

Name: Alaa Abdalzaher Fahim

Day 2

Fill an array of 5 elements from the user, Sort it in descending and ascending orders then display the output.

The screenshot shows a Python IDE with a dark theme. The Explorer panel on the left shows a project structure with folders 'tasks', 'Day1', and 'Day2', and files 'Task1.py' through 'Task5.py'. The main editor displays the code for 'Task1.py'. The Terminal panel at the bottom shows the execution of the script, which prompts the user to enter 5 numbers and displays the list and its sorted versions in both ascending and descending order.

```
File Edit Selection View Go Run Terminal Help python-iti
```

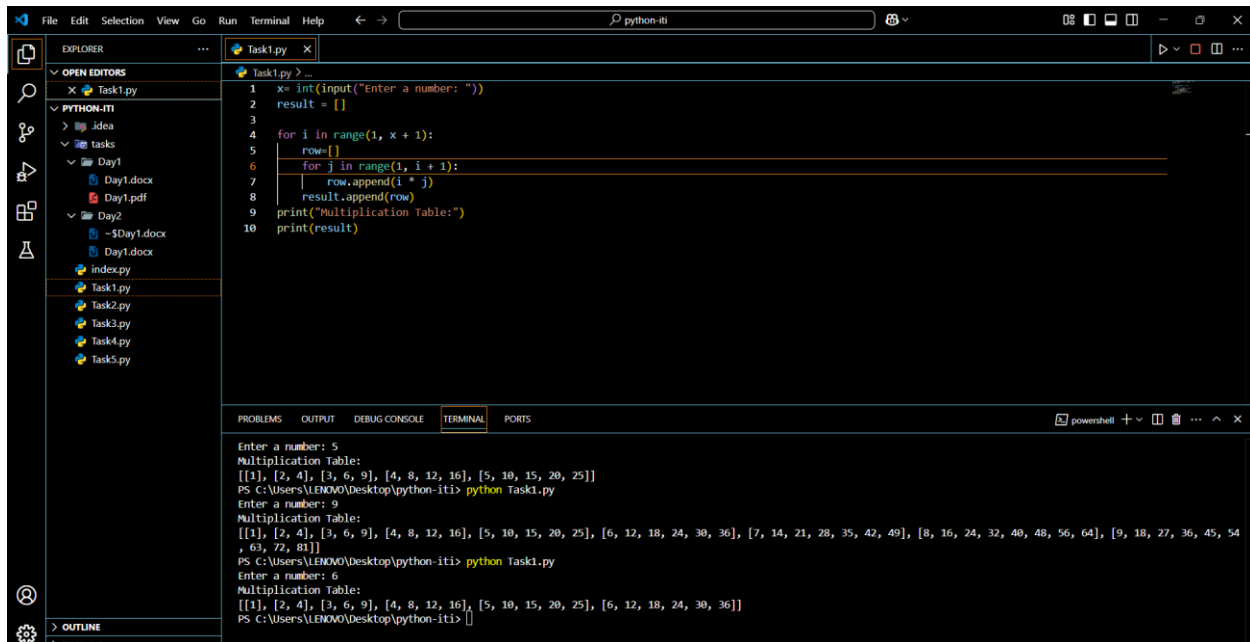
```
Task1.py
```

```
1 L = []
2 for i in range(5):
3     x = int( input(f"Enter a number: ") )
4     L.append(x)
5
6 print("Your list is: ", L)
7 print("Asecending sort of list is: ", sorted(L))
8 print("Desecending sort of list is: ", sorted(L,reverse=True))
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
Enter a number: 5
Enter a number: 6
Enter a number: 9
Enter a number: 4
Your list is: [1, 5, 6, 9, 4]
Asecending sort of list is: [1, 4, 5, 6, 9]
Desecending sort of list is: [9, 6, 5, 4, 1]
PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter a number: 5
Enter a number: 6
Enter a number: 4
Enter a number: 8
Enter a number: 7
Your list is: [5, 6, 4, 8, 7]
Asecending sort of list is: [4, 5, 6, 7, 8]
Desecending sort of list is: [8, 7, 6, 5, 4]
PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter a number: 400
Enter a number: 699
Enter a number: 5
Enter a number: 4698
Enter a number: 4553
Your list is: [400, 699, 5, 4698, 4553]
Asecending sort of list is: [5, 400, 699, 4553, 4698]
Desecending sort of list is: [4698, 4553, 699, 400, 5]
PS C:\Users\LENOVO\Desktop\python-iti>
```

Write a program that generates a multiplication table from 1 to the number passed.



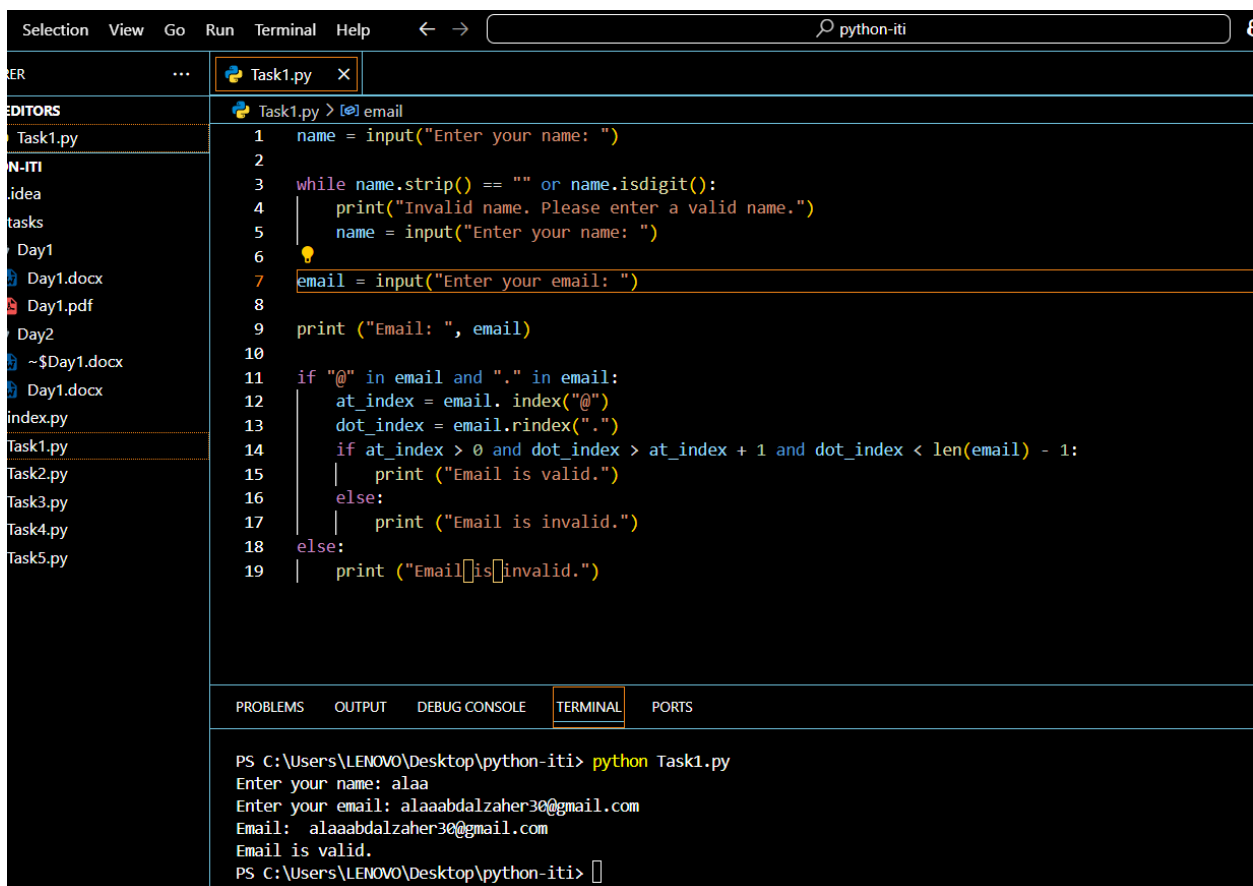
The screenshot shows a Python IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a project named 'python-iti' with several files including 'Task1.py'. The code editor displays the following Python code:

```
1 x= int(input("Enter a number: "))
2 result = []
3
4 for i in range(1, x + 1):
5     row=[]
6     for j in range(1, i + 1):
7         row.append(i * j)
8     result.append(row)
9 print("Multiplication Table:")
10 print(result)
```

The terminal window shows the execution of the program. It prompts the user to enter a number, and for each input, it prints the multiplication table and the corresponding list of lists.

```
Enter a number: 5
Multiplication Table:
[[1], [2, 4], [3, 6, 9], [4, 8, 12, 16], [5, 10, 15, 20, 25]]
PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter a number: 9
Multiplication Table:
[[1], [2, 4], [3, 6, 9], [4, 8, 12, 16], [5, 10, 15, 20, 25], [6, 12, 18, 24, 30, 36], [7, 14, 21, 28, 35, 42, 49], [8, 16, 24, 32, 40, 48, 56, 64], [9, 18, 27, 36, 45, 54, 63, 72, 81]]
PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter a number: 6
Multiplication Table:
[[1], [2, 4], [3, 6, 9], [4, 8, 12, 16], [5, 10, 15, 20, 25], [6, 12, 18, 24, 30, 36]]
PS C:\Users\LENOVO\Desktop\python-iti> []
```

Ask the user for his name then confirm that he has entered his name (not an empty string/integers). then proceed to ask him for his email and print all this data (Bonus) check if it is a valid email or not




The screenshot shows a Python IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The code in Task1.py prompts for a name, validates it (must not be empty or a digit), prompts for an email, and then checks if the email is valid based on the presence of '@' and '.' characters. The terminal shows the execution of the script with the input 'alaa' for the name and 'alaaabdalzaher30@gmail.com' for the email, resulting in the output 'Email is valid.'

```
1 name = input("Enter your name: ")
2
3 while name.strip() == "" or name.isdigit():
4     print("Invalid name. Please enter a valid name.")
5     name = input("Enter your name: ")
6
7 email = input("Enter your email: ")
8
9 print ("Email: ", email)
10
11 if "@" in email and "." in email:
12     at_index = email.index("@")
13     dot_index = email.rindex(".")
14     if at_index > 0 and dot_index > at_index + 1 and dot_index < len(email) - 1:
15         print ("Email is valid.")
16     else:
17         print ("Email is invalid.")
18 else:
19     print ("Email is invalid.")
```

PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter your name: alaa
Enter your email: alaaabdalzaher30@gmail.com
Email: alaaabdalzaher30@gmail.com
Email is valid.
PS C:\Users\LENOVO\Desktop\python-iti>

Mario pyramid using list



The image shows a code editor with a file explorer on the left and a terminal at the bottom. The file explorer lists several files, including `Task1.py`. The code editor displays the following Python code:

```
1 h = int(input("Enter the height of the Mario pyramid: "))
2
3 p = []
4 for i in range(1, h + 1):
5     spaces = h - i
6     hashes = i
7     row = " " * spaces + "*" * hashes
8     p.append("".join(row))
9
10
11 for row in p:
12     print(row)
```

The terminal at the bottom shows the execution of the script. It prompts the user to enter the height of the Mario pyramid. The first execution is for height 4, resulting in a pyramid of 4 rows. The second execution is for height 5, resulting in a pyramid of 5 rows.

```
PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter the height of the Mario pyramid: 4
*
**
***
****

PS C:\Users\LENOVO\Desktop\python-iti> python Task1.py
Enter the height of the Mario pyramid: 5
*
**
***
****
*****

PS C:\Users\LENOVO\Desktop\python-iti>
```