**AI ASSISTED CODING LAB**

**ASSIGNMENT 10.2**

**ENROLLMENT NO :**2503A51L25

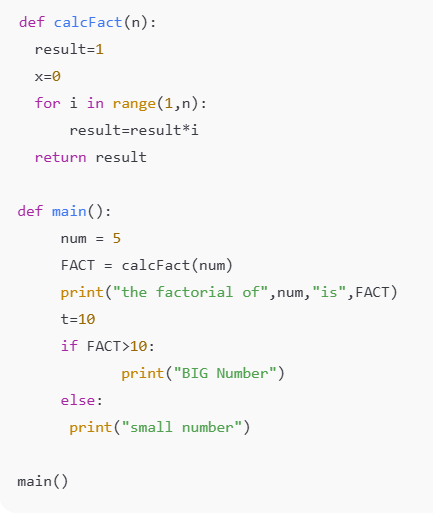
**BATCH NO:** 19

**NAME:** SRIRAMOJU AJAY

TASK1

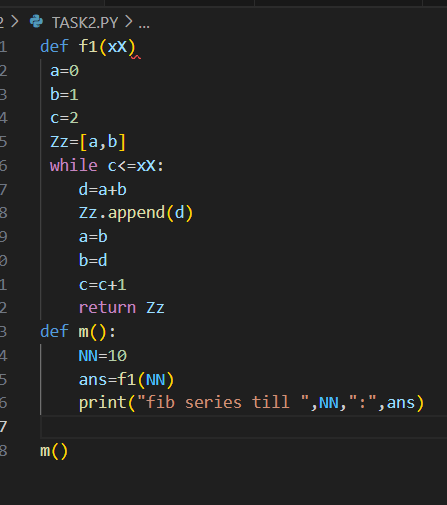
TASK1 DESCRIPTION:-**AI-Assisted Code Review (Basic Errors)**

* Write python program as shown below.
* Use an AI assistant to review and suggest corrections

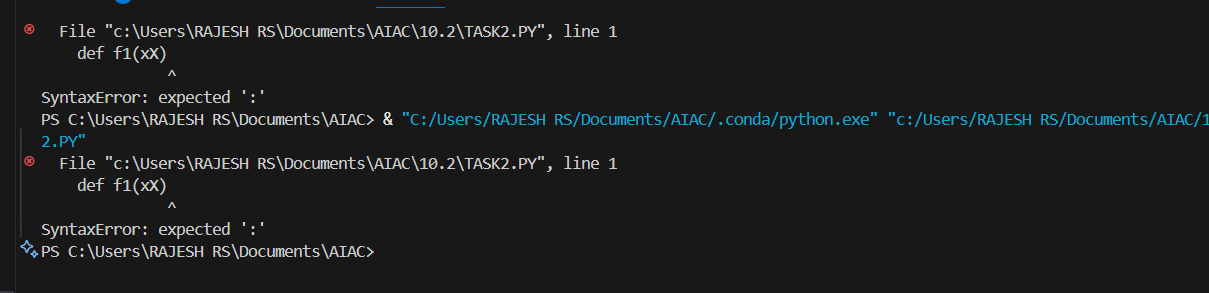


CODE:-

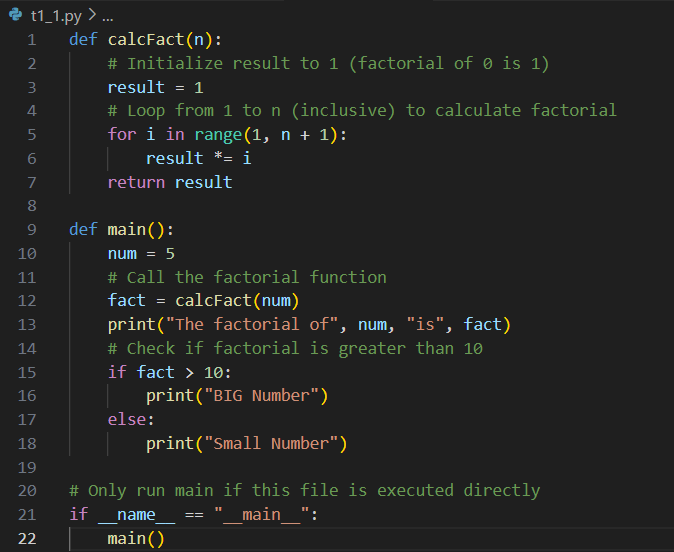
Error Code:-



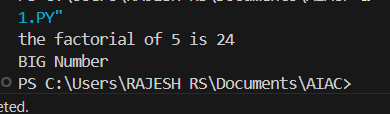
OUTPUT:-



Correct Code:-



OUTPUT:-



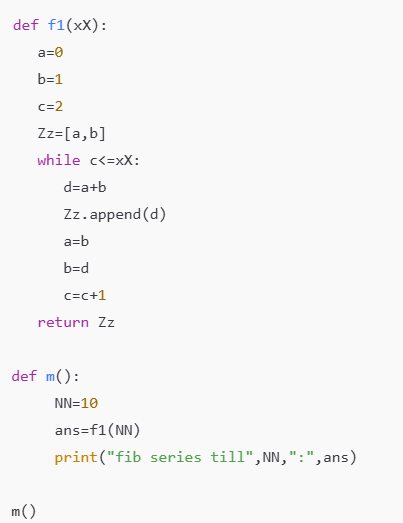
OBSERVATION:-

In this task, I wrote a Python program that contained basic syntax or logical errors. I then used an AI assistant to review the code and suggest corrections. The AI successfully identified the errors and provided a corrected version of the code, along with explanations. This helped me understand common mistakes and how to fix them efficiently.

TASK2

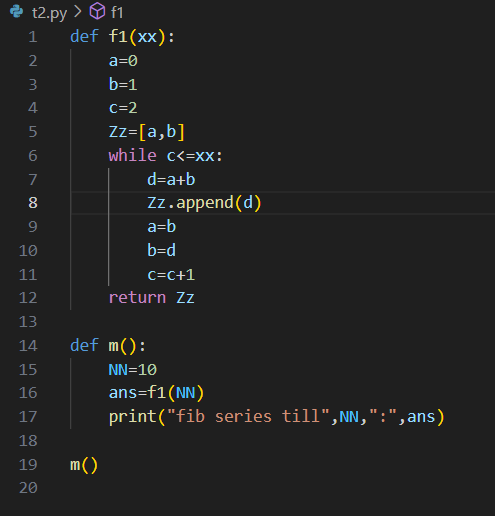
TASK2 DESCRIPTION:- **Automatic Inline Comments**

* Write the Python code for Fibonacci as shown below and execute.
* Ask AI to improve variable names, add comments, and apply PEP8 formatting (cleaned up).
* Students evaluate which suggestions improve readability most.

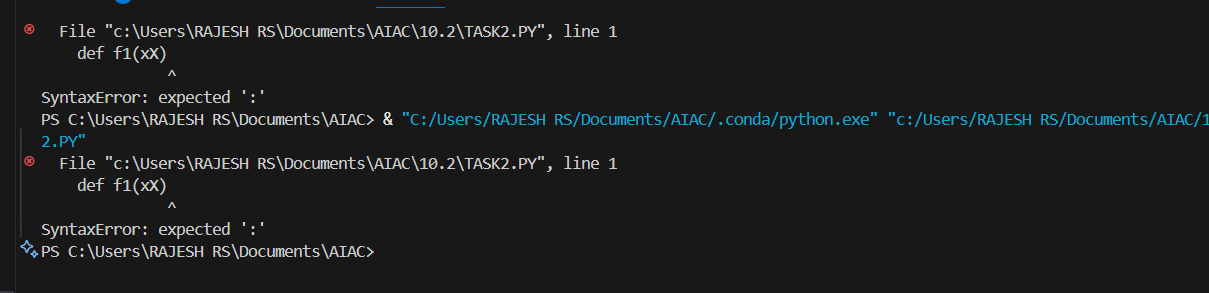


CODE:-

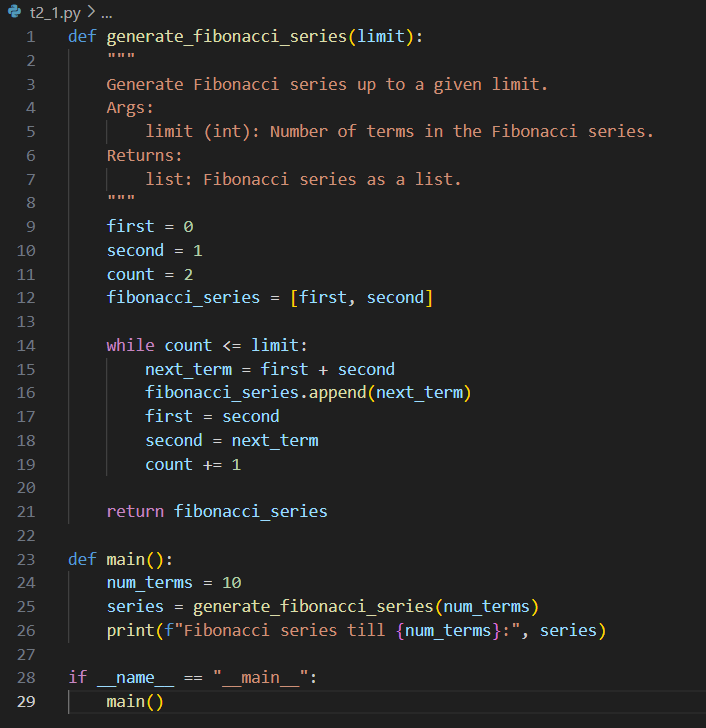
Given Code:-



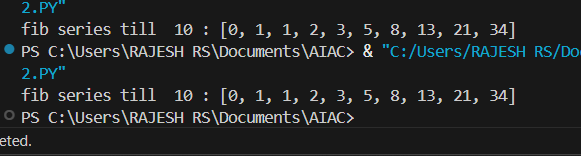
Output:-



AI Generated Code:-



Output:-



OBSERVATION:-

For this task, I implemented a Python program to generate Fibonacci numbers. After writing the initial code, I asked the AI assistant to improve variable names, add comments, and format the code according to PEP8 guidelines. The AI-generated version was clearer and easier to understand, especially for beginners.

TASK3

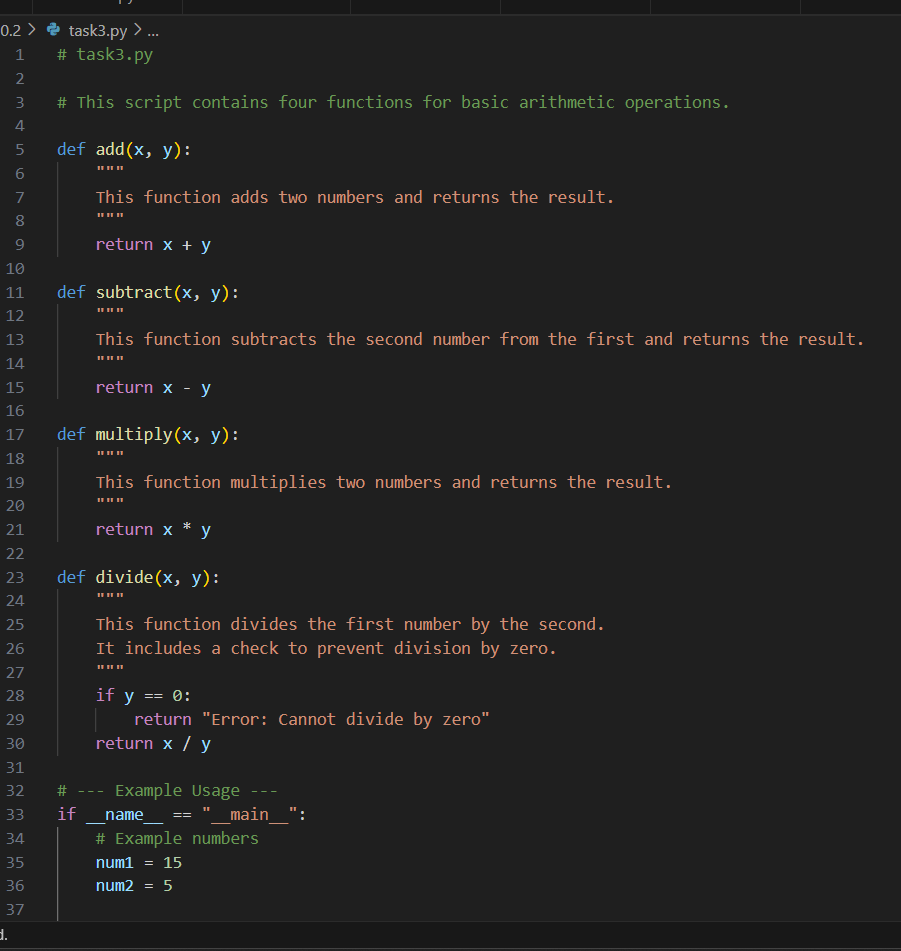
TASK3 DESCRIPTION:-

* Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide).
* Incorporate manual **docstring** in code with NumPy Style
* Use AI assistance to generate a module-level docstring + individual function docstrings.
* Compare the AI-generated docstring with your manually written one

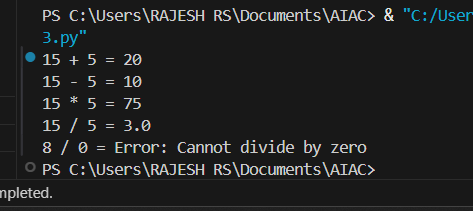
PROMPT 1:

Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide)

CODE GENERATED :



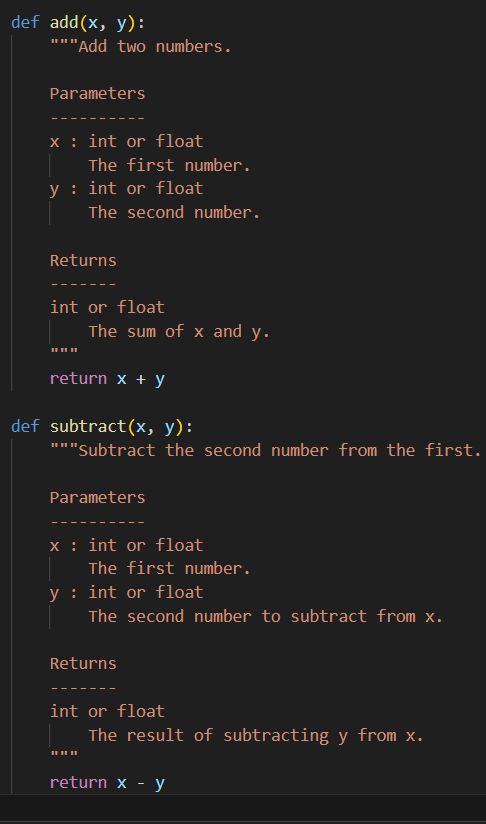
OUTPUT:

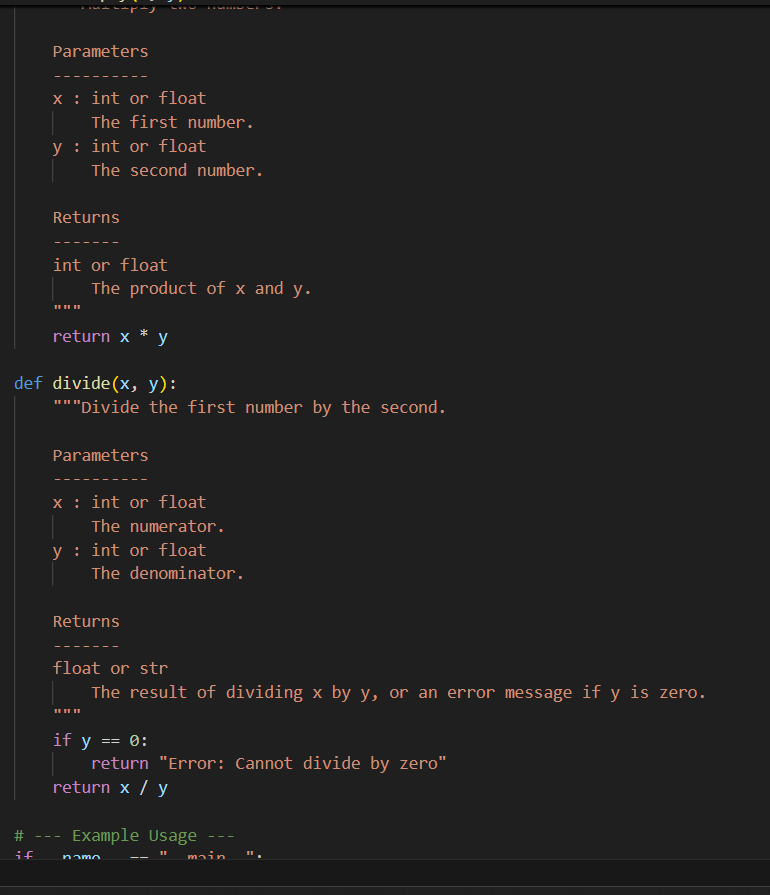


PROMOT 2:

Incorporate manual **docstring** in code with NumPy Style

CODE GENERATED:

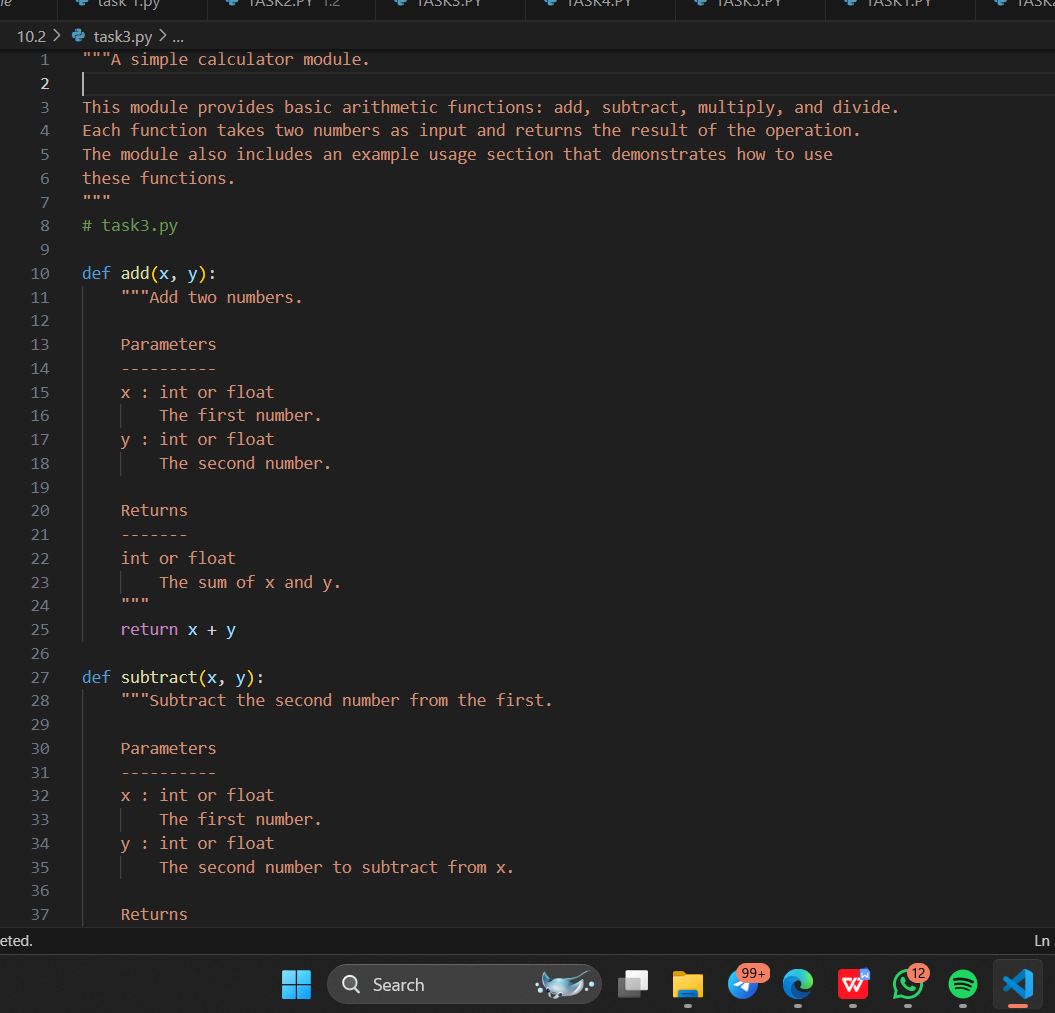




**PROMPT 3:**

 Generate a module-level docstring + individual function docstrings.

CODE GENERATED :



OBSERVATION:-

In this task, I created a Python script containing multiple functions such as add, subtract, multiply, and divide. I manually wrote docstrings using the NumPy style format to describe each function’s purpose and parameters. Then, I asked the AI to generate its own module-level and function-level docstrings.