AI ASSISTED CODING

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

SAI SREEYESH **2303A510h8**

BATCH – 03 **16 – 01 – 2026**

ASSIGNMENT – 2.5

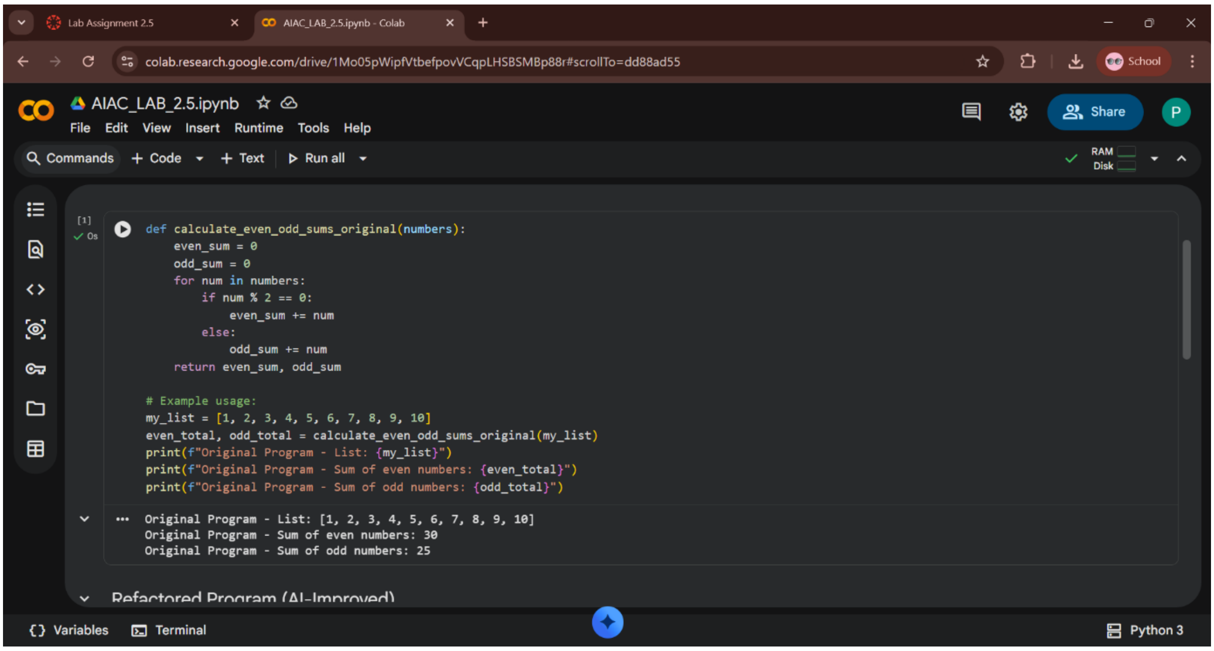
Lab 2: Exploring Additional AI Coding Tools beyond Copilot – Gemini (Colab) and Cursor AI.

Task 1: Refactoring Odd/Even Logic (List Version).

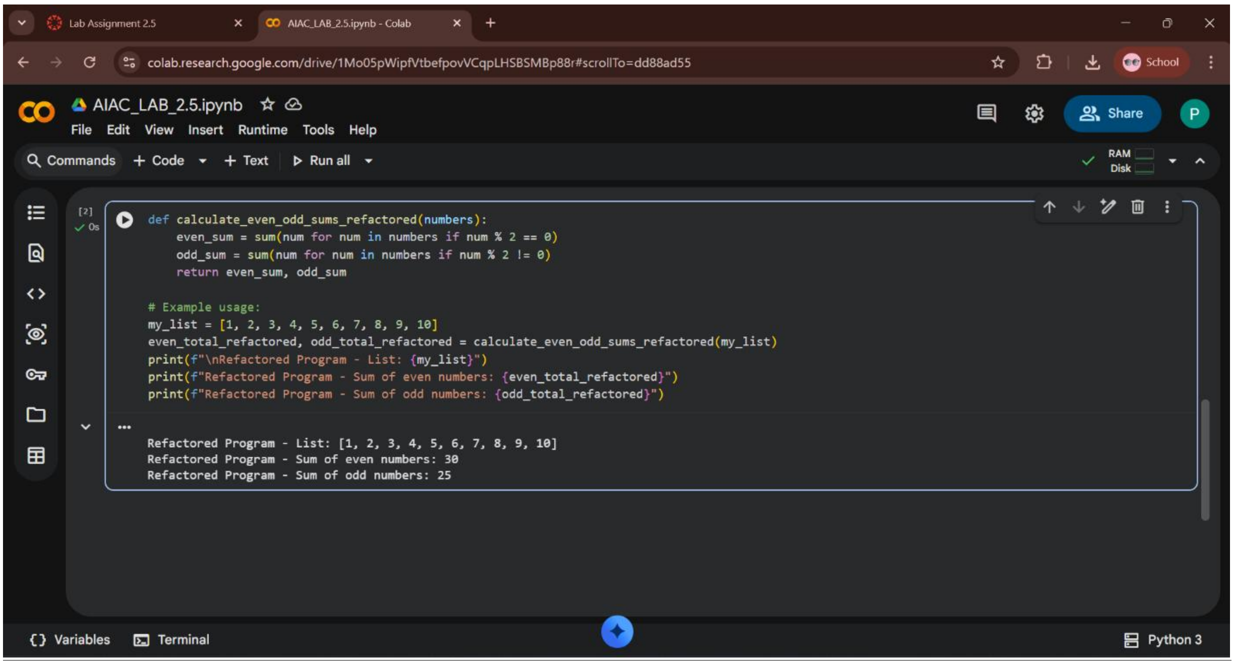
Prompt: Write a Python program that takes a list of integers and calculates the sum of even numbers and the sum of odd numbers separately. After generating the working program, refactor (improve) the code using AI to make it cleaner, more readable, and efficient. Provide both the original code and the refactored version.

Code:

Original Code:



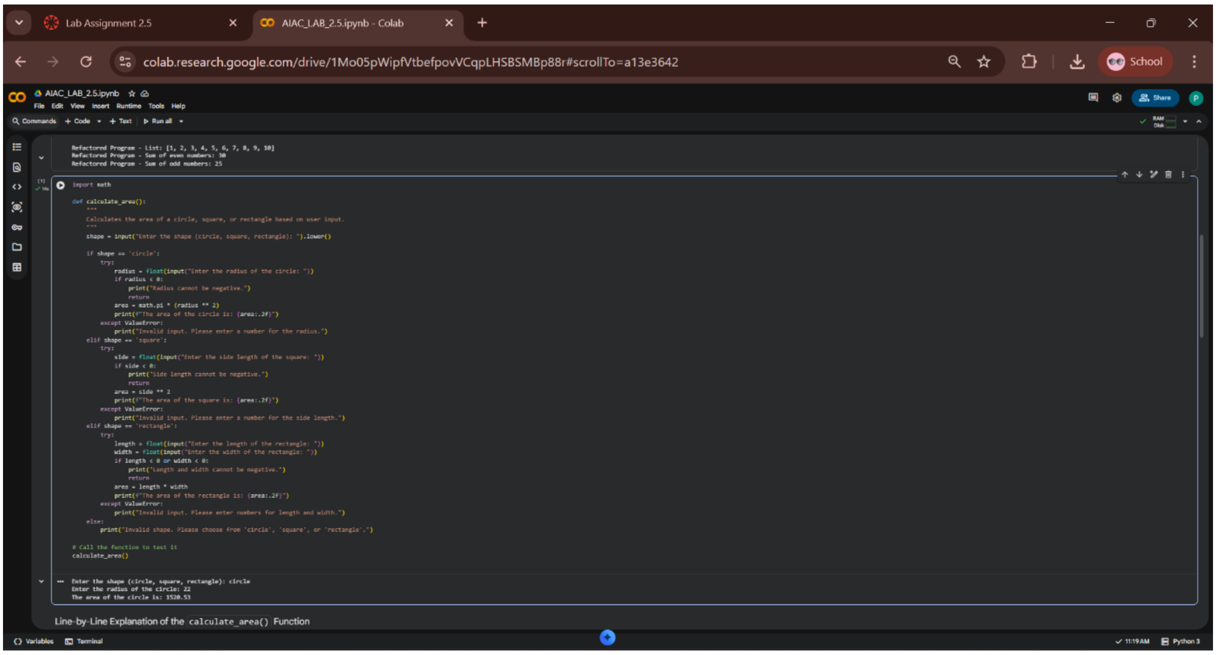
AI Improved Code:



Task 2: Area Calculation Explanation.

Prompt: Write a Python function that calculates the area of different shapes (circle, square, and rectangle) based on user input. Then explain the function line by line in simple language so that a junior developer can easily understand it.

Code:



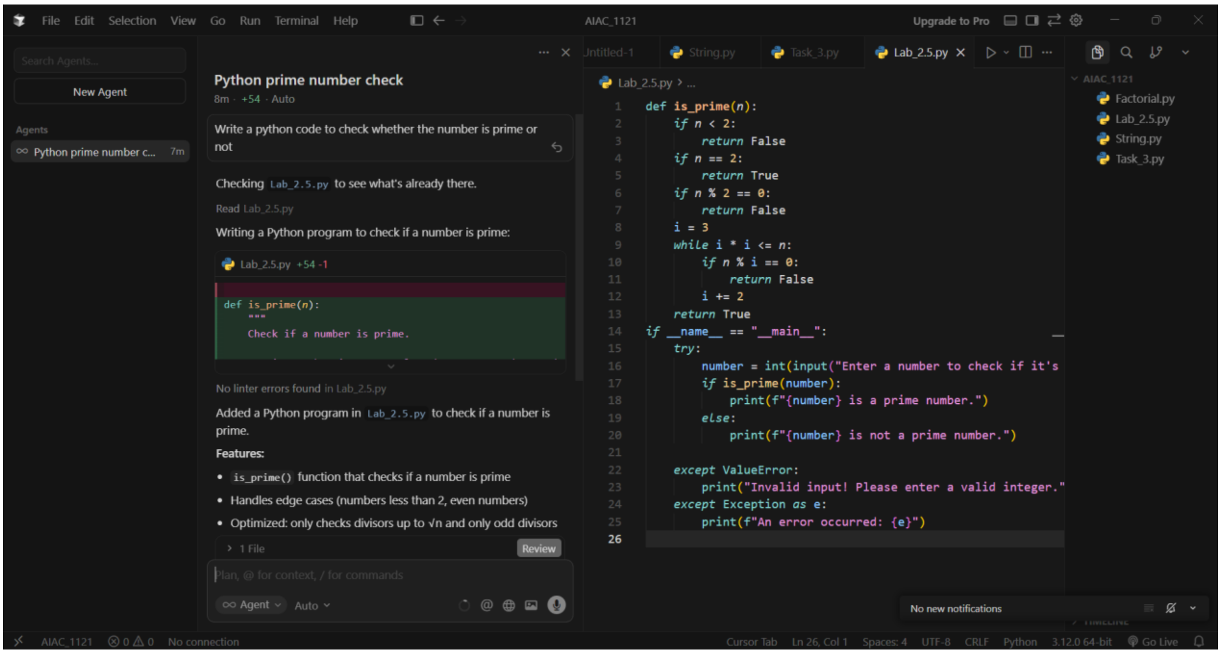
Explanation:

The calculate_area() function helps users find the area of circles, squares, or rectangles. It asks for the shape and necessary dimensions, like radius or side length, then calculates and prints the result. The function includes error handling to manage invalid inputs and ensures dimensions are non-negative, using Python's math module for calculations.

Task 3: Prompt Sensitivity Experiment.

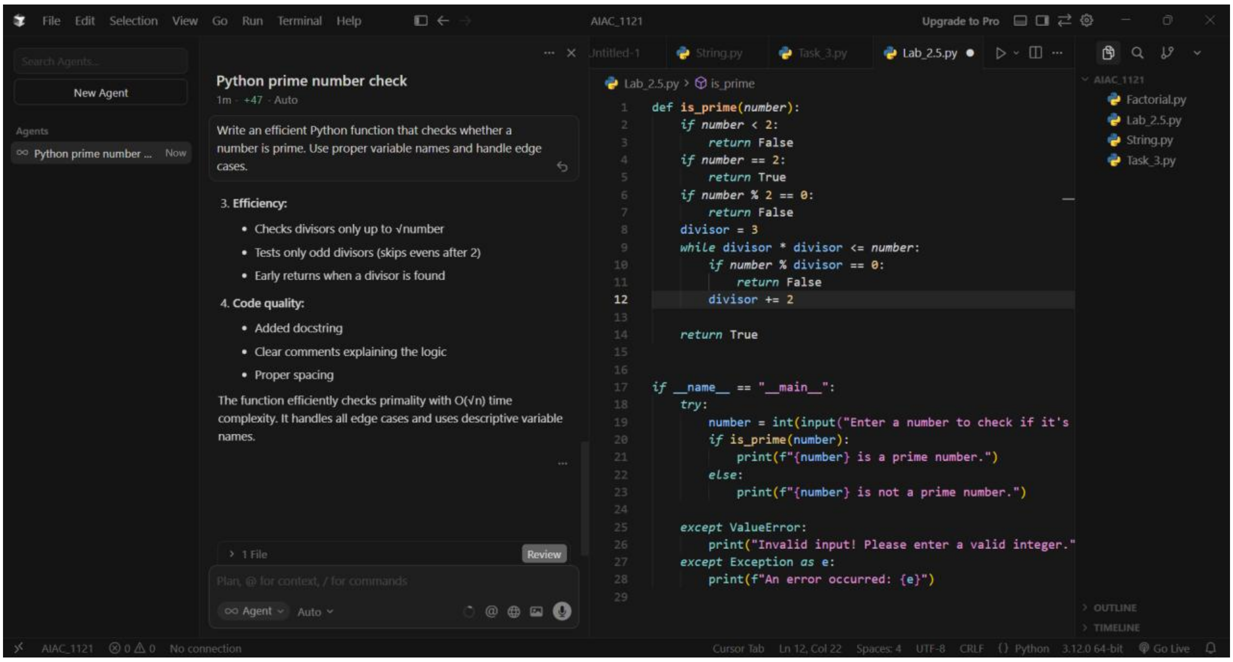
Prompt 1(Simple): Write a Python Code to check whether the given Number is Prime or Not

Code:



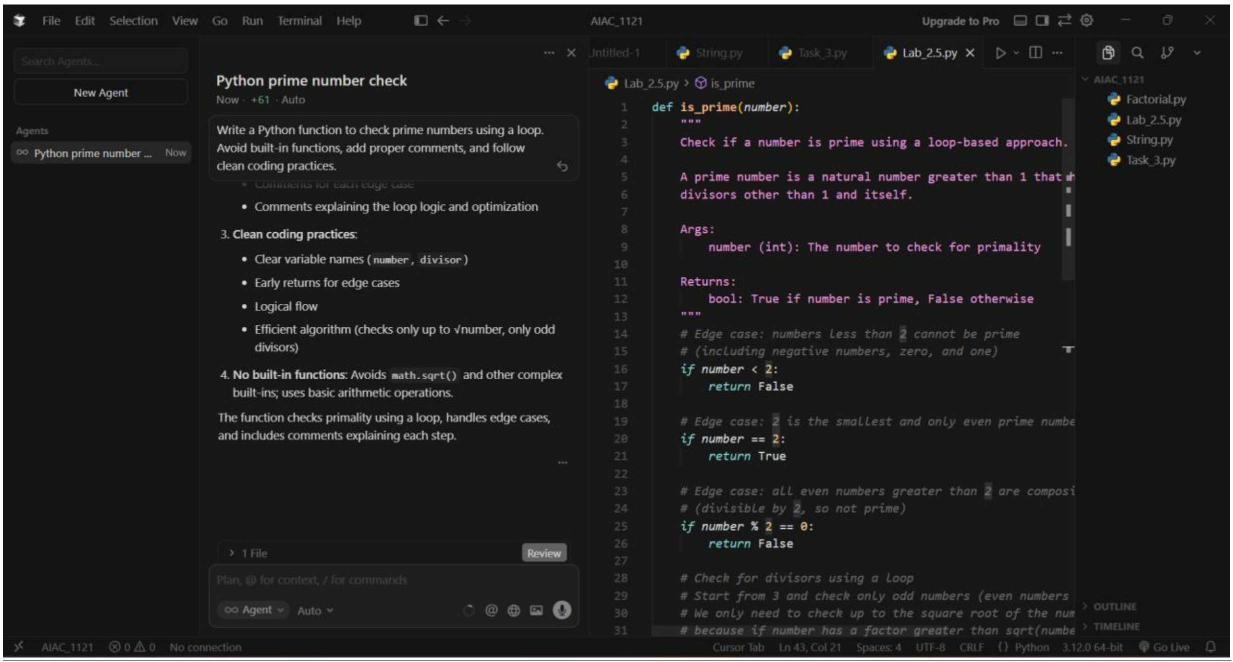
Prompt 2(Medium & Detailed): Write an efficient Python function that checks whether a number is prime. Use proper variable names and handle edge cases.

Code:



Prompt 3(Hard & More Detailed): Write a Python function to check prime numbers using a loop. Avoid built-in functions, add proper comments, and follow clean coding practices.

Code:



Task 4: Tool Comparison Reflection.

Prompt: Based on my experiments with Gemini, GitHub Copilot, and Cursor AI, compare these three tools in terms of:

- Ease of use
- Code quality
- Clarity of explanation
- Overall usefulness for students and developers
  Write a short reflection (8–10 lines).

Short Reflection:

Based on my experience, all three AI tools—Gemini, GitHub Copilot, and Cursor AI—are useful for AI-assisted coding, but in different ways. Gemini was very helpful for understanding concepts because it provided clear explanations along with code. GitHub Copilot generated concise and professional-level code, making it suitable for real-world development. However, Copilot gave limited explanations compared to Gemini. Cursor AI was useful for experimenting with different prompts and observing how code changes based on instructions. It also helped in refactoring and improving existing code effectively. In terms of usability, Gemini was the most beginner-friendly, while Copilot and Cursor were better suited for experienced programmers. Overall, Gemini is best for learning, Copilot for coding speed, and Cursor AI for refinement and experimentation.

THANK YOU !!