

# Name – Amit Kumar

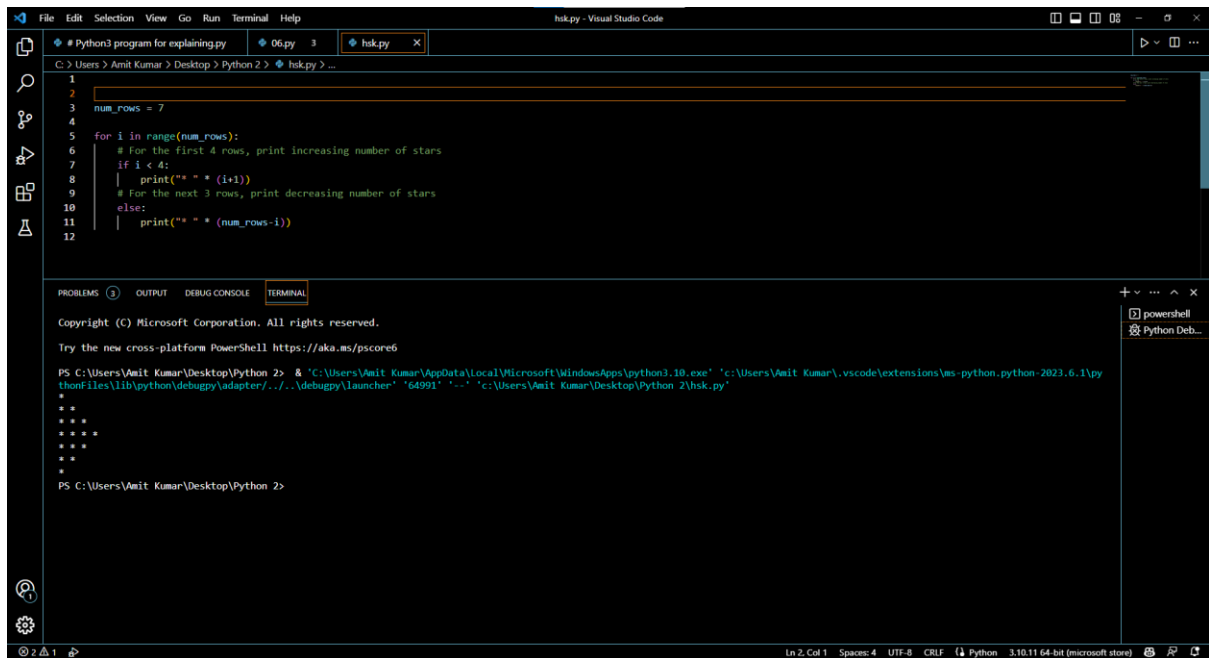
## Batch No – 7

**1. Write a program to print the following pattern :**

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

### Syntax

```
num_rows = 7  
  
for i in range(num_rows):  
    # For the first 4 rows, print increasing number of stars  
    if i < 4:  
        print("* " * (i+1))  
    # For the next 3 rows, print decreasing number of stars  
    else:  
        print("* " * (num_rows-i))
```



The screenshot shows the Visual Studio Code editor with a Python file named `hsk.py`. The code is a program that prints a pattern of stars based on the number of rows. The terminal output shows the execution of the program, which prints a pattern of stars that increases and then decreases.

```
1
2
3 num_rows = 7
4
5 for i in range(num_rows):
6     # For the first 4 rows, print increasing number of stars
7     if i < 4:
8         print("* " * (i+1))
9     # For the next 3 rows, print decreasing number of stars
10    else:
11        print("* " * (num_rows-i))
12
```

Terminal Output:

```
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Amit Kumar\Desktop\Python 2> & 'C:\Users\Amit Kumar\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Amit Kumar\.vscode\extensions\ms-python.python-2023.6.1\pythonfiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '64991' '-.' 'c:\Users\Amit Kumar\Desktop\Python 2\hsk.py'
*
* *
* * *
* * * *
* * * *
* * *
* *
*
PS C:\Users\Amit Kumar\Desktop\Python 2>
```

**2. Write a program to accept 5 even and 5 odd numbers from the user and display :**

- sum of even numbers,
- product of odd numbers
- absolute difference of the sum and product.

## Syntax

# Initialize variables

even\_sum = 0

odd\_product = 1

# Loop to accept 5 even and 5 odd numbers

for i in range(5):

num = int(input("Enter an even number: "))

```

while num % 2 != 0:

    num = int(input("Invalid input! Please enter an even number: "))

even_sum += num

```

```

num = int(input("Enter an odd number: "))

while num % 2 == 0:

    num = int(input("Invalid input! Please enter an odd number: "))

odd_product *= num

```

```

# Calculate absolute difference

```

```

diff = abs(even_sum - odd_product)

```

```

# Print results

```

```

print("Sum of even numbers:", even_sum)

```

```

print("Product of odd numbers:", odd_product)

```

```

print("Absolute difference:", diff)

```

The screenshot shows a Visual Studio Code window with a Python file named 'Hackathon 2.0 Amit.py'. The code in the editor is as follows:

```

16 even_sum = 0
17 odd_product = 1
18
19 # Loop to accept 5 even and 5 odd numbers
20 for i in range(5):
21     num = int(input("Enter an even number: "))
22     while num % 2 != 0:
23         num = int(input("Invalid input! Please enter an even number: "))
24     even_sum += num
25
26     num = int(input("Enter an odd number: "))
27     while num % 2 == 0:
28         num = int(input("Invalid input! Please enter an odd number: "))
29     odd_product *= num
30
31 # Calculate absolute difference
32 diff = abs(even_sum - odd_product)
33
34 # Print results
35 print("Sum of even numbers:", even_sum)
36 print("Product of odd numbers:", odd_product)
37 print("Absolute difference:", diff)
38

```

The terminal output shows the execution of the script, where the user enters 5 even numbers (2, 1, 4, 3, 6) and 5 odd numbers (5, 8, 7, 10, 9). The final output is:

```

Sum of even numbers: 30
Product of odd numbers: 945
Absolute difference: 915

```

**3. Create a class named Item that holds data about an item in a retail store. The class should have the following three properties:**

- name: the name property is a String object that holds the name of the item.
- price: the price property is a double variable that holds the item's retail price
- quantity: the quantity property is an int variable that holds the number of units currently in inventory

Write the following four methods to retrieve the values from the three fields and their current inventory value

- getName( ) returns the item name of type String
- getPrice( ) returns the price of the item of type double

- `getQuantity( )` returns the number of quantities of type `int`
- `getValue( )` that returns the current inventory value (`quantity * price`) of type `double`

## Syntax.

```
class Item:
    def __init__(self, name: str, price: float, quantity: int):
        self.name = name
        self.price = price
        self.quantity = quantity

# create a new item
item = Item("apple", 0.99, 10)

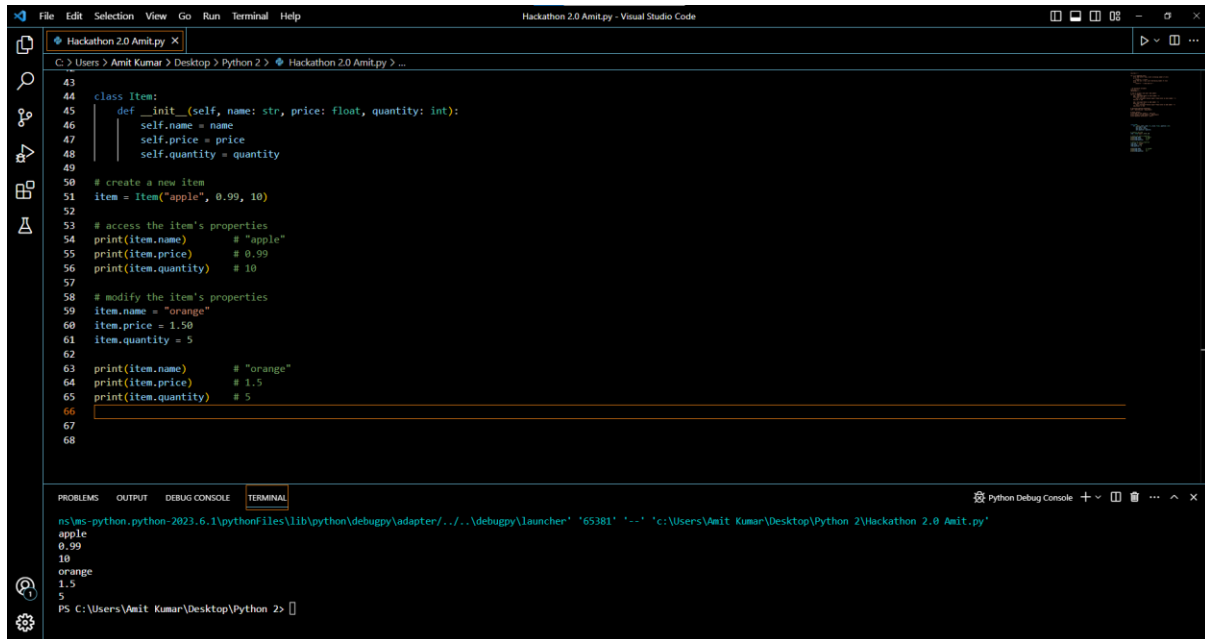
# access the item's properties
print(item.name)    # "apple"
print(item.price)   # 0.99
print(item.quantity) # 10

# modify the item's properties
item.name = "orange"
item.price = 1.50
item.quantity = 5
```

```
print(item.name)    # "orange"
```

```
print(item.price)   # 1.5
```

```
print(item.quantity) # 5
```



The screenshot shows a Visual Studio Code editor window with a Python file named 'Hackathon 2.0 Amit.py'. The code defines a class 'Item' with attributes 'name', 'price', and 'quantity'. It creates an instance 'item' with 'apple', '0.99', and '10'. Then it modifies the instance's attributes to 'orange', '1.5', and '5'. The terminal output shows the execution of the script, displaying the initial and modified values of the 'item' object.

```
43
44 class Item:
45     def __init__(self, name: str, price: float, quantity: int):
46         self.name = name
47         self.price = price
48         self.quantity = quantity
49
50 # create a new item
51 item = Item("apple", 0.99, 10)
52
53 # access the item's properties
54 print(item.name)    # "apple"
55 print(item.price)   # 0.99
56 print(item.quantity) # 10
57
58 # modify the item's properties
59 item.name = "orange"
60 item.price = 1.5
61 item.quantity = 5
62
63 print(item.name)    # "orange"
64 print(item.price)   # 1.5
65 print(item.quantity) # 5
66
67
68
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
msys-python.python-2023.6.1\pythonfiles\lib\python\debugpy\adapter\...\debugpy\launcher '65381' '-' 'c:\Users\Amit Kumar\Desktop\Python 2\Hackathon 2.0 Amit.py'
apple
0.99
10
orange
1.5
5
PS C:\Users\Amit Kumar\Desktop\Python 2>
```

## Major Question 2

1. Ask the user number of rows to be generated of a series. Suppose user enters no. of rows = 5 then the series shall be :

9

99

999

9999

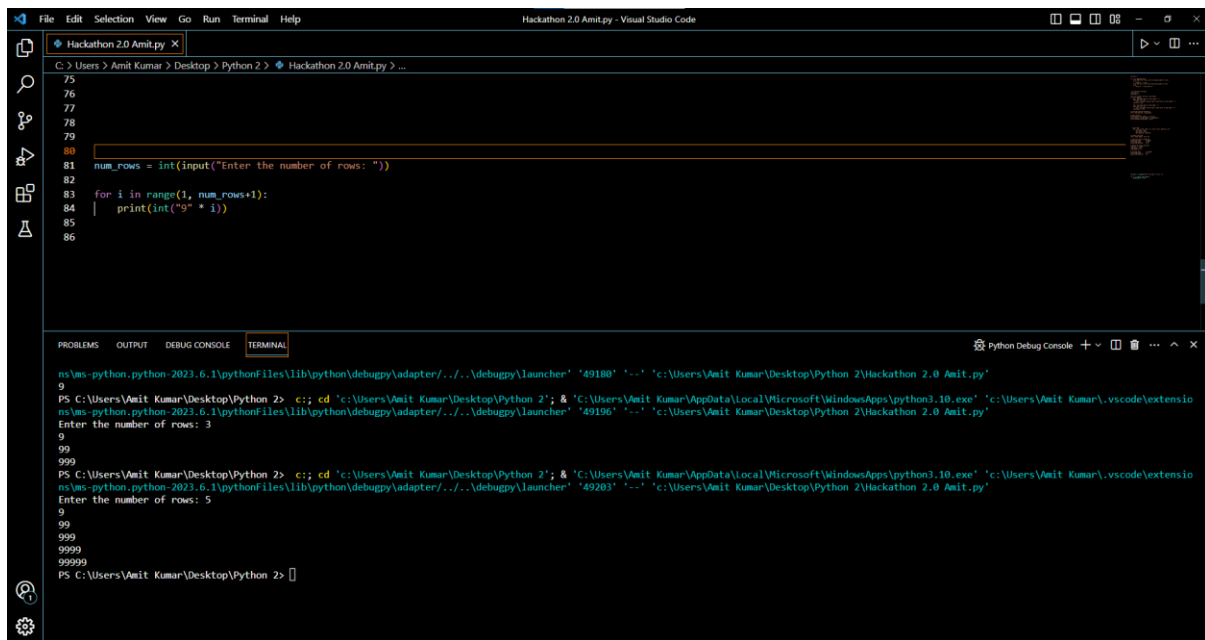
99999

## Syntax

```
num_rows = int(input("Enter the number of rows: "))
```

```
for i in range(1, num_rows+1):
```

```
    print(int("9" * i))
```



The screenshot shows a Visual Studio Code window titled 'Hackathon 2.0 Amit.py - Visual Studio Code'. The editor displays a Python script with the following code:

```
75  
76  
77  
78  
79  
80  
81 num_rows = int(input("Enter the number of rows: "))  
82  
83 for i in range(1, num_rows+1):  
84     print(int("9" * i))  
85  
86
```

The terminal output shows the program being executed twice. In the first run, the user enters '3', and the output is:

```
9  
99  
999
```

In the second run, the user enters '5', and the output is:

```
9  
99  
999  
9999  
99999
```

**2. Write a program to accept a number from the user and check whether the number entered is prime or not**

# Syntax

# get input from user

```
num = int(input("Enter a number: "))
```

# check if number is prime

if num > 1:

```
    for i in range(2, int(num/2)+1):
```

```
        if (num % i) == 0:
```

```
            print(num, "is not a prime number")
```

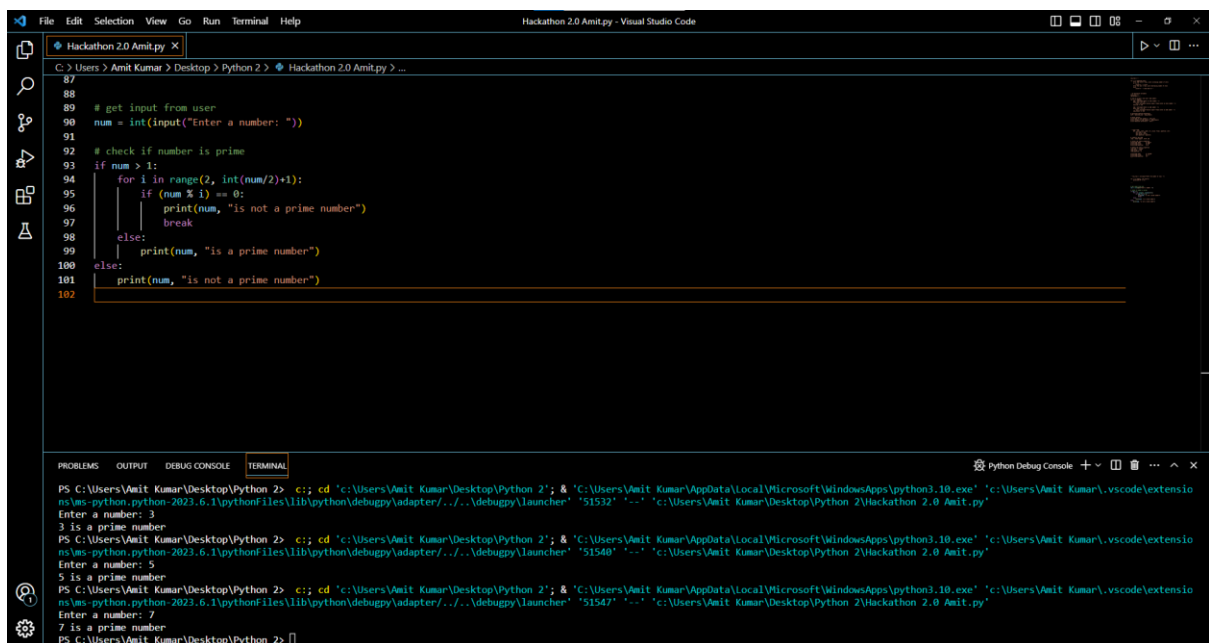
```
            break
```

else:

```
    print(num, "is a prime number")
```

else:

```
    print(num, "is not a prime number")
```



```
87
88
89 # get input from user
90 num = int(input("Enter a number: "))
91
92 # check if number is prime
93 if num > 1:
94     for i in range(2, int(num/2)+1):
95         if (num % i) == 0:
96             print(num, "is not a prime number")
97             break
98         else:
99             print(num, "is a prime number")
100 else:
101     print(num, "is not a prime number")
102
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Python Debug Console

```
PS C:\Users\Amit Kumar\Desktop\Python 2> cd 'c:\Users\Amit Kumar\Desktop\Python 2'; & 'c:\Users\Amit Kumar\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Amit Kumar\.vscode\extensions\ms-python.python-2023.6.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '51532' '-...' 'c:\Users\Amit Kumar\Desktop\Python 2\Hackathon 2.0 Amit.py'
Enter a number: 3
3 is a prime number
PS C:\Users\Amit Kumar\Desktop\Python 2> cd 'c:\Users\Amit Kumar\Desktop\Python 2'; & 'c:\Users\Amit Kumar\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Amit Kumar\.vscode\extensions\ms-python.python-2023.6.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '51540' '-...' 'c:\Users\Amit Kumar\Desktop\Python 2\Hackathon 2.0 Amit.py'
Enter a number: 5
5 is a prime number
PS C:\Users\Amit Kumar\Desktop\Python 2> cd 'c:\Users\Amit Kumar\Desktop\Python 2'; & 'c:\Users\Amit Kumar\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Amit Kumar\.vscode\extensions\ms-python.python-2023.6.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '51547' '-...' 'c:\Users\Amit Kumar\Desktop\Python 2\Hackathon 2.0 Amit.py'
Enter a number: 7
7 is a prime number
PS C:\Users\Amit Kumar\Desktop\Python 2>
```