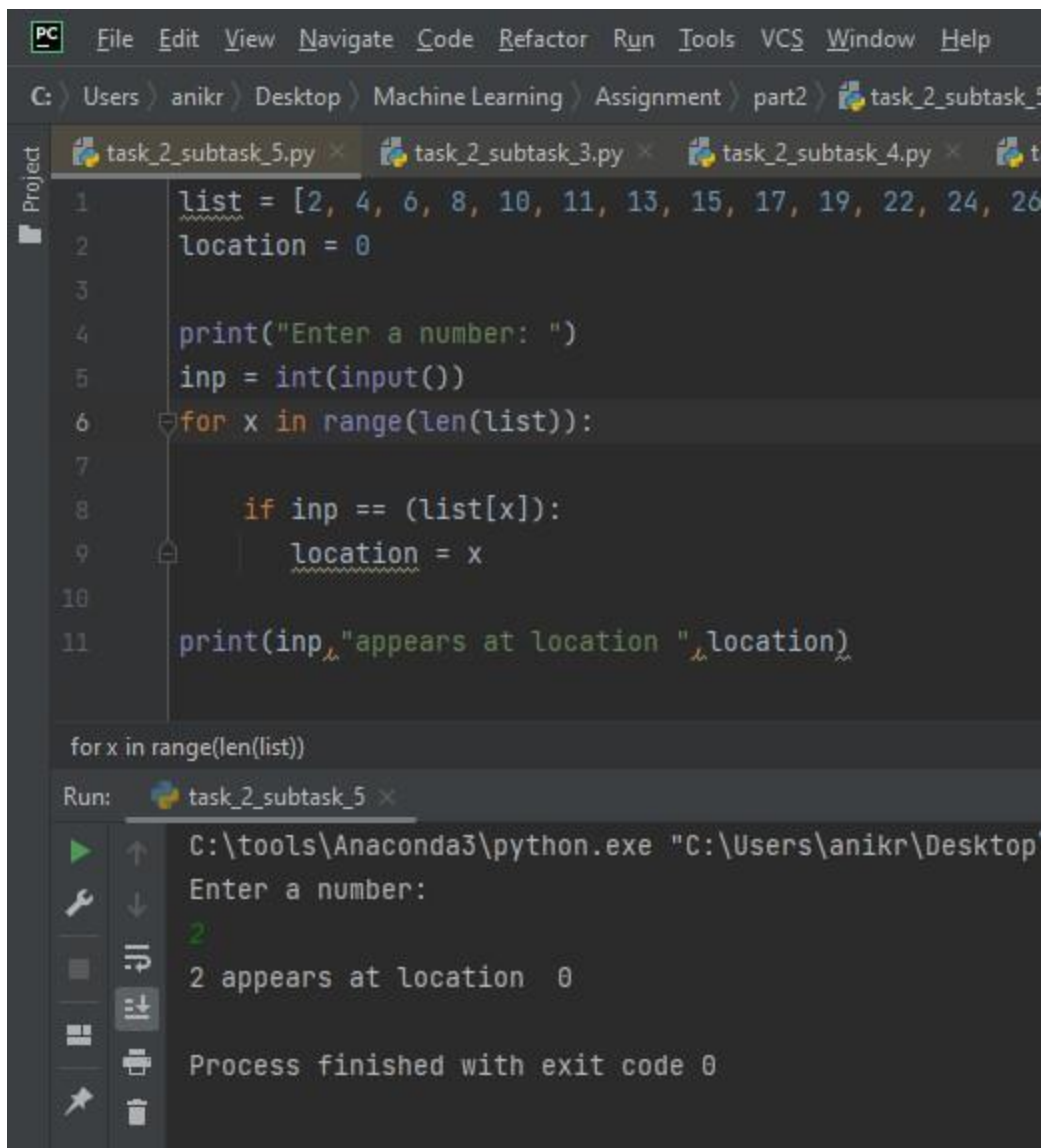
```
list = [2, 4, 6, 8, 10, 11, 13, 15, 17, 19, 22, 24, 26, 28]
location = 0

print("Enter a number: ")
inp = int(input())
for x in range(len(list)):

    if inp == (list[x]):
        location = x

print(inp,"appears at location ",location)
```

Output:



The screenshot shows an IDE window with the file path `C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_5.py`. The code editor displays the same Python code as in the previous block. Below the editor, the Run console shows the execution output: `Enter a number:` followed by the input `2`, and then `2 appears at location 0`. The console also indicates `Process finished with exit code 0`.

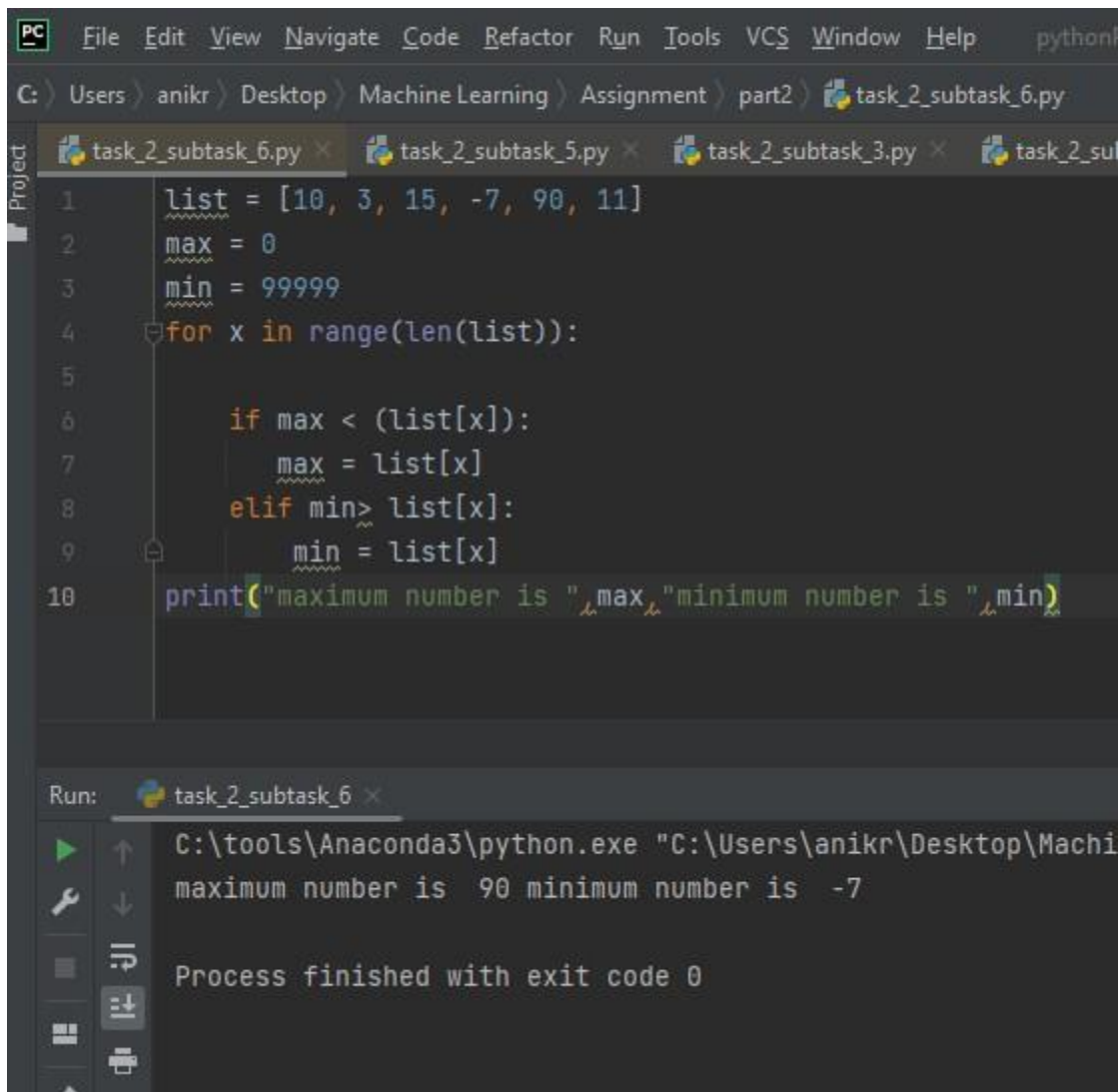
Subtask 6:

Code:

```
list = [10, 3, 15, -7, 90, 11]
max = 0
min = 99999
for x in range(len(list)):

    if max < (list[x]):
        max = list[x]
    elif min > list[x]:
        min = list[x]
print("maximum number is ",max,"minimum number is ",min)
```

Output:



The screenshot shows an IDE window with the file path `C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_6.py`. The code is the same as in the previous block. Below the editor, the Run console shows the command `C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machi` and the output `maximum number is 90 minimum number is -7`. The process finished with exit code 0.

```
PC File Edit View Navigate Code Refactor Run Tools VCS Window Help python
C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_2_subtask_6.py
task_2_subtask_6.py task_2_subtask_5.py task_2_subtask_3.py task_2_su
1 list = [10, 3, 15, -7, 90, 11]
2 max = 0
3 min = 99999
4 for x in range(len(list)):
5
6     if max < (list[x]):
7         max = list[x]
8     elif min > list[x]:
9         min = list[x]
10 print("maximum number is ",max,"minimum number is ",min)
Run: task_2_subtask_6
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machi
maximum number is 90 minimum number is -7
Process finished with exit code 0
```

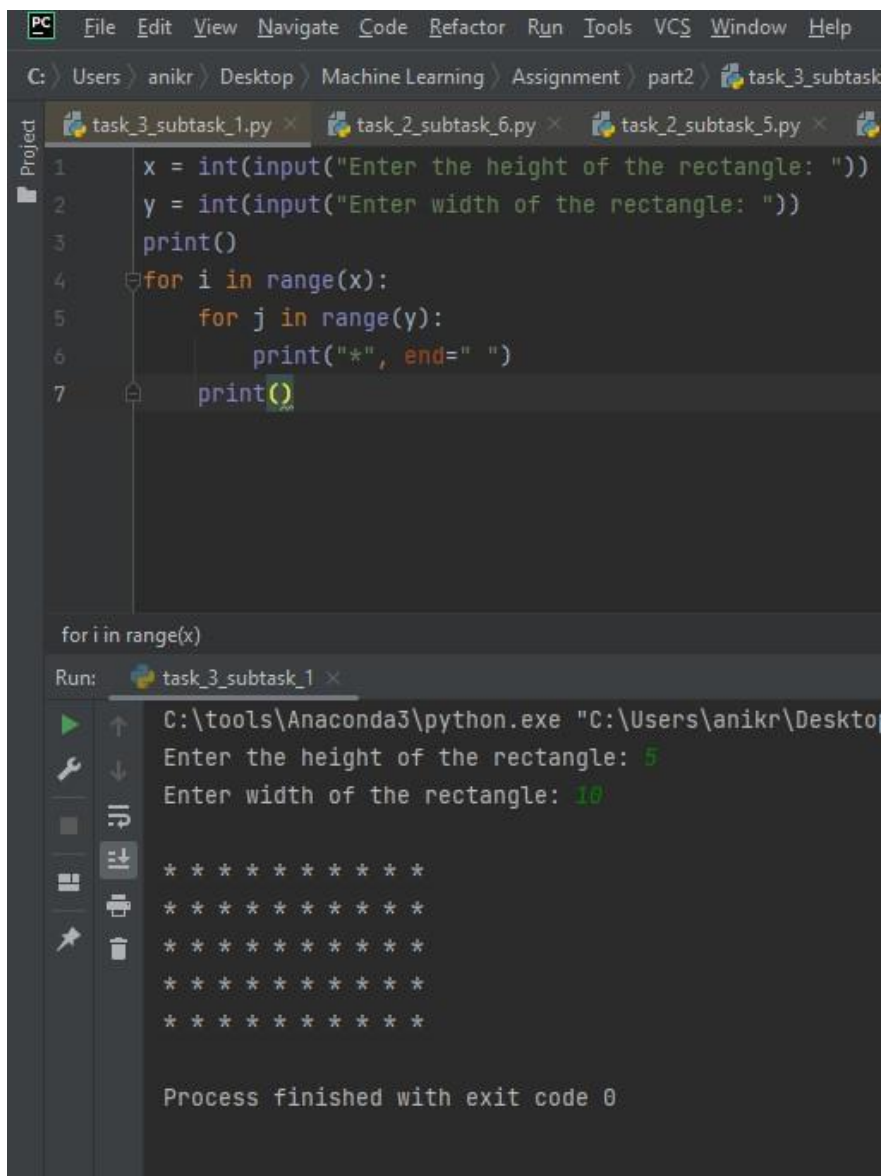
Task 3

Subtask 1:

Code:

```
x = int(input("Enter the height of the rectangle: "))
y = int(input("Enter width of the rectangle: "))
print()
for i in range(x):
    for j in range(y):
        print("*", end=" ")
    print()
```

Output:



The screenshot shows an IDE window with the following components:

- File Explorer:** Shows the project path: C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_3_subtask1.py.
- Code Editor:** Displays the Python code for printing a rectangle of asterisks. The code is as follows:

```
1 x = int(input("Enter the height of the rectangle: "))
2 y = int(input("Enter width of the rectangle: "))
3 print()
4 for i in range(x):
5     for j in range(y):
6         print("*", end=" ")
7     print()
```
- Run Console:** Shows the execution of the code. The user input is 5 for height and 10 for width. The output is a rectangle of 5 rows and 10 columns of asterisks.

```
Run: task_3_subtask_1.py
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_3_subtask1.py"
Enter the height of the rectangle: 5
Enter width of the rectangle: 10

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

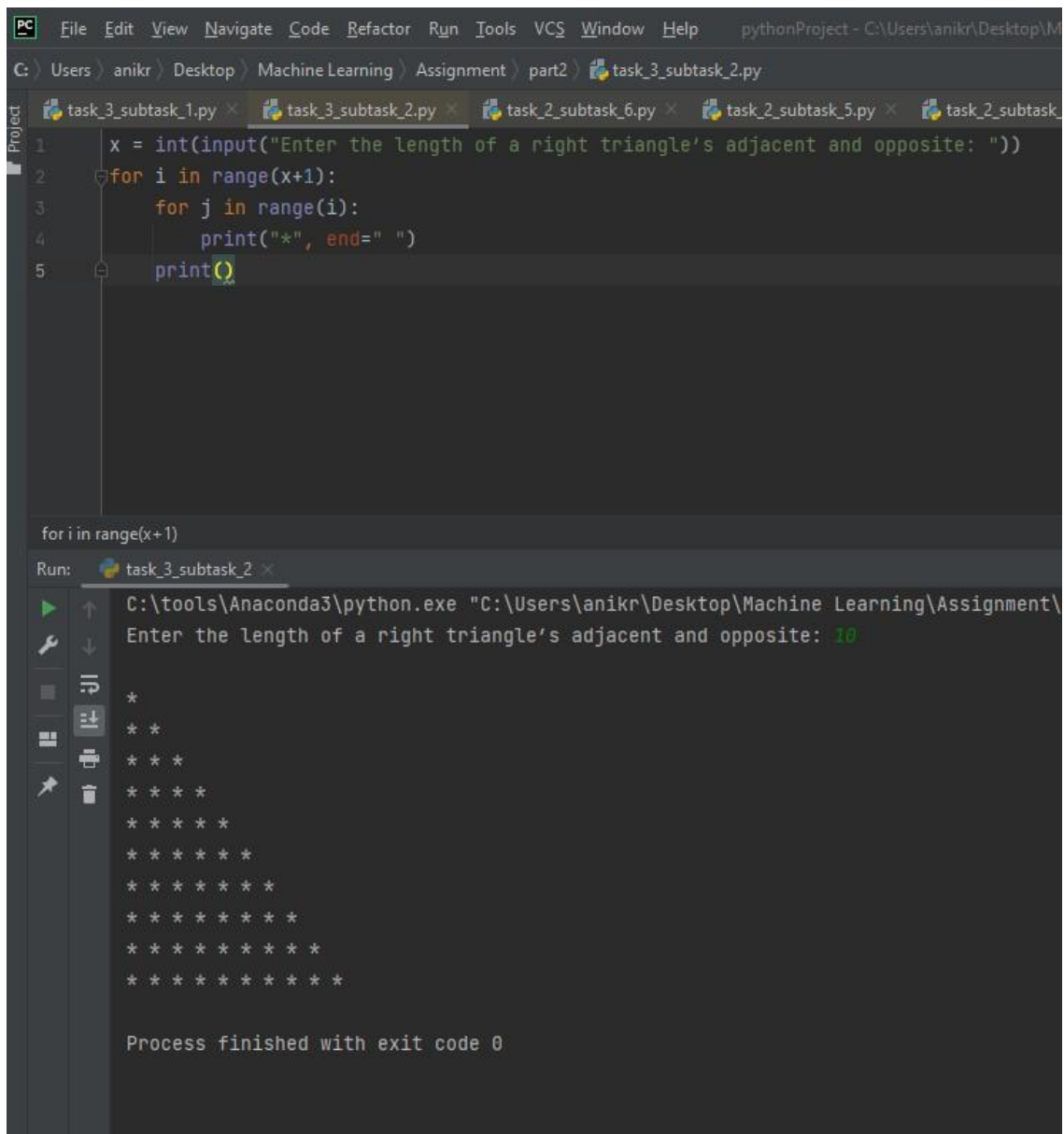
Process finished with exit code 0
```

Subtask 2:

Code:

```
x = int(input("Enter the length of a right triangle's adjacent and opposite: "))
for i in range(x+1):
    for j in range(i):
        print("*", end=" ")
    print()
```

Output:



The screenshot shows an IDE window with the following components:

- Editor:** Contains the Python code for Subtask 2. The code is as follows:

```
x = int(input("Enter the length of a right triangle's adjacent and opposite: "))
for i in range(x+1):
    for j in range(i):
        print("*", end=" ")
    print()
```
- Run Console:** Shows the execution of the program. The input is 10, and the output is a right-angled triangle of asterisks:

```
C:\tools\Anaconda3\python.exe "C:\Users\anikr\Desktop\Machine Learning\Assignment\part2\task_3_subtask_2.py"
Enter the length of a right triangle's adjacent and opposite: 10
*
* *
* * *
* * * *
* * * * *
* * * * * *
* * * * * * *
* * * * * * * *
* * * * * * * * *
* * * * * * * * * *
```
- Process Status:** At the bottom, it says "Process finished with exit code 0".

Subtask 3:

Code:

```
x = int(input("enter the length of a left triangle's adjacent and opposite:"))
for i in range(x+1):
    for j in range(x+1-i):
        print(" ", end=" ")
    for k in range(i):
        print("*", end=" ")
    print()
```

Output:

[illegible]