**LAB ASSIGNMENT - 03**

**Name** : B.Bharathkuamr

**Hall Ticket No** : 2303a51007

**Course Title** : AI Assistant Coding

**Experiment – Prompt Engineering Techniques**

**Task Description :**

Design and refine prompts using different prompting strategies to generate Python programs for basic computational problems.

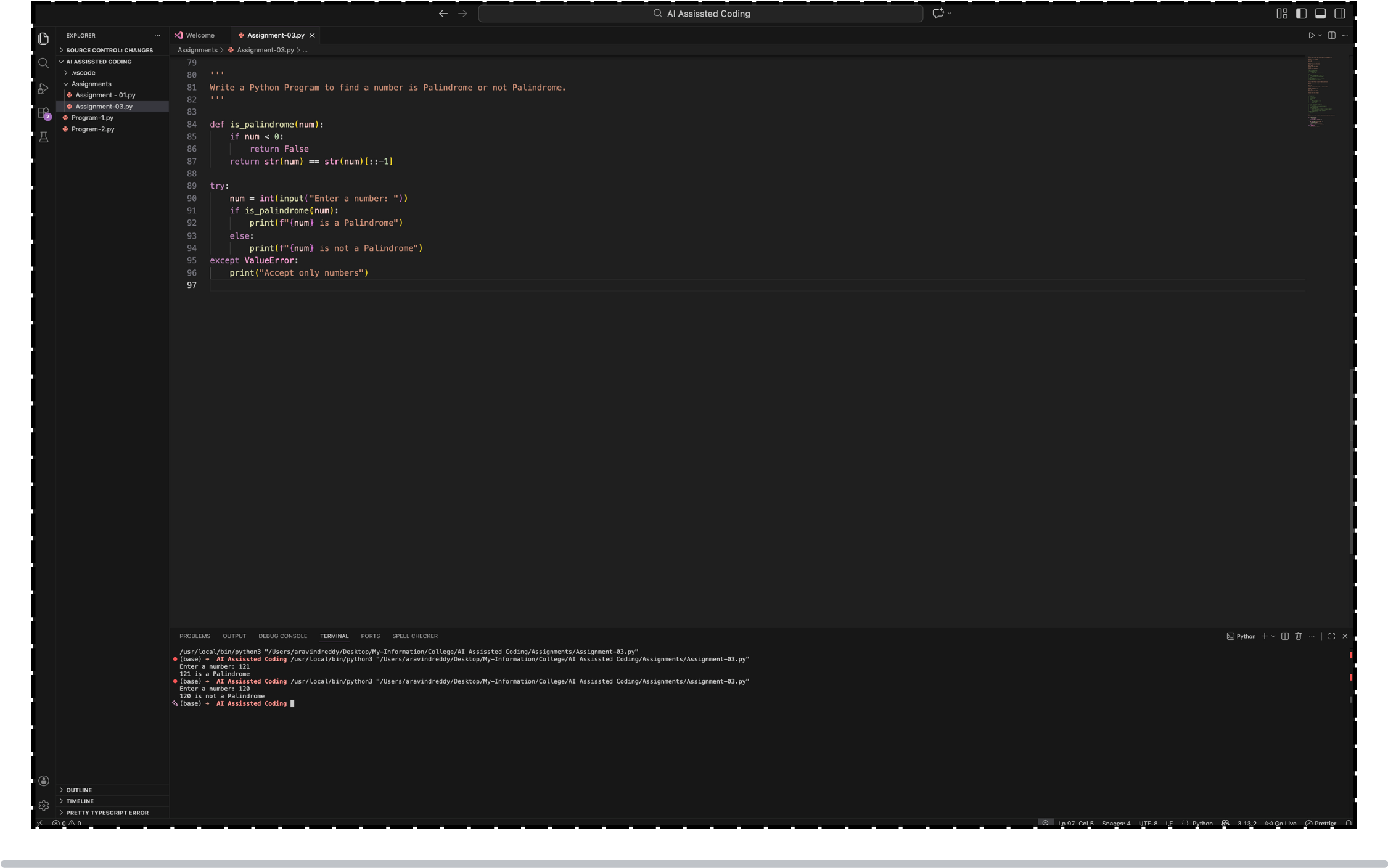
**Question 1 : Zero-Shot Prompting (Palindrome Number Program)** Write a zero-shot prompt (without providing any examples) to generate a Python function that checks whether a given number is a palindrome.

**Task:**

Record the AI-generated code.

Test the code with multiple inputs.

Identify any logical errors or missing edge-case handling.



**Question 2: One-Shot Prompting (Factorial Calculation)**

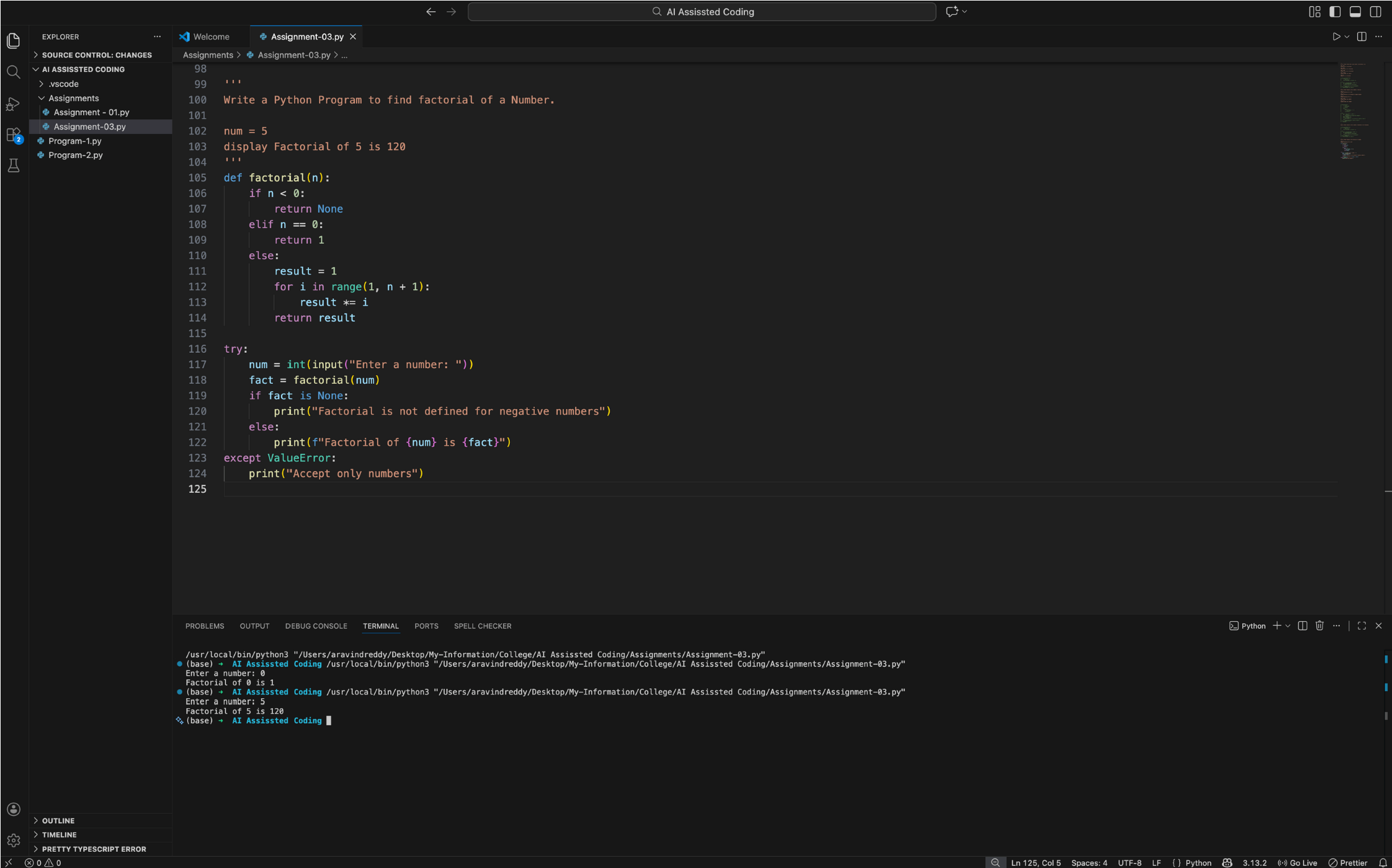
Write a one-shot prompt by providing one input-output example and ask the AI to generate a Python function to compute the factorial of a given number.

**Example:**

Input: 5 → Output: 120

**Task:**

* **Compare the generated code with a zero-shot solution.**
* **Examine improvements in clarity and correctness.**



**Question 3: Few-Shot Prompting (Armstrong Number Check)** Write a few-shot prompt by providing multiple input-output examples to guide the AI in generating a Python function to check whether a given number is an Armstrong number.

**Examples:**

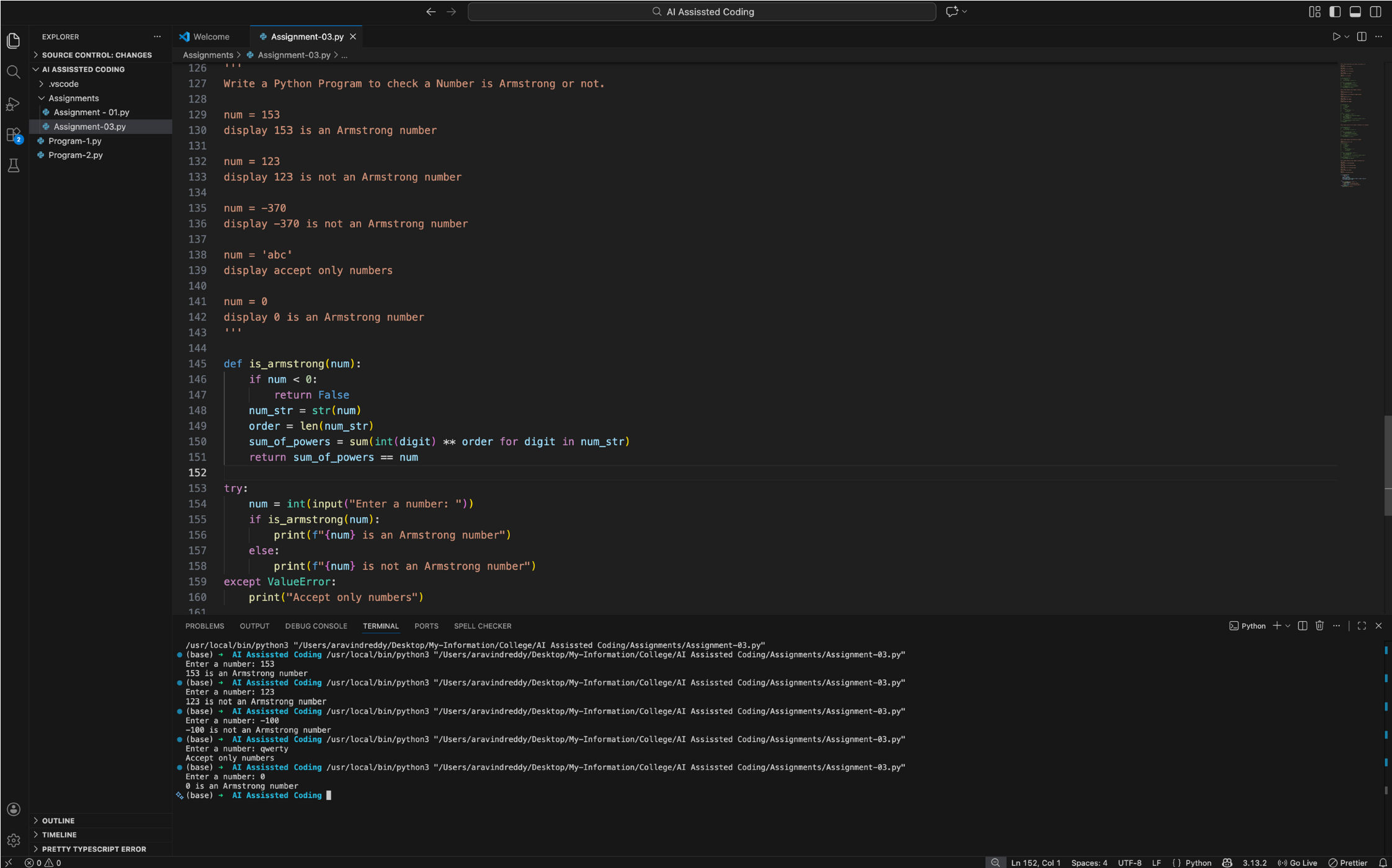
* Input: 153 → Output: Armstrong Number
* Input: 370 → Output: Armstrong Number• Input: 123 → Output: Not an Armstrong Number

**Task:**

* **Analyze how multiple examples influence code structure andaccuracy.**
* **Test the function with boundary values and invalid inputs.**

**(Optional Extension)**

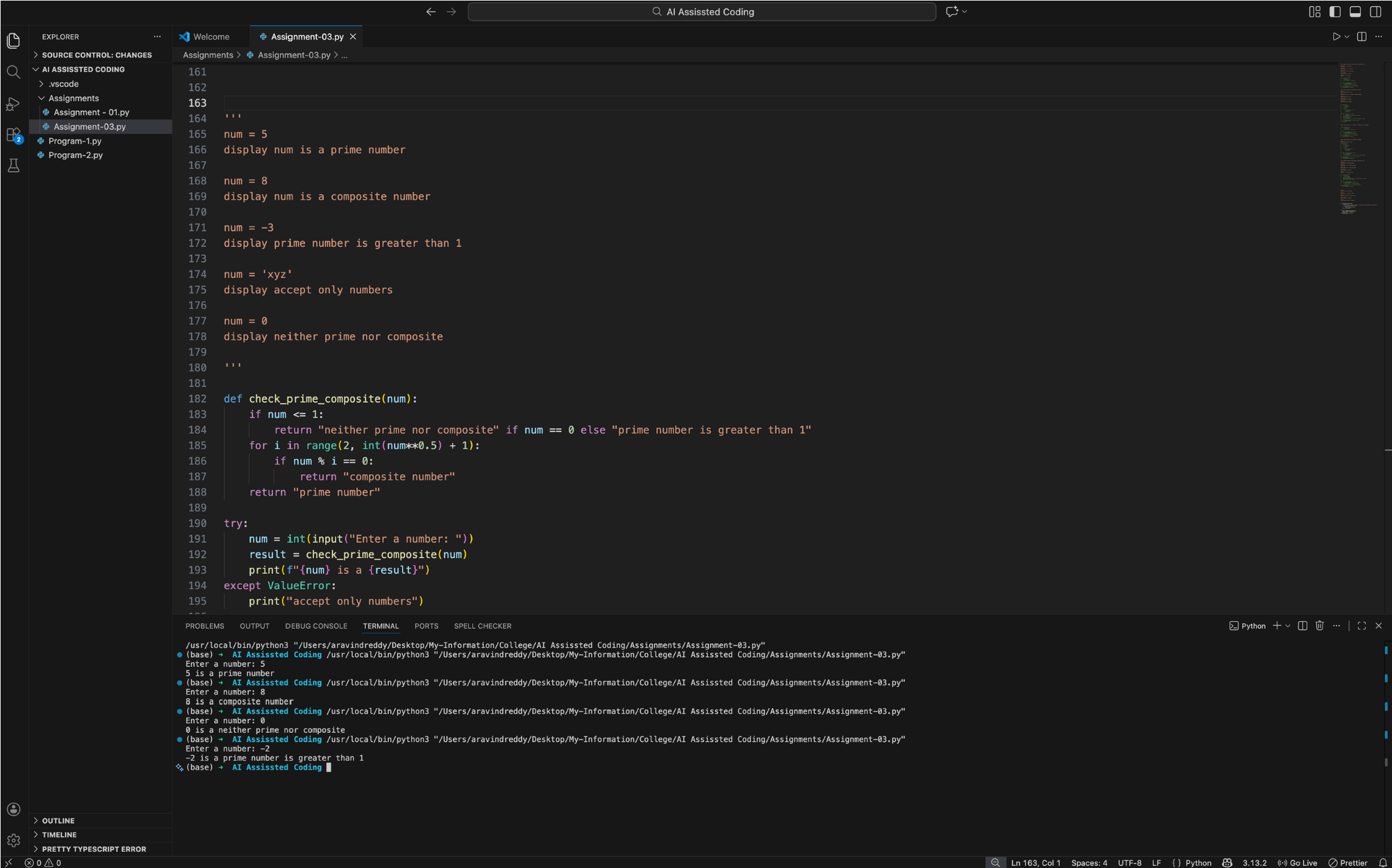
**Question 4: Context-Managed Prompting (Optimized Number Classification)** Design a context-managed prompt with clear instructions and constraints to generate an optimized Python program that classifies a number as prime, composite, or neither.



**Task:**

* Ensure proper input validation.
* Optimize the logic for efficiency.
* Compare the output with earlier prompting strategies.

**Question 5: Zero-Shot Prompting (Perfect Number Check)** Write a zero-shot prompt (without providing any examples) to generate a Python function that checks whether a given number is a perfect number.



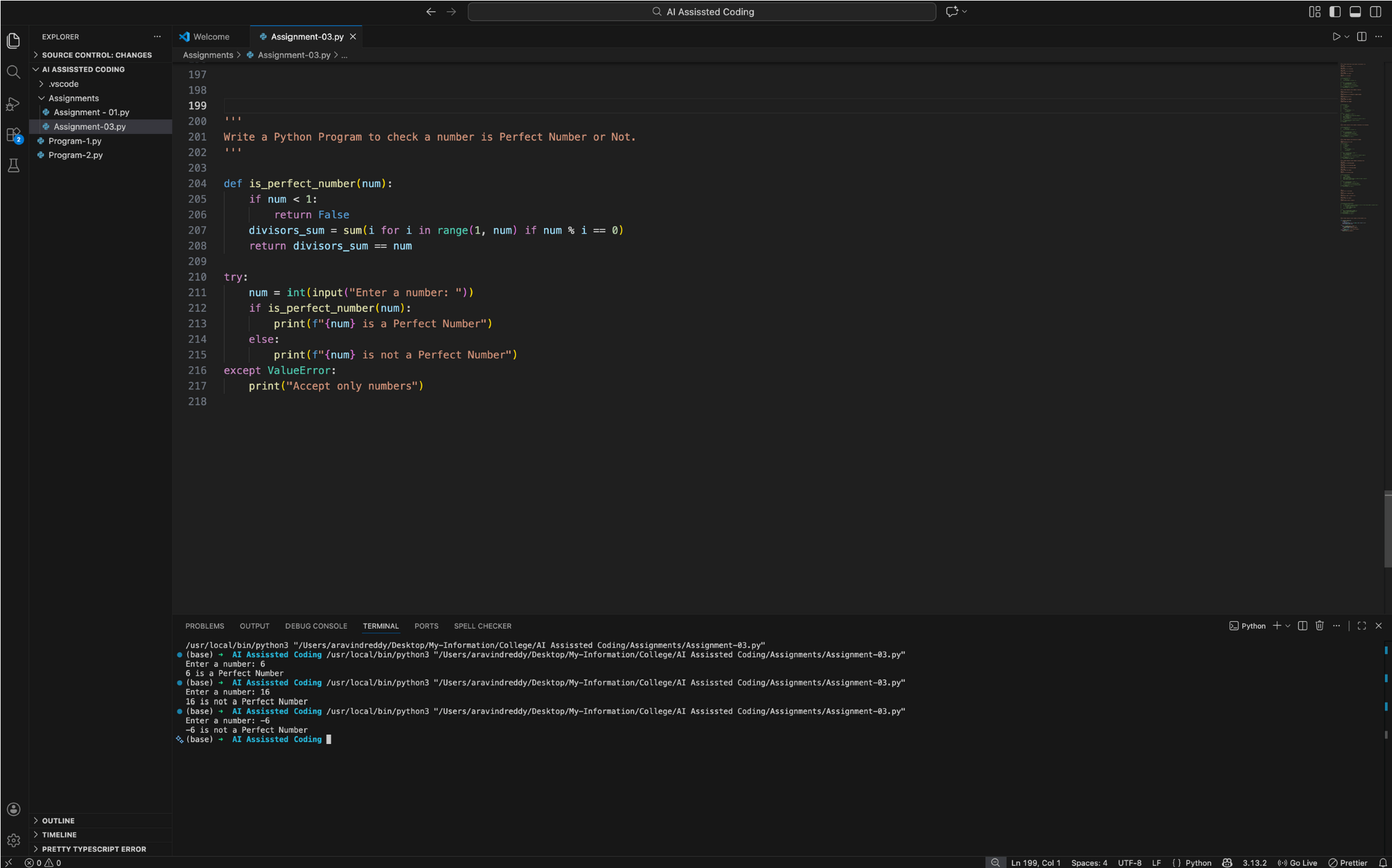
Task:

* Record the AI-generated code.
* Test the program with multiple inputs.
* Identify any missing conditions or inefficiencies in the logic.

**Question 6: Few-Shot Prompting (Even or Odd Classification with Validation)** Write a few-shot prompt by providing multiple input-output examples to guide the AI in generating a Python program that determines whether a given number is even or odd, including proper input validation.

Examples:

* Input: 8 → Output: Even



* Input: 15 → Output: Odd
* Input: 0 → Output: Even

Task:

* Analyze how examples improve input handling and output

clarity.

* Test the program with negative numbers and non-integer inputs

