Assignment 6.2

Name:K.Priya

Roll Number: 2303A51194

Batch – 04

Task 1: (Classes – Data Validation)

• Prompt AI to generate a Student class with attributes: name, roll_no, and marks. Add a method is_pass() that returns whether the student has passed (marks ≥ 40).

```python
class Student:
    def __init__(self, name, roll_no, marks):
        self.name = name
        self.roll_no = roll_no
        self.marks = marks

    def is_pass(self):
        return self.marks >= 40

s1 = Student("Priya", 1194, 65)
print(s1.is_pass())
```

Expected Output-1

• Python class with constructor and validation logic

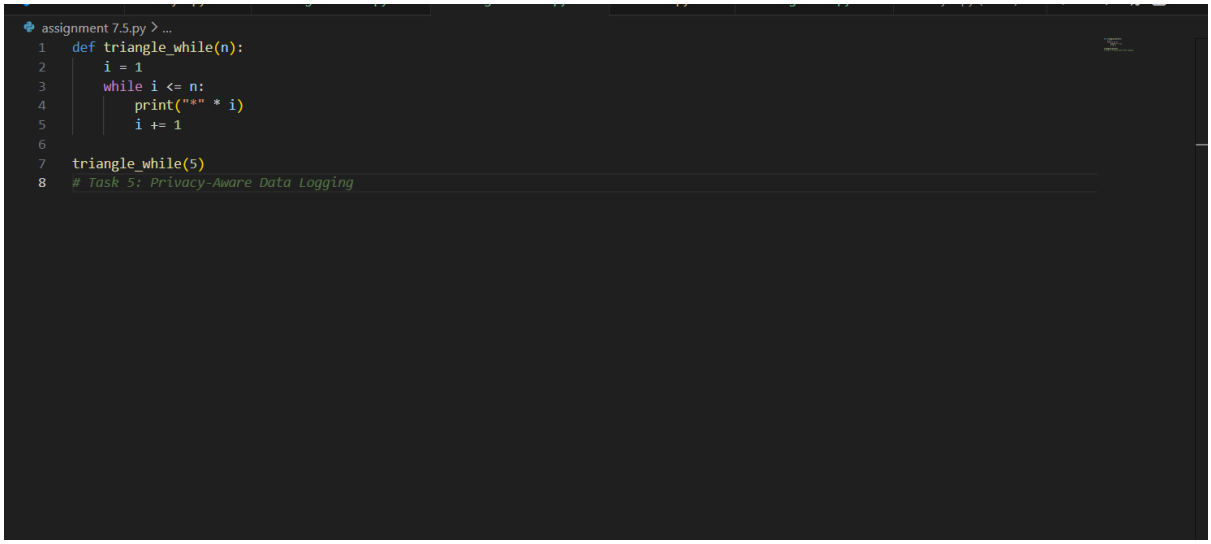
• Clear method implementation returning pass/fail status

• Students analyze correctness and clarity of class design

```
PS C:\Users\hp\OneDrive\Desktop\ai>  & 'c:\Users\hp\AppData\Local\Microsoft\WindowsApps\python3.13.exe' 'c:\Users\hp\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58761' '--' 'C:\Users\hp\OneDrive\Desktop\ai\assignment 7.5.py'
[1]
[2]
PS C:\Users\hp\OneDrive\Desktop\ai>  c:; cd 'c:\Users\hp\OneDrive\Desktop\ai'; & 'c:\Users\hp\AppData\Local\Microsoft\WindowsApps\python3.13.exe' 'c:\Users\hp\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '63783' '--' 'C:\Users\hp\OneDrive\Desktop\ai\assignment 7.5.py'
True
PS C:\Users\hp\OneDrive\Desktop\ai>
```

Task 2: (Loops – Pattern Generation)

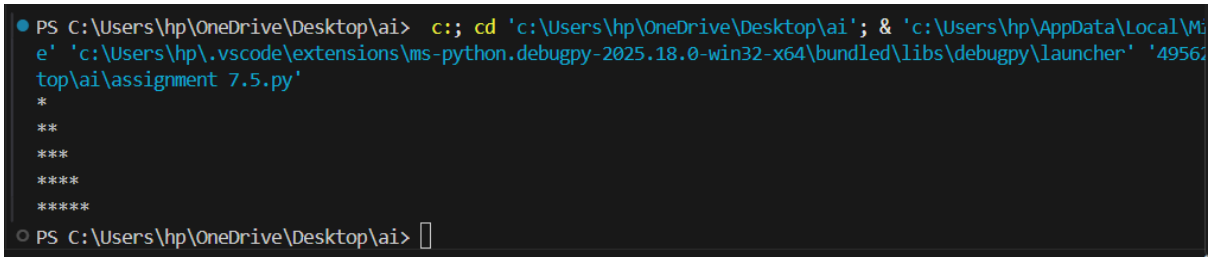
Ask AI to generate a function that prints a right-angled triangle star pattern using a for loop. Then regenerate the same pattern using a while loop.



Output-2

• Correct pattern output using both loop types

• Logical loop structure with proper conditions



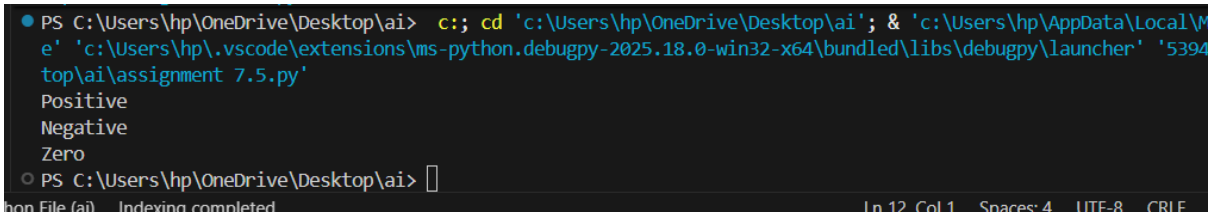
Task 3: (Conditional Statements – Number Analysis)

Ask AI to write a function that checks whether a given number is positive, negative, or zero using if-elif-else. Test the function with multiple inputs.

```python
assignment 7.5.py > ...
def check_number(num):
    if num > 0:
        return "Positive"
    elif num < 0:
        return "Negative"
    else:
        return "Zero"

print(check_number(10))
print(check_number(-5))
print(check_number(0))
```

Output-3

- Function correctly classifies numbers
- Proper handling of all conditions
- Students analyze decision logic



Task 4: (Nested Conditionals)

Generate a function check_discount(age, is_member) that

determines discount eligibility:

- Age ≥ 60 → Senior discount
- Member → Additional discount

Use nested if statements.

```python
def check_discount(age, is_member):
    if age >= 60:
        if is_member:
            return "Senior + Member Discount"
        else:
            return "Senior Discount"
    else:
        if is_member:
            return "Member Discount"
        else:
            return "No Discount"

print(check_discount(65, True))
print(check_discount(30, False))
# Task 5: Privacy-Aware Data Logging
```

Output-4

• Python code using nested conditionals

• Clear explanation of decision flow

```
PS C:\Users\hp\OneDrive\Desktop\ai>  c:; cd 'c:\Users\hp\OneDrive\Desktop\ai'; & 'c:\Users\hp\AppData\Local\Microsoft\WindowsApps\python3.13.exe' 'c:\Users\hp\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '56979' '--' 'C:\Users\hp\OneDrive\Desktop\ai\assignment 7.5.py'
Senior + Member Discount
No Discount
PS C:\Users\hp\OneDrive\Desktop\ai>
```

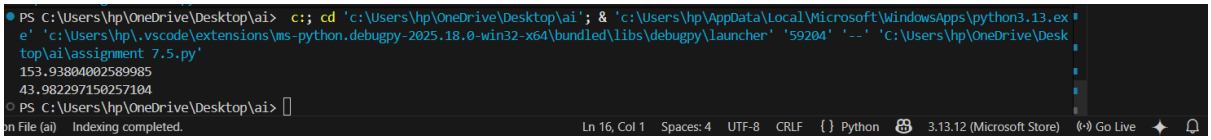
Task 5: (Class – Mathematical Opera)

• Ask AI to create a Circle class with methods to calculate area ()

and circumference () given the radius.

```python
import math

class Circle:
    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return math.pi * self.radius * self.radius

    def circumference(self):
        return 2 * math.pi * self.radius

c = Circle(7)
print(c.area())
print(c.circumference())
# Task 5: Privacy-Aware Data Logging
```

Output-5

• Correct mathematical computation

- Well-structured class with methods
- Code explanation provided