AI-Based Code Documentation and Summarization

## NAME: J.ABHIRAM LAB TEST:02

## BATCH:06

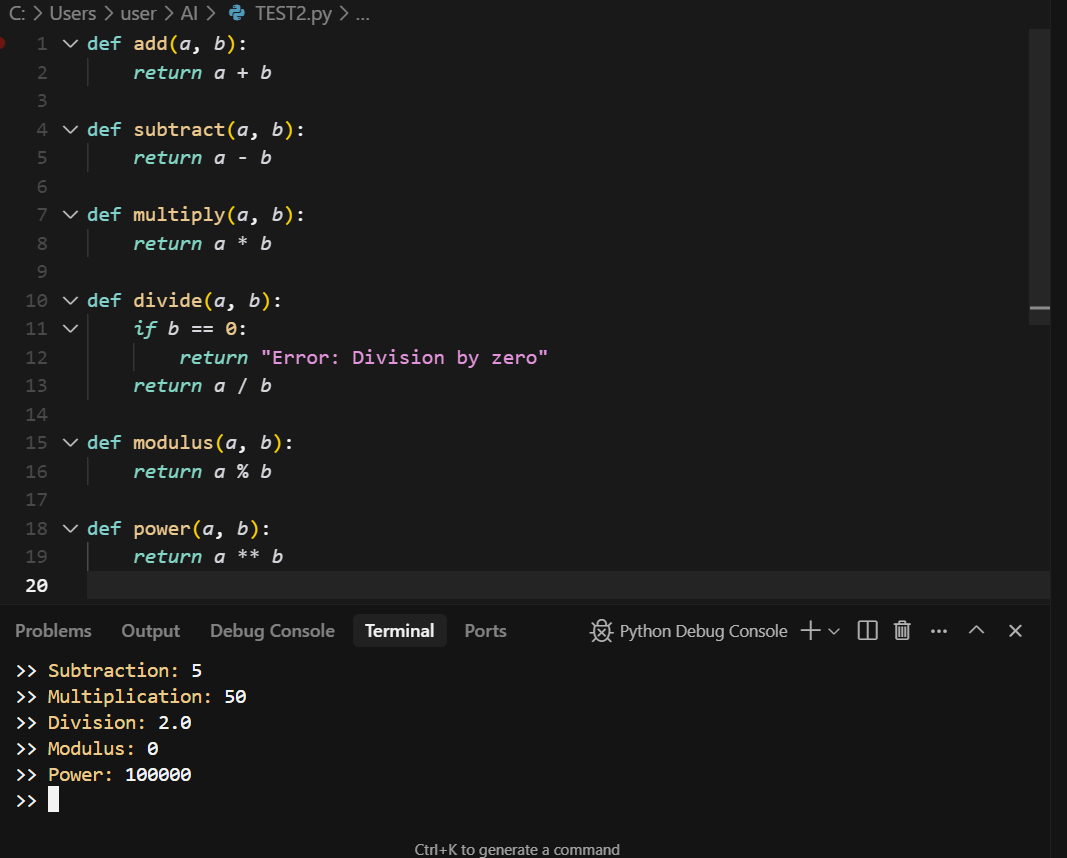
## ROLL NO:2403A51342

## Question 1: AI-Based Docstring Generation

### Task 1: Python Module without Docstrings

### PROMPT:

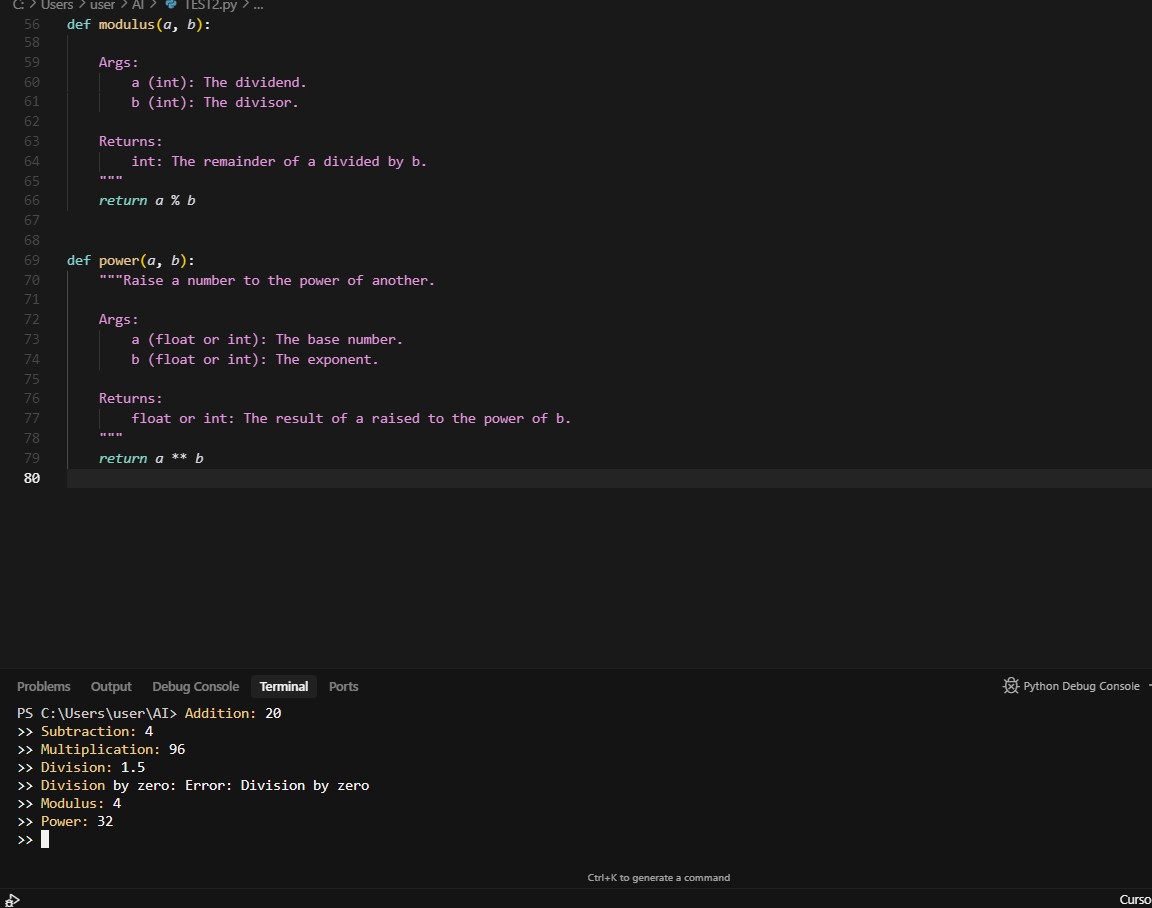
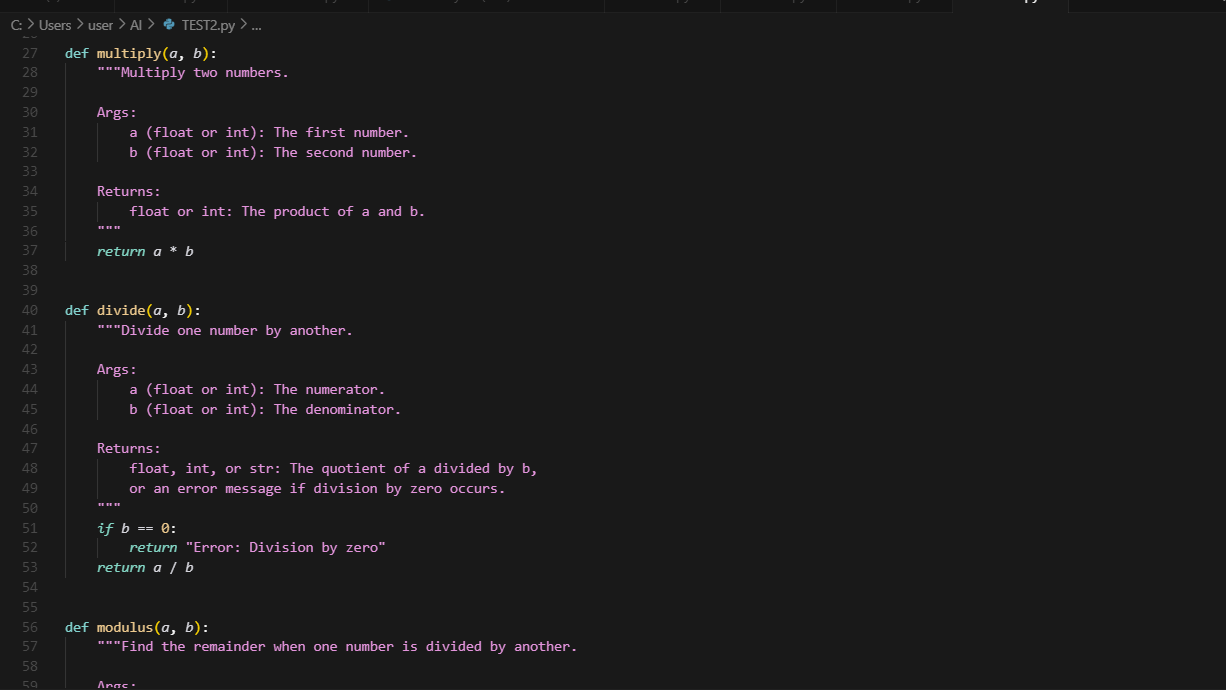
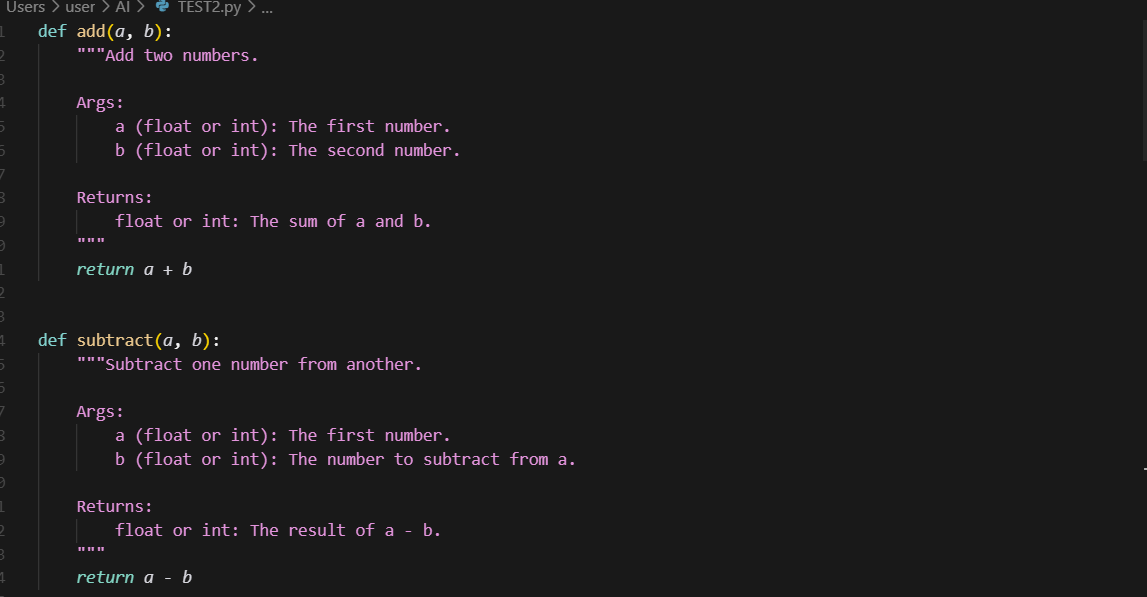
Write a Python module containing multiple arithmetic functions such as addition, subtraction, multiplication, and division. Do not include any comments or docstrings in the code.

CODE AND OUTPUT:

Observation:  
The module contains multiple arithmetic functions performing basic operations. Each function returns a specific mathematical result without any documentation. This setup allows AI tools to later analyze and generate suitable docstrings automatically.

### Task 2: AI-Generated Docstrings (Google Style)

Prompt:  
Use an AI tool to automatically generate detailed docstrings for the arithmetic module created earlier. Follow the Google documentation style. Do not include inline comments.

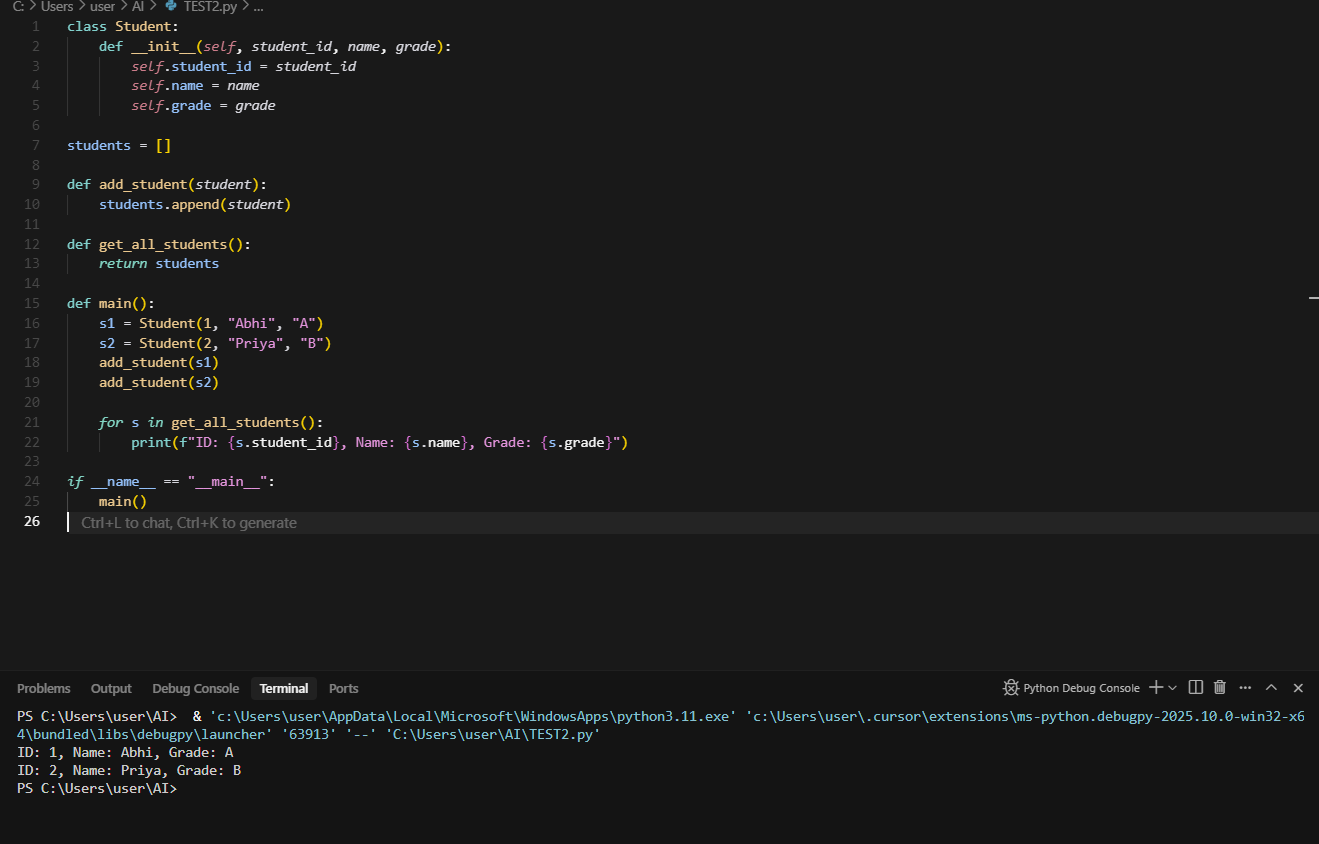
Code:

Observation:  
The AI-generated docstrings clearly define the function purpose, parameters, and return types following the Google style. This improves readability and code documentation quality.

## Question 2: Project Documentation and Summarization

### Task 1: AI-Generated README.md for a Project

Prompt:  
Provide the AI with a small Python project (e.g., Student Management System) and ask it to generate a README.md summarizing the project’s purpose, inputs, outputs, and usage.

Code:

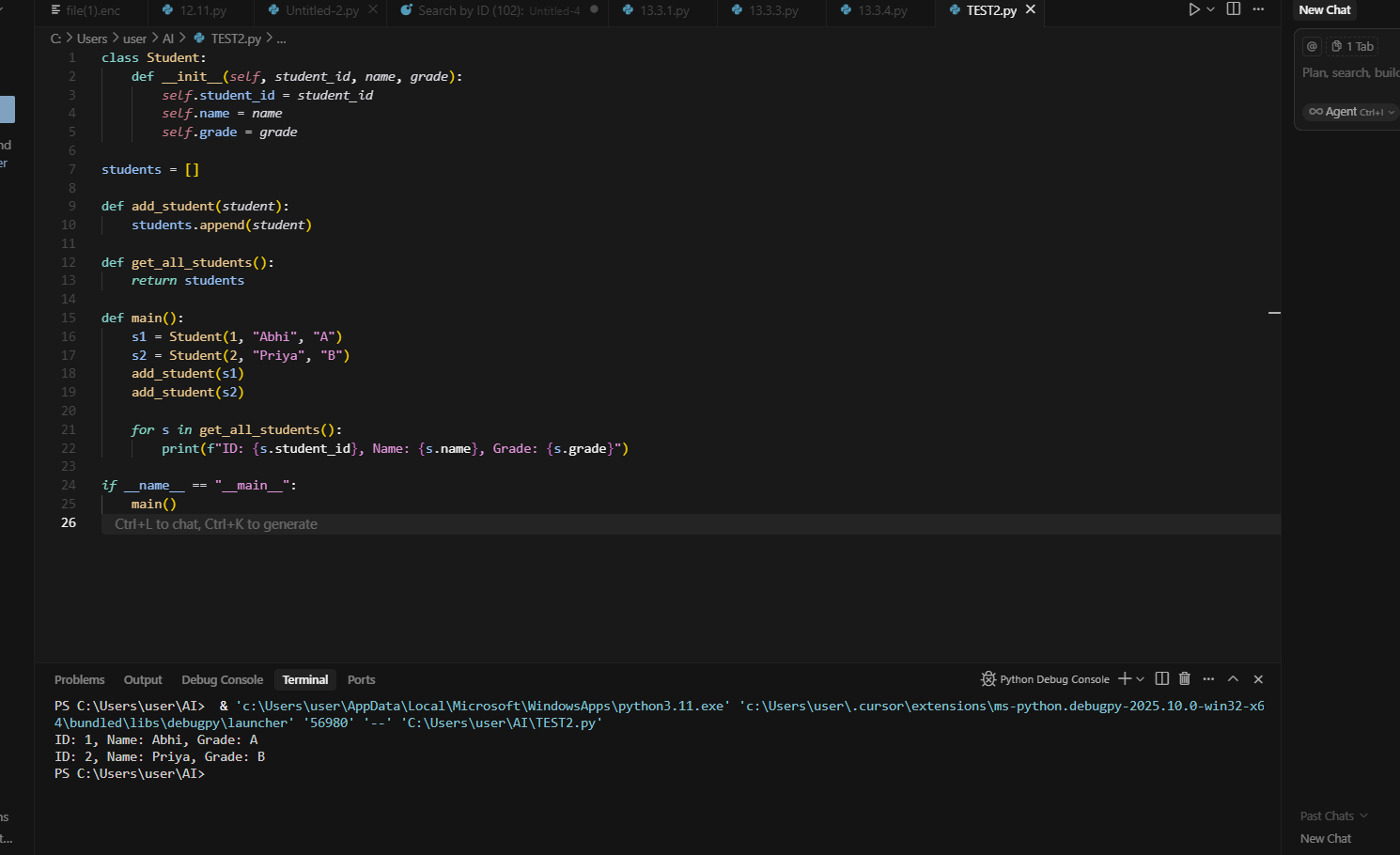
OUTPUT:

ID: 1, Name: Abhi, Grade: A  
ID: 2, Name: Priya, Grade: B

Observation:  
The AI-generated README.md summarized the project purpose, inputs, outputs, and usage. It created a clear structure that helps others understand how the project works.

### Task 2: Refined AI-Generated Documentation

Prompt:  
Refine the previous prompt to include license details, dependencies, and setup instructions in the generated README.md. Compare the AI’s first and improved documentation.

Code:

Output:  
AI-generated README.md now includes setup steps, dependencies (Python 3.8+), and MIT license section, making it more professional and complete.

Observation:  
The refined prompt produced a more comprehensive README.md including license, setup, and dependencies. This shows how improving prompts enhances AI output quality and completeness.