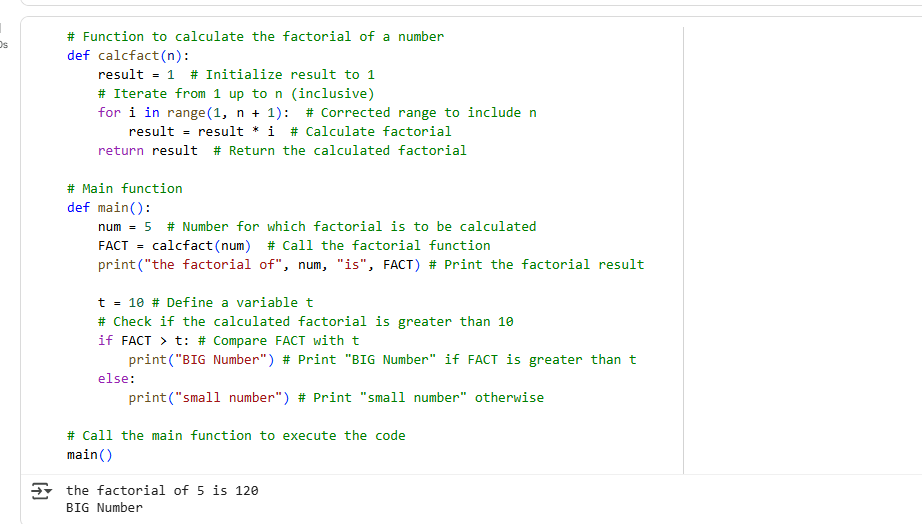
Assignment 10.2

HTNO:2403A51347

Batch:14

Name:THARUNI.K

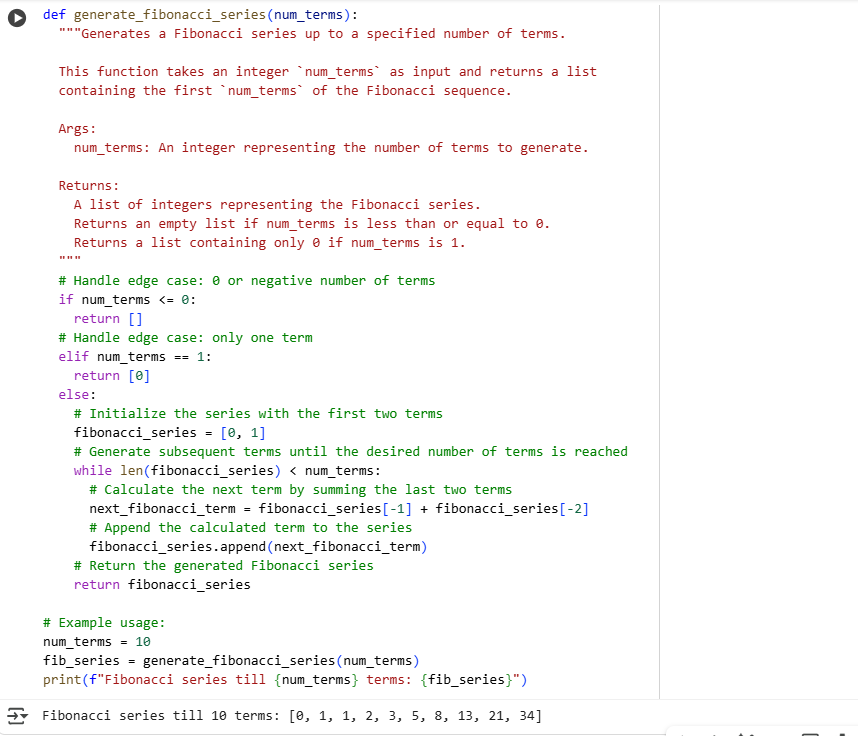
Task Description#1 AI-Assisted Code Review (Basic Errors)  
• Write python program as shown below.  
• Use an AI assistant to review and suggest corrections.



Prompt:correct the python code and give comment lines for better understanding.

Observation: The corrected code successfully calculates the factorial of 5 as 120 and correctly identifies it as a "BIG Number" based on the condition FACT > t where t is 10.

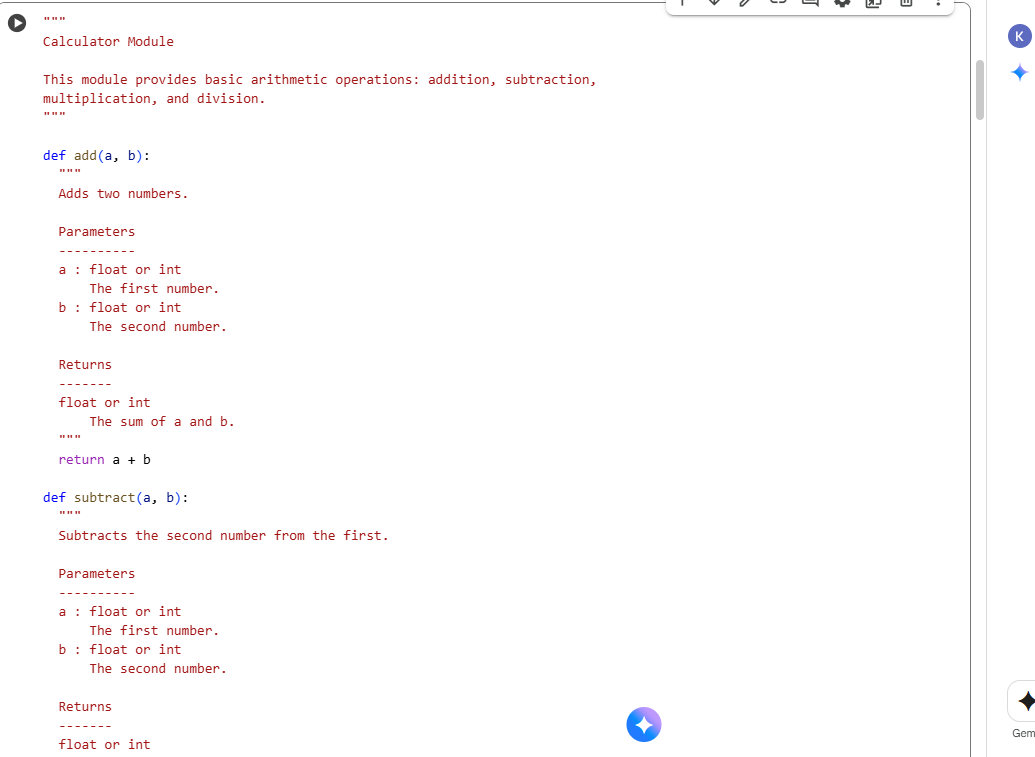
Task Description#2 Automatic Inline Comments  
• Write the Python code for Fibonacci as shown below and execute.  
• Ask AI to improve variable names, add comments, and apply PEP8 formatting  
(cleaned up).  
• Students evaluate which suggestions improve readability most. one.

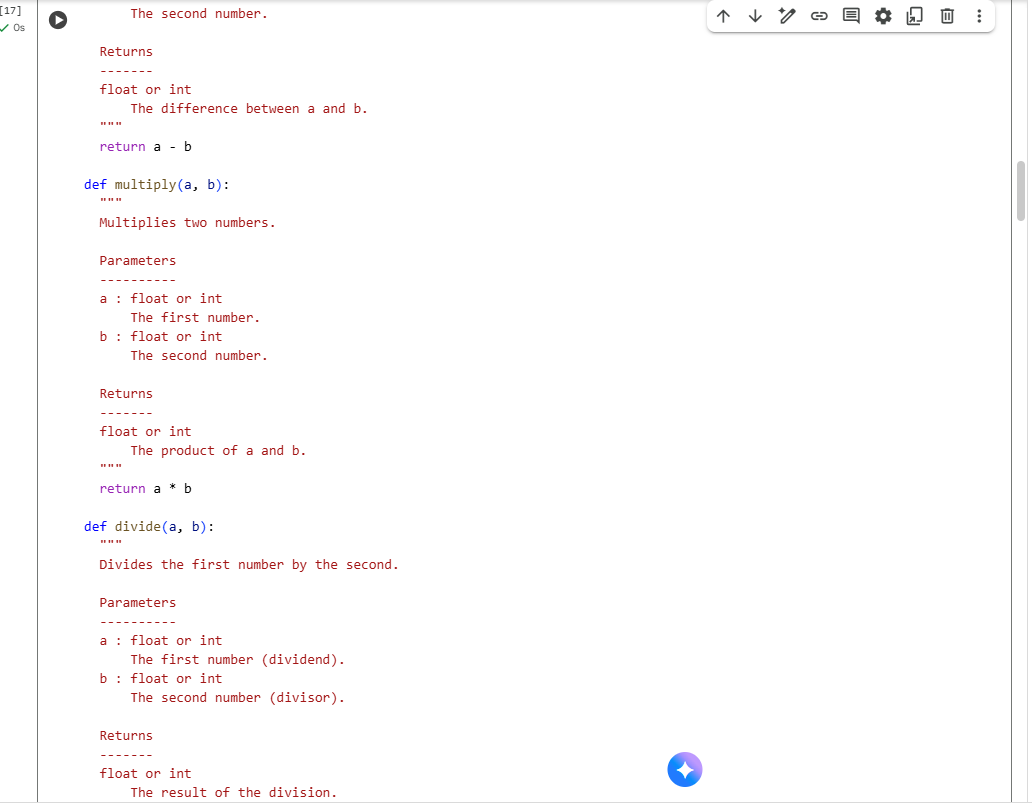


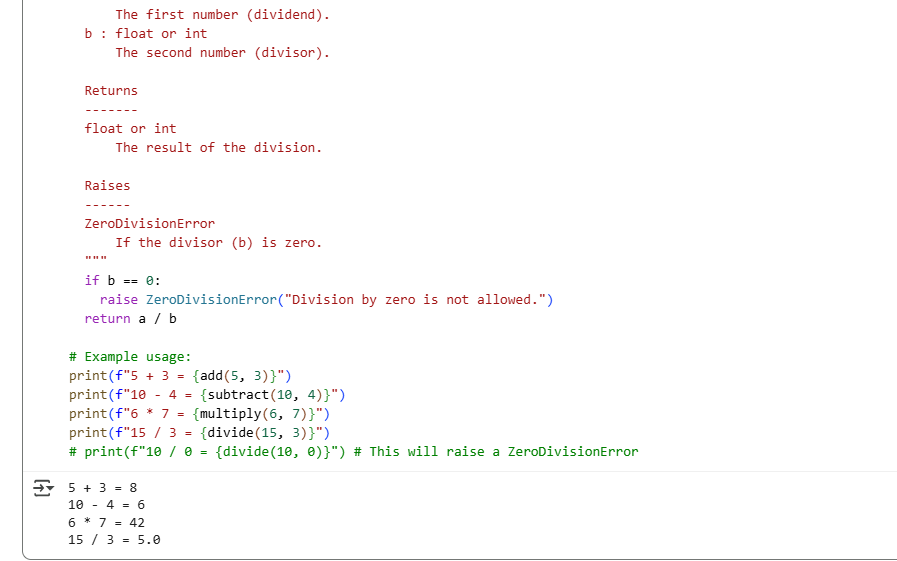
Prompt: I need python code of fibnocci series with comment lines.

Observation: the corrected code successfully generates and prints the Fibonacci series up to 10 terms, which is [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]. The inclusion of comments, improved variable names, and PEP8 formatting also significantly enhance the readability and maintainability of the code.

Task Description#3  
• 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: Need this code using numpy components.

Observation: The code successfully defines functions for basic arithmetic operations (addition, subtraction, multiplication, and division) and includes dynamic output demonstrating their usage. The code also correctly calculates and prints the results of the example operations: 5 + 3 = 8, 10 - 4 = 6, 6 \* 7 = 42, and 15 / 3 = 5.0. The inclusion of NumPy-style docstrings provides clear documentation for each function.