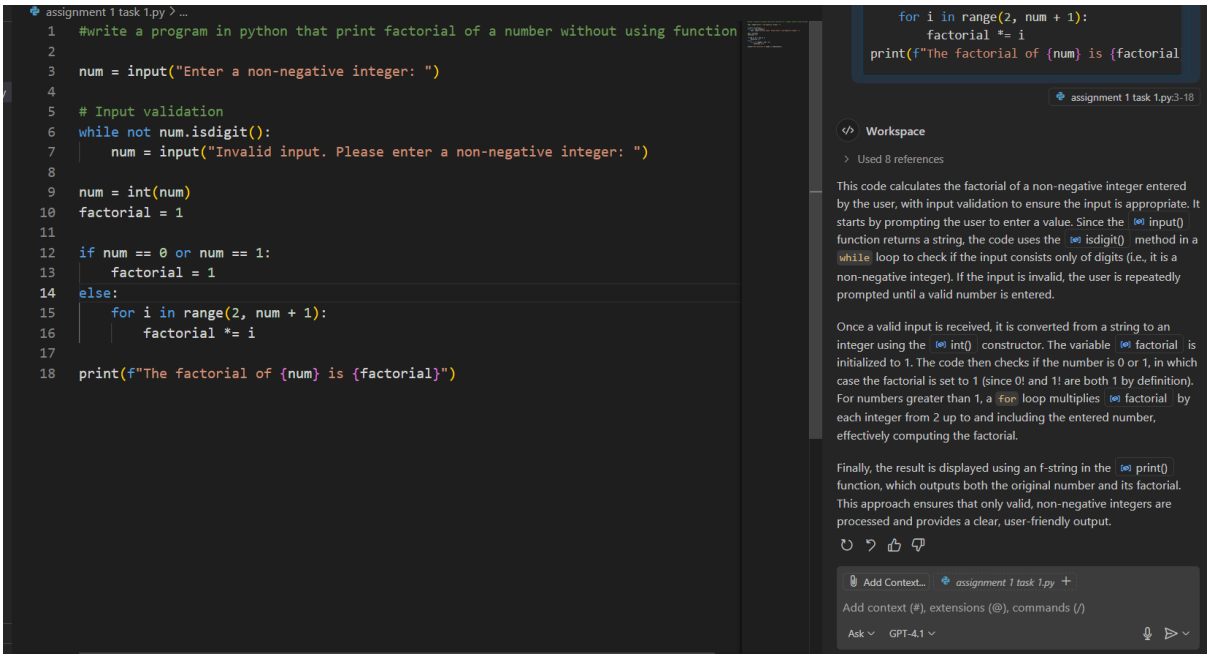
LAB ASSIGNMENT- 1.1

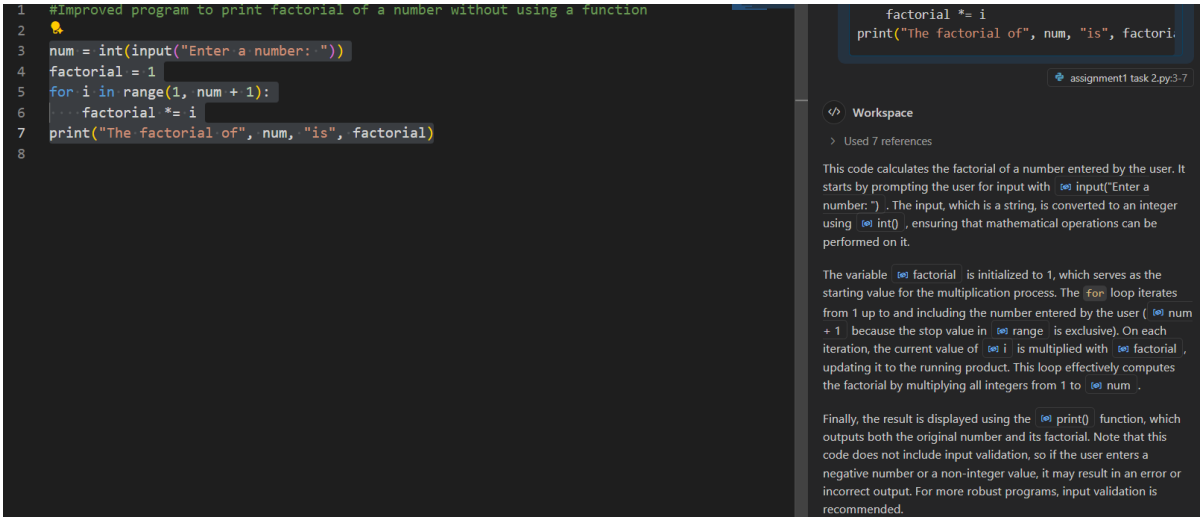
TASK 1:

Factorial without functions



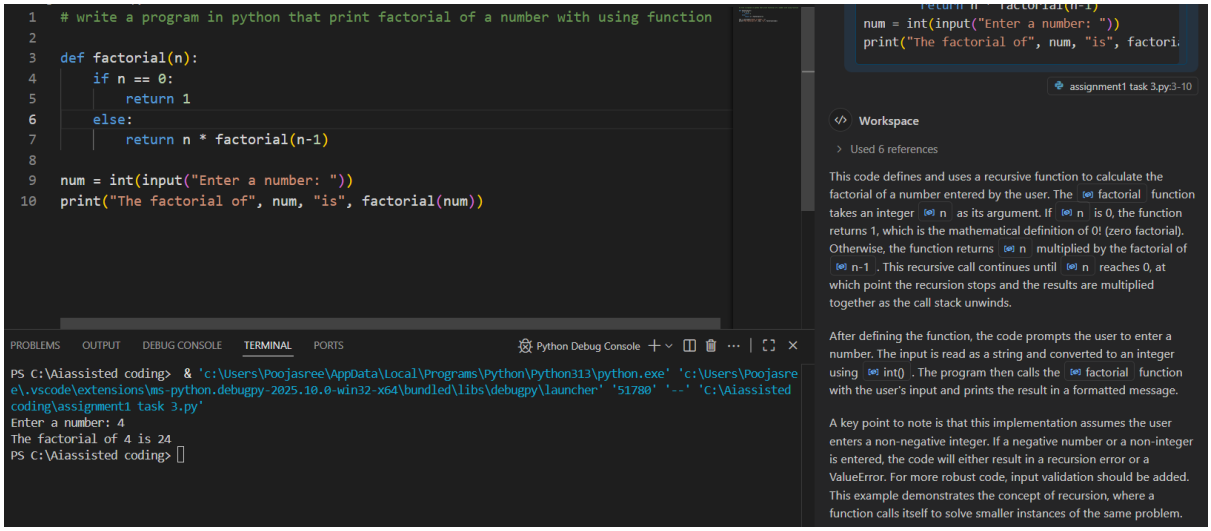
TASK 2:

Improving Efficiency



TASK3:

FACTORIAL WITH FUNCTIONS



TASK 4

COMPARING FACTORIAL WITH FUNCTION AND WITH OUT FUNCTION

The difference between calculating factorial **with functions** and **without functions** is:

With Functions

- The logic for factorial is placed inside a function (e.g., factorial(n)).
- You can reuse the function for different inputs.
- The code is modular, easier to read, and maintain.

Without Functions

- The logic is written directly in the main code, not inside a function.
- You cannot reuse the code easily for different inputs.
- The code is less organized.

TASK 5

Iterative vs Recursive Factorial

assignment 1 task 5.py > factorial_recursive

```python
def factorial_iterative(n):
    result = 1
    for i in range(2, n + 1):
        result *= i
    print(result)

# 2
def factorial_recursive(n):
    if n == 0:
        result = 1
    else:
        result = n * factorial_recursive(n - 1)
    print(result)
```

Iterative Factorial

- Uses a loop (like for or while) to calculate the factorial.
- Updates a result variable step by step.
- Generally uses less memory (no function call stack).

Recursive Factorial

- The function calls itself to solve smaller subproblems.
- Has a base case (n == 0).
- Uses more memory due to function call stack.