

Name: Narige Rohith

Enroll no : 2403A54101

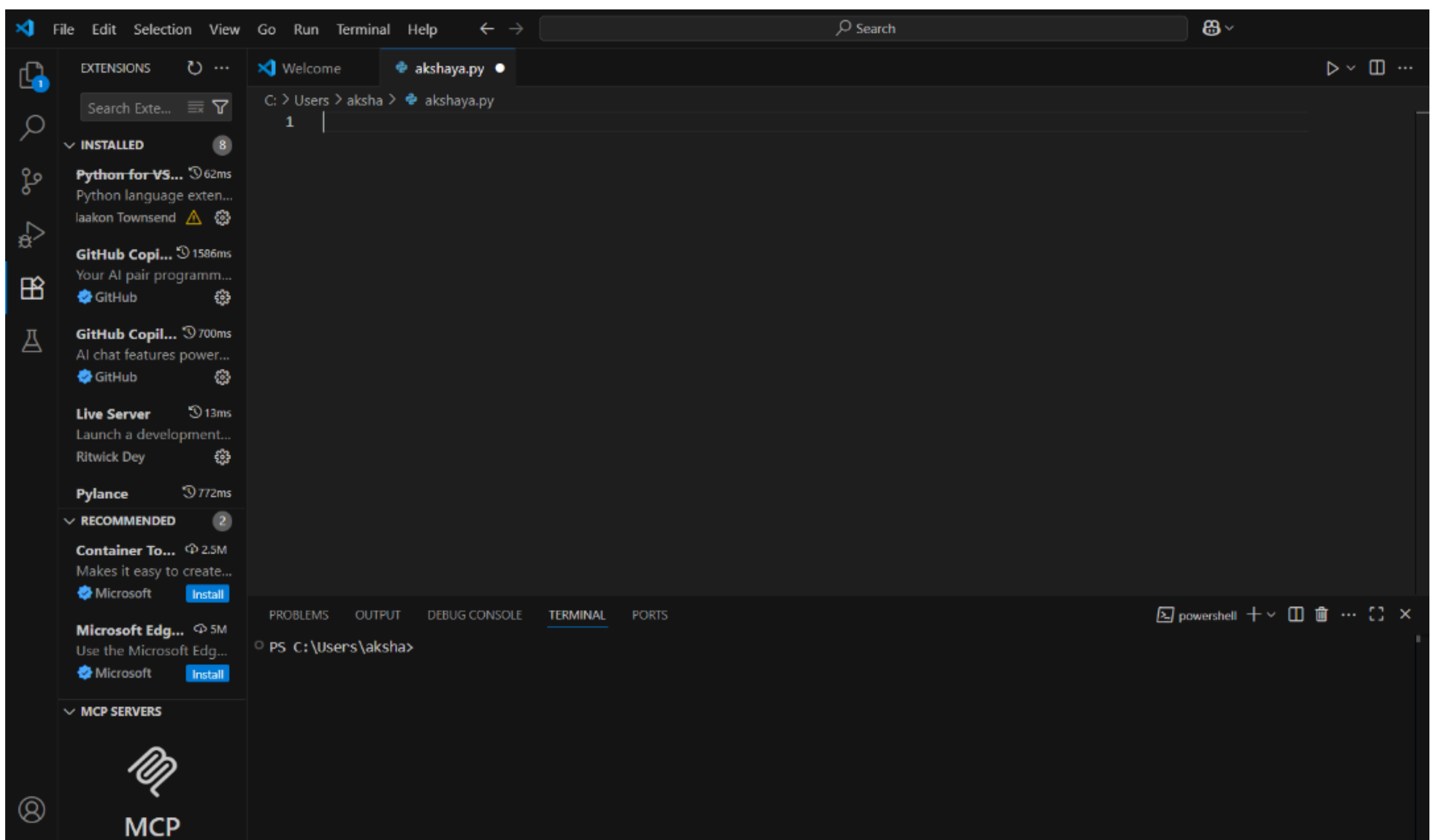
Batch : (DS)batch-4

Task 0

- Install and configure GitHub Copilot in VS Code. Take screenshots of each step.

Expected Output

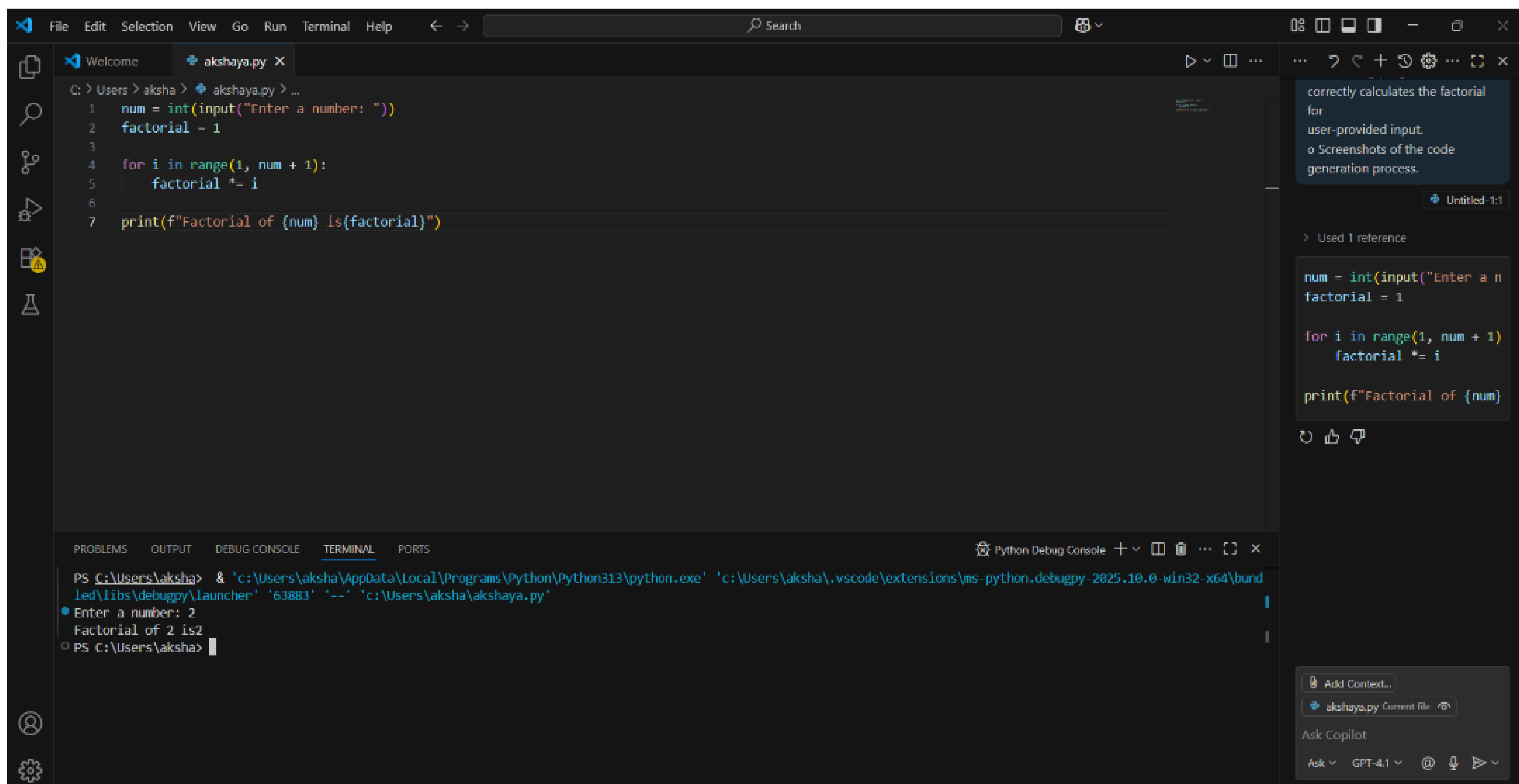
- Install and configure GitHub Copilot in VS Code. Take screenshots of each step.



Task 1: Factorial without Functions

- Description:
Use GitHub Copilot to generate a Python program that calculates the factorial of a number without defining any functions (using loops directly in the main code).
- Expected Output:
 - A working program that correctly calculates the factorial for user-provided input.
 - Screenshots of the code generation process.

prompt : generate a python program which takes a number from the user and give the factorial as the output ,without using any functions



Task 2: Improving Efficiency

- Description:
Examine the Copilot-generated code from Task 1 and demonstrate how its efficiency can be improved (e.g., removing unnecessary variables, optimizing loops).
 - Expected Output:
 - Original and improved versions of the code.
 - Explanation of how the improvements enhance performance.
- prompt : improve the efficiency from the previous code by removing unnecessary variables,optimizing loops etc...

The screenshot shows the VS Code interface with a file named `akshaya.py` open. The code in the editor is as follows:

```
1 num = int(input("Enter a number: "))
2 result = 1
3
4 for i in range(2, num + 1):
5     result *= i
6
7 print(f"Factorial of {num} is {result}")
```

Below the code editor, the terminal shows the execution of the script:

```
PS C:\Users\aksha> & 'c:\Users\aksha\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\aksha\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bund
led\libs\debugpy\launcher' '63883' '--' 'c:\Users\aksha\akshaya.py'
Enter a number: 2
Factorial of 2 is 2
PS C:\Users\aksha> c;; cd 'c:\Users\aksha'; & 'c:\Users\aksha\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\aksha\.vscode\extensions\ms-python.debugpy
-2025.10.0-win32-x64\bundled\libs\debugpy\launcher' '63916' '--' 'c:\Users\aksha\akshaya.py'
Enter a number: 5
Factorial of 5 is 120
PS C:\Users\aksha>
```

On the right side of the editor, there is a panel showing the 'Improved Code' and an 'Explanation'.

Improved Code:

```
for i in range(1, num + 1)
    factorial *= i

print(f"factorial of {num}")
```

Explanation:

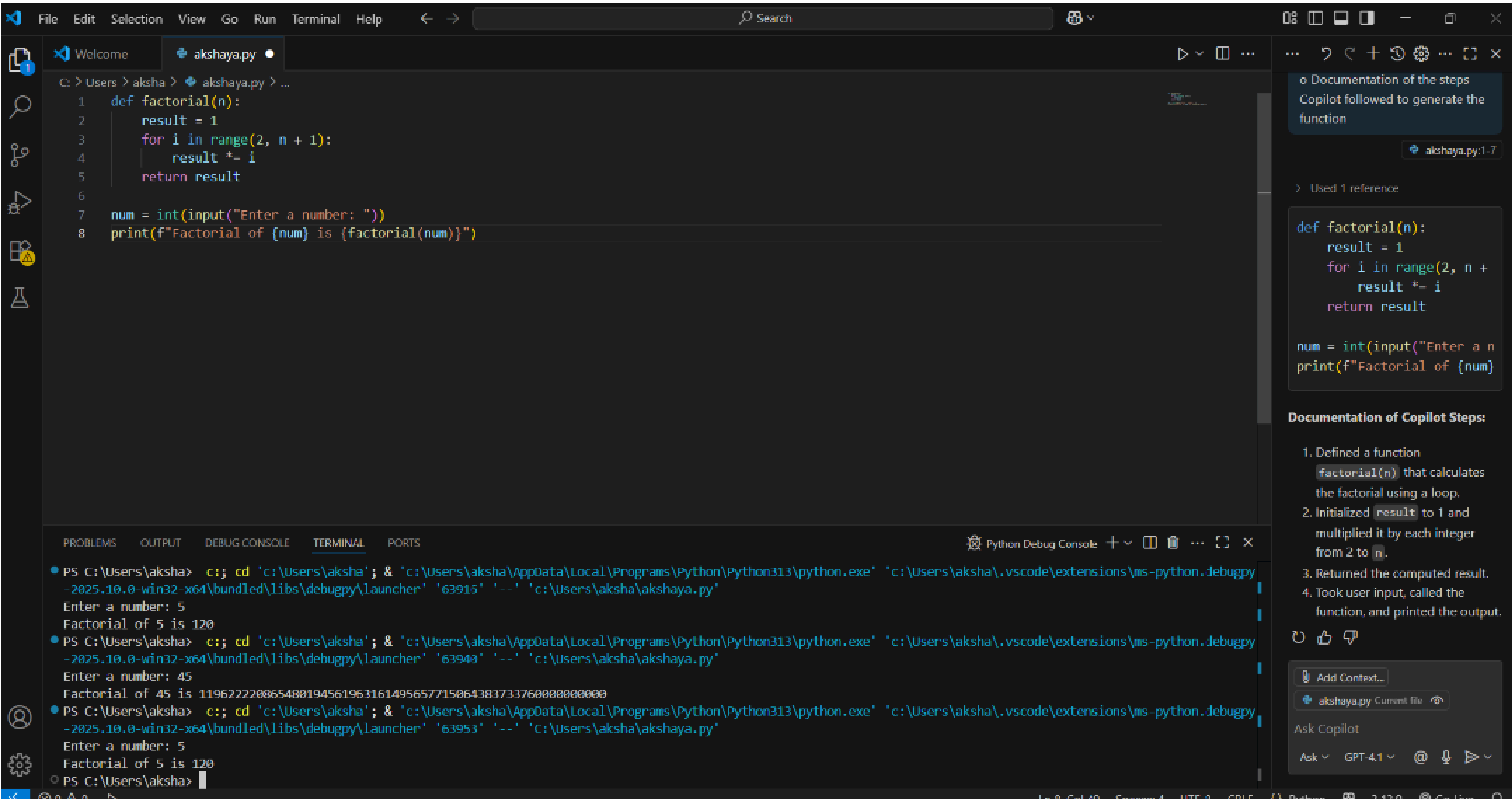
- The loop now starts from 2 instead of 1, since multiplying by 1 is redundant.
- The variable name `result` is more descriptive.
- This reduces one unnecessary multiplication per run, slightly improving efficiency, especially for large numbers.

At the bottom of the right panel, there is a section for 'Add Context...' and 'Ask Copilot'.

Task 3:Factorial with Functions

- Description:
Use GitHub Copilot to generate a Python program that calculates the factorial of a number using a user-defined function.
- Expected Output:
 - Correctly working factorial function with sample outputs.
 - Documentation of the steps Copilot followed to generate the function.

Prompt :generate a python program which take the input from the user and give factorial of it by taking an user defined function



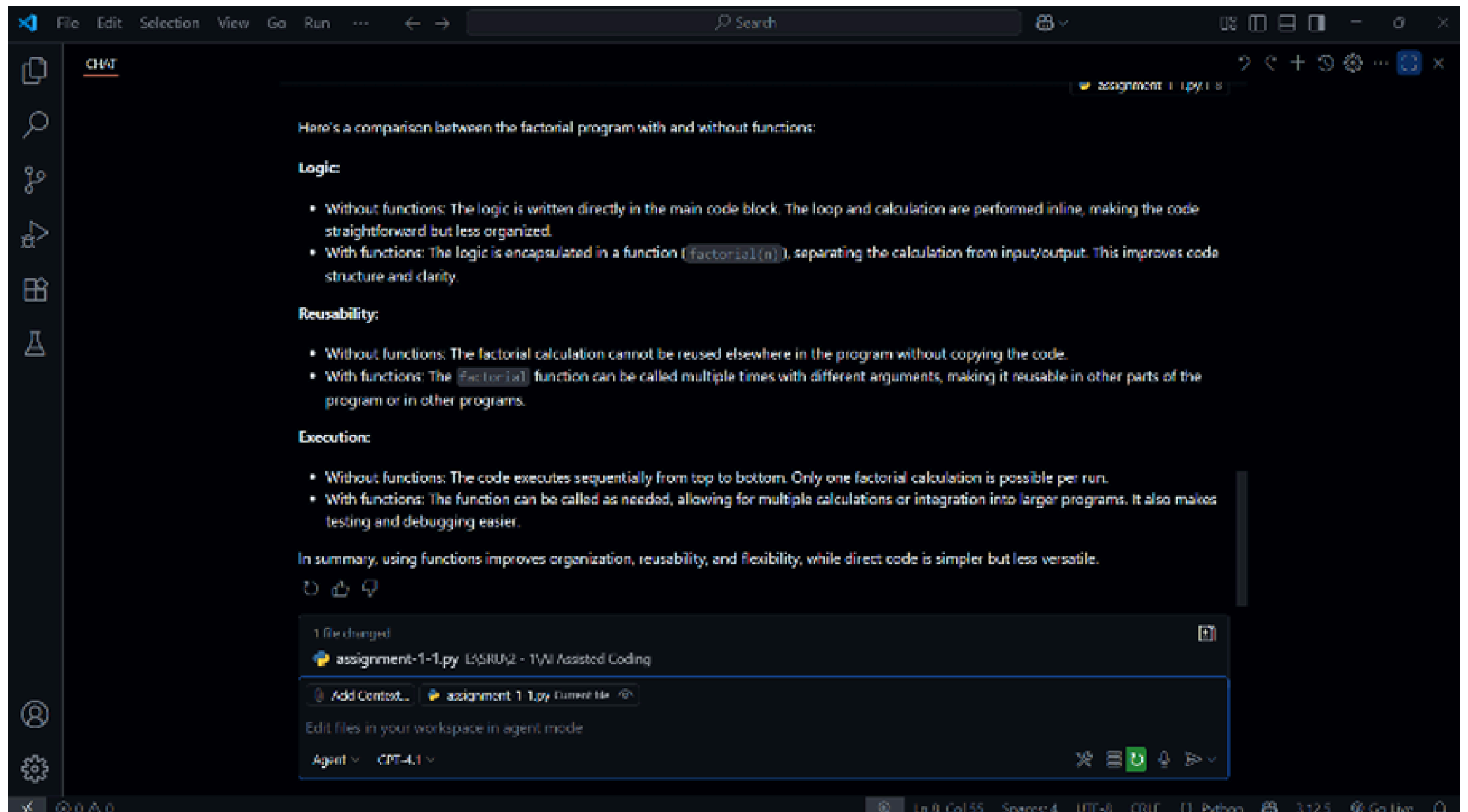
Task 4: Comparative Analysis – With vs Without Functions

- Description:
Differentiate between the Copilot-generated factorial program with functions and without functions in terms of logic, reusability, and execution.
- Expected Output:

A comparison table or short report explaining the differences

Prompt:

- Differentiate between program with functions and without functions in terms of logic, reusability, and execution.



Task 5: Iterative vs Recursive Factorial

- Description:
Prompt GitHub Copilot to generate both iterative and recursive versions of the factorial function.
- Expected Output:
 - Two correct implementations.
 - A documented comparison of logic, performance, and execution flow between iterative and recursive approaches.

- Prompt : generate both iterative and recursive versions of the factorial function

