AI ASSISTED CODING

NAME:B.ARJUN

ROLL NO:2403A510A9

ASSIGNMENT:1.1

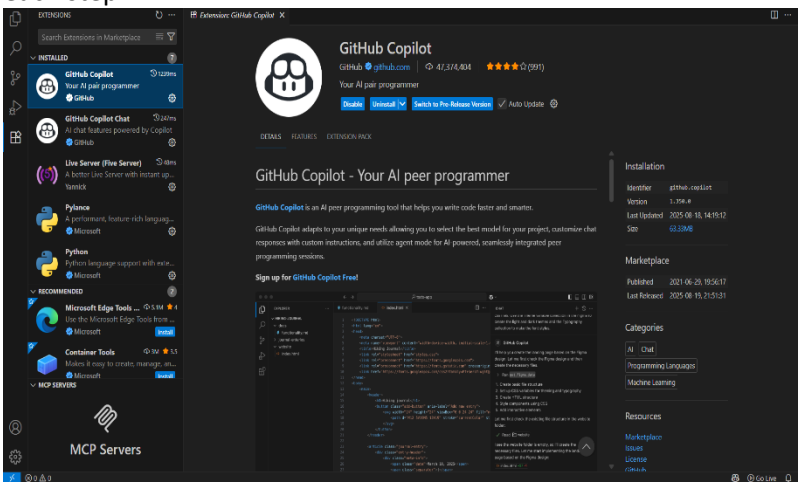
1

Task 0

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

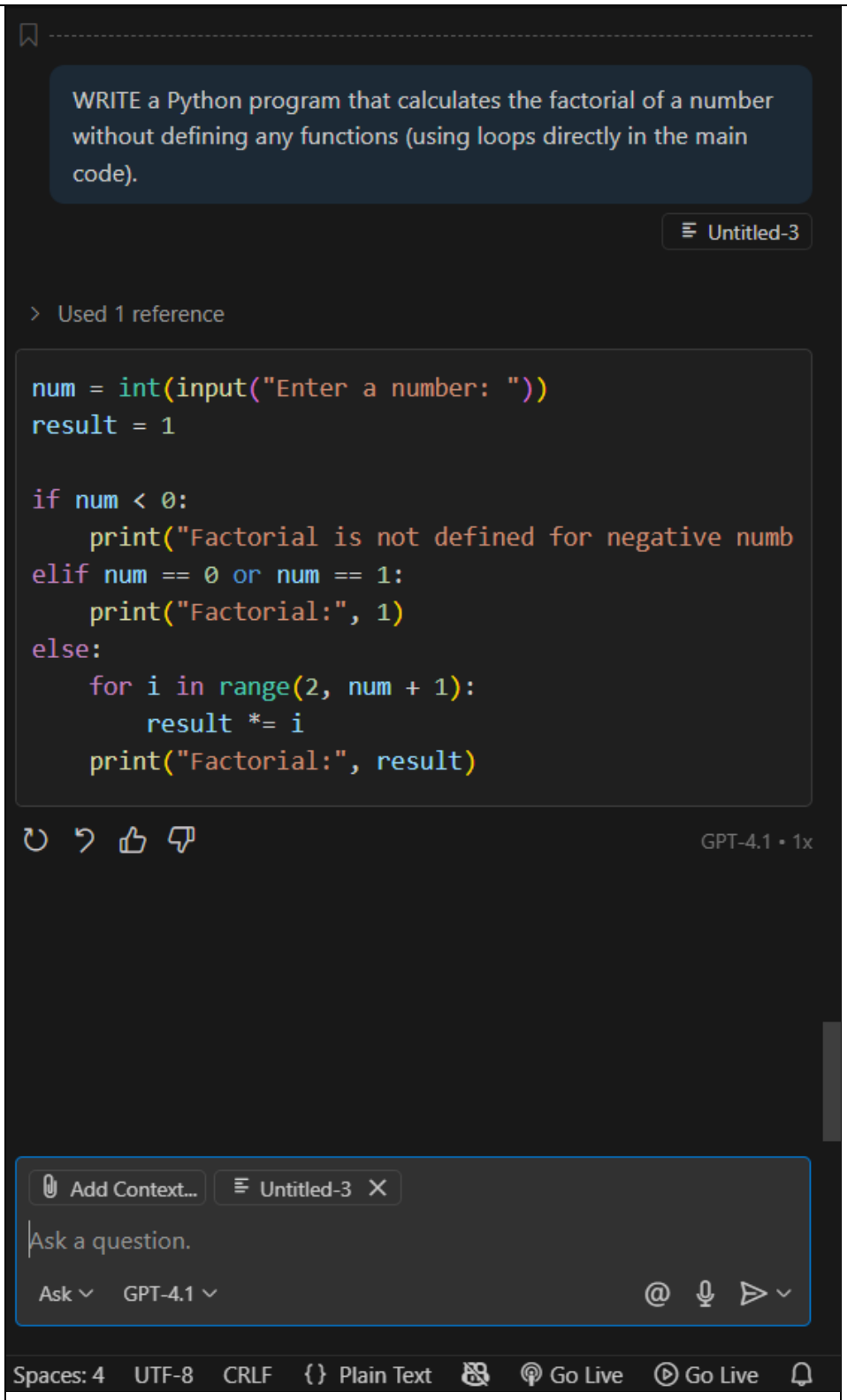
Expected Output

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



Task 1: Factorial without Functions

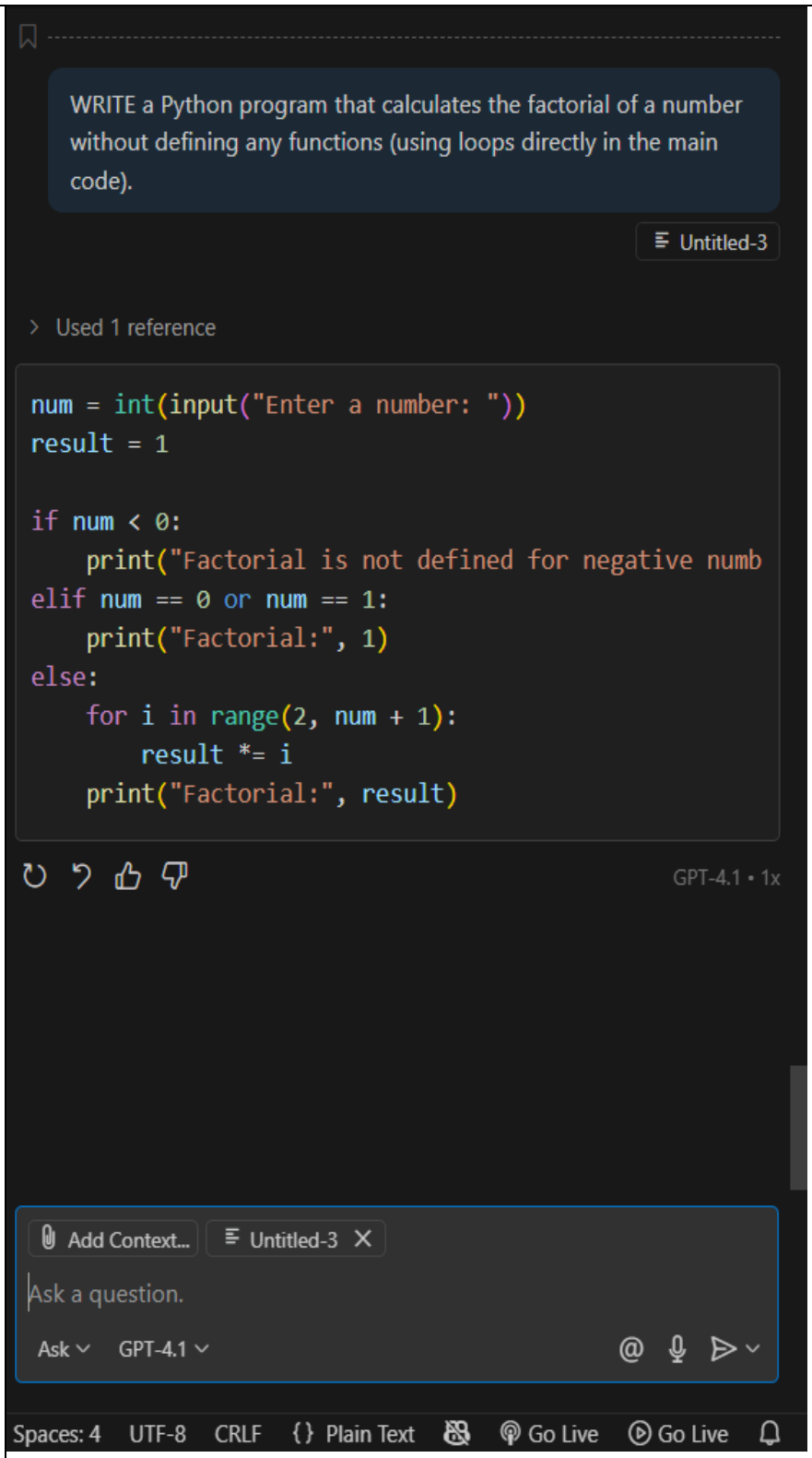
- Description:
  Use GitHub Copilot to generate a Python program that calculates the factorial of a number without defining any functions (using loops directly in the main code).
- Expected Output:
  - A working program that correctly calculates the factorial for user-provided input.
  - Screenshots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



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.
	ORIGINAL CODE:	



IMPROVED VERSION OF CODE:

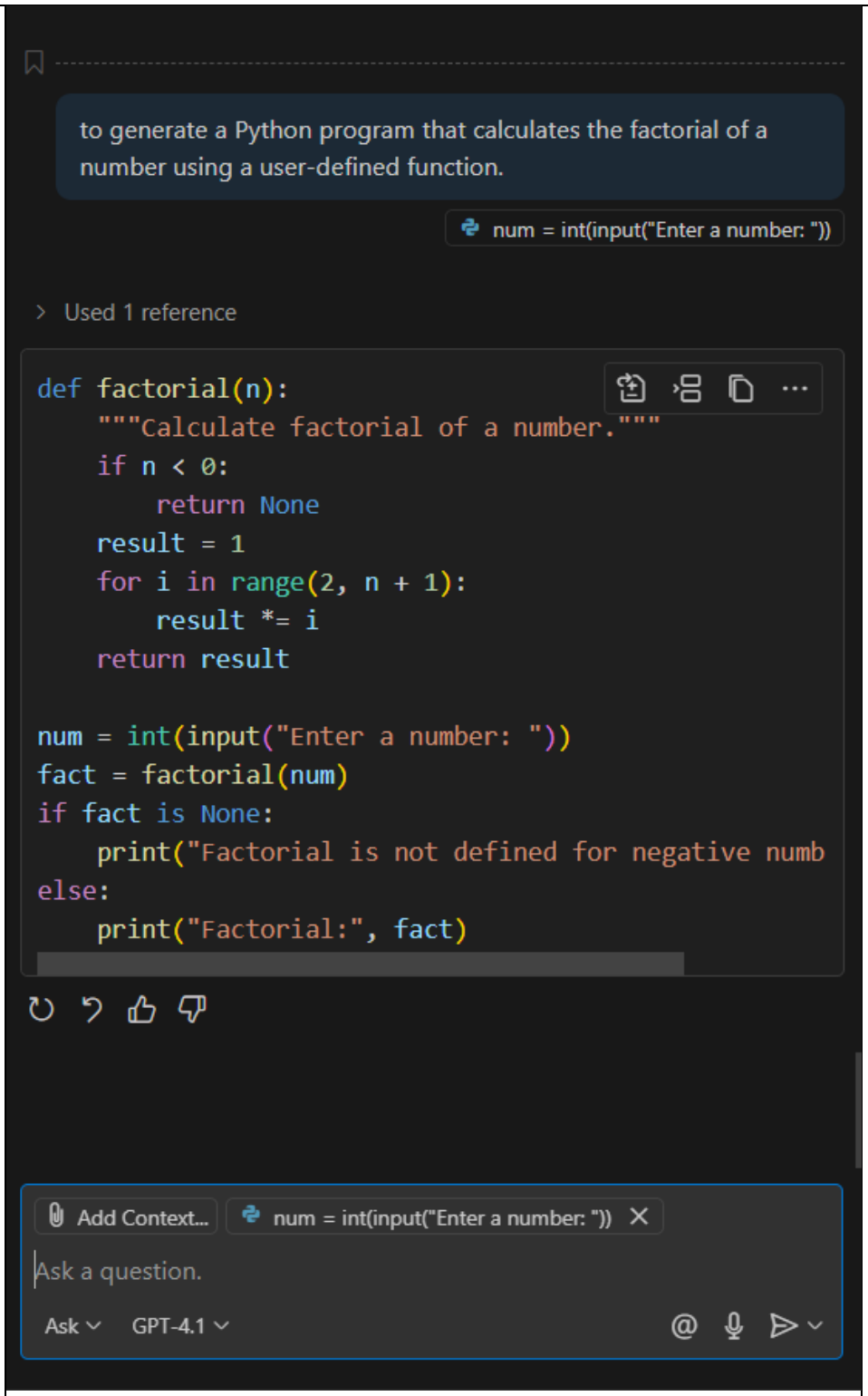
```python
num = int(input("Enter a number: "))

if num < 0:
    print("Factorial is not defined for negative numbers.")
else:
    result = 1
    for i in range(2, num + 1):
        result *= i
    print("Factorial:", result)
```

Explanation of Improvements:

- The elif block checking for num == 0 or num == 1 is removed because the loop naturally handles these cases by not executing any multiplication when num is 0 or 1.
- The result = 1 initialization is only done when needed, avoiding it when the input is negative.
- Overall, the structure is cleaner and still easy to understand.

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.



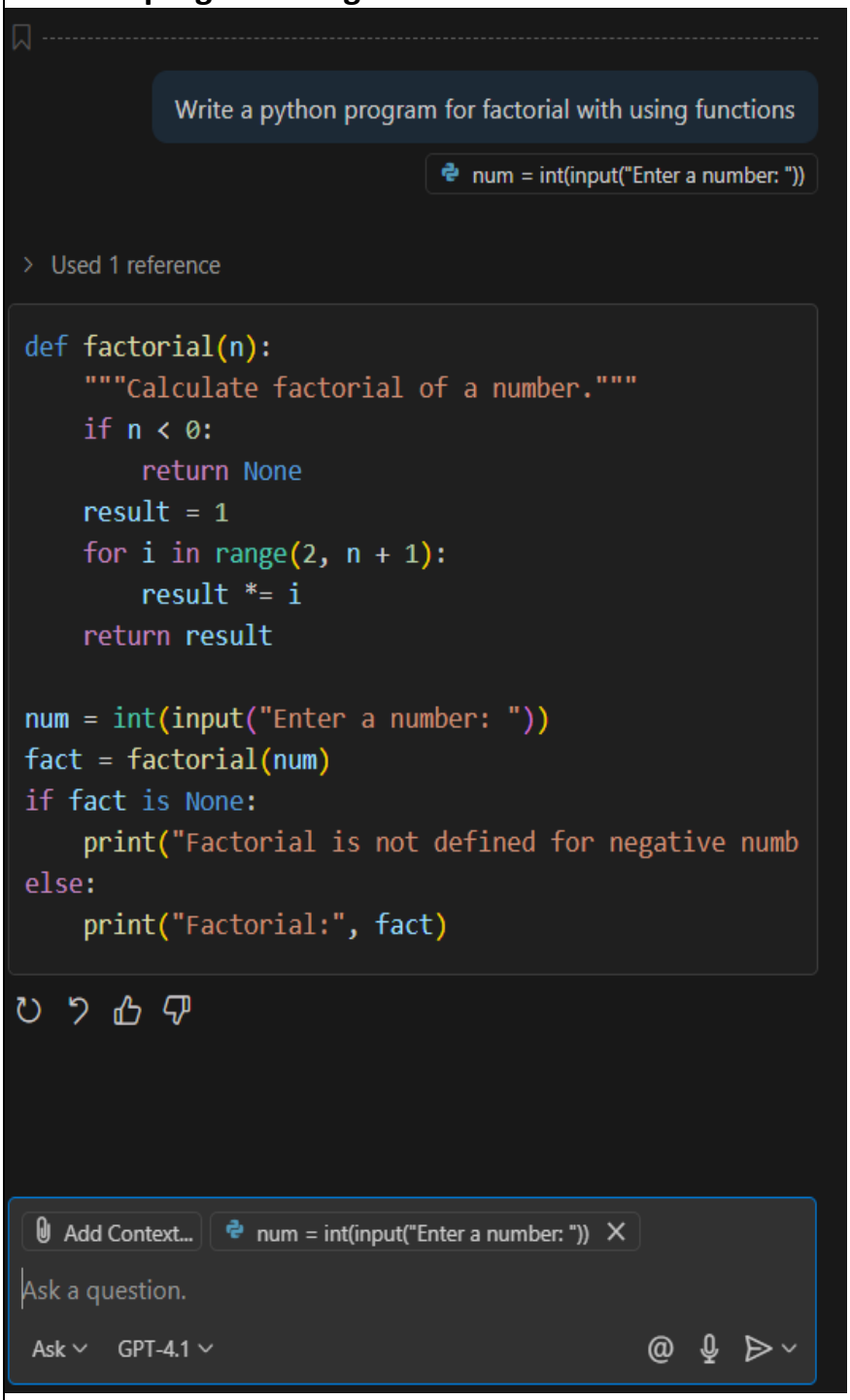
Task 4: Comparative Analysis – With vs Without Functions

- 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.

Factorial program using functions :



Factorial program without using functions:

```python
# Input: Get number from user
number = int(input("Enter a number to calculate factorial: "))

# Reusability logic: factorial calculation block
start = 2               # Starting point of multiplication
end = number + 1        # End point for range
factorial = 1           # Variable to store result

# Check for valid input
if number < 0:
    print("Factorial is not defined for negative numbers.")
else:
    for i in range(start, end):
        factorial *= i  # Multiply and accumulate
    print("Factorial:", factorial)

```

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

ITERATIVE VERSION:.

```python
def factorial_iterative(n):
    if n < 0:
        raise ValueError("Factorial is not defined for negative numbers.")
    result = 1
    for i in range(2, n + 1):
        result *= i
    return result

```

RECURSIVE VERSION:

```python
def factorial_recursive(n):
    if n < 0:
        raise ValueError("Factorial is not defined for negative numbers.")
    if n == 0 or n == 1:
        return 1
    return n * factorial_recursive(n - 1)

```

Submission Requirements

1. Generate code for each task with comments.
2. Screenshots of Copilot suggestions.
3. Comparative analysis reports (Task 4 and Task 5).

4. Sample inputs/outputs demonstrating correctness.

Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots

Evaluation Criteria:

Criteria	Max Marks
Successful Setup of Copilot	0.5
Comparative Analysis – With vs Without Functions	1
Iterative vs Recursive Factorial	1
Total	**2.5 Marks**