ASSIGNMENT – 3.4

2303A51850

BATCH – 13

Task 1: Zero-shot Prompt – Fibonacci Series Generator

Task Description #1

• Without giving an example, write a single comment prompt asking GitHub Copilot to generate a Python function to print the first N Fibonacci numbers.

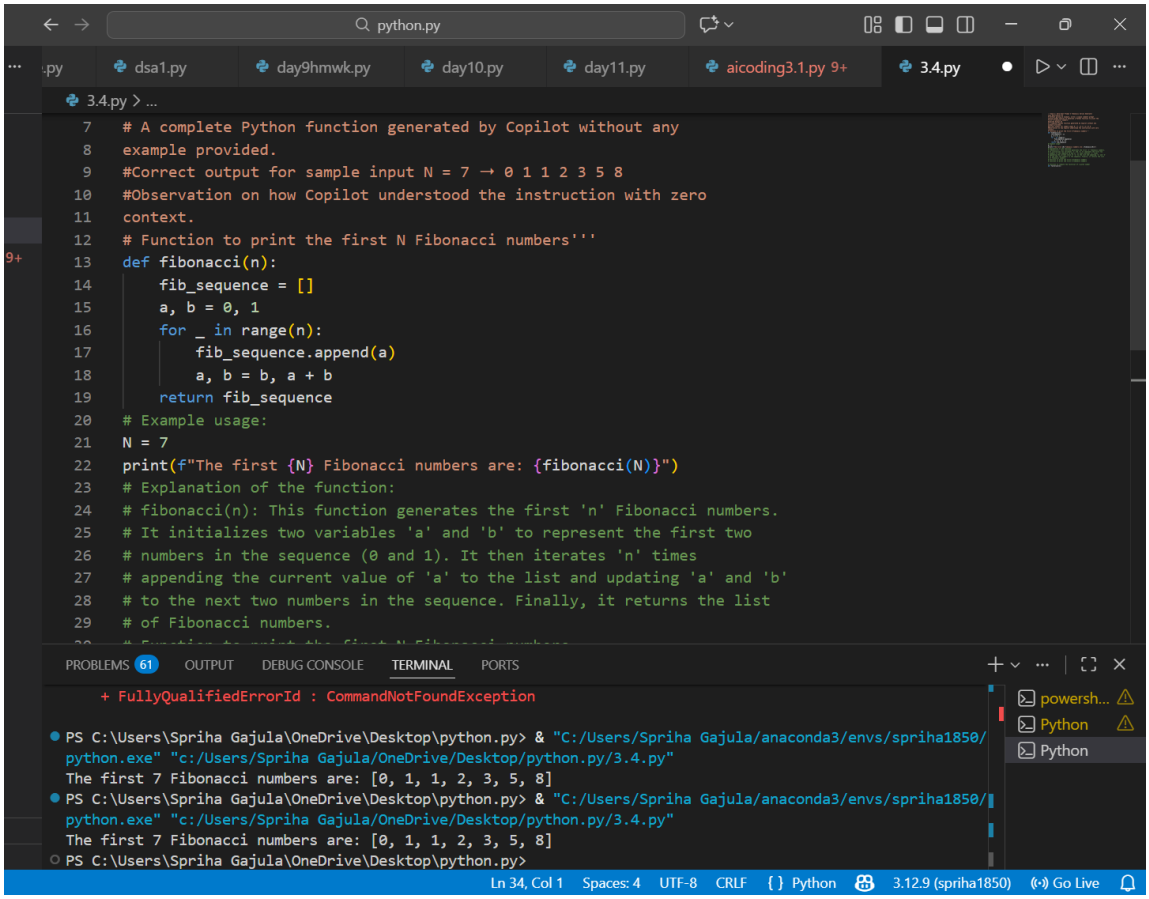
Expected Output #1

• A complete Python function generated by Copilot without any example provided.

• Correct output for sample input N = 7 → 0 1 1 2 3 5 8

• Observation on how Copilot understood the instruction with zero context.

CODE:



JUSTIFICATION:

fibonacci(n): This function generates the first 'n' Fibonacci numbers.

It initializes two variables 'a' and 'b' to represent the first two numbers in the sequence (0 and 1). It then iterates 'n' times appending the current value of 'a' to the list and updating 'a' and 'b' to the next two numbers in the sequence. Finally, it returns the list of Fibonacci numbers.

Task 2: One-shot Prompt – List Reversal Function

Task Description #2

• Write a comment prompt to reverse a list and provide one example below the comment to guide Copilot.

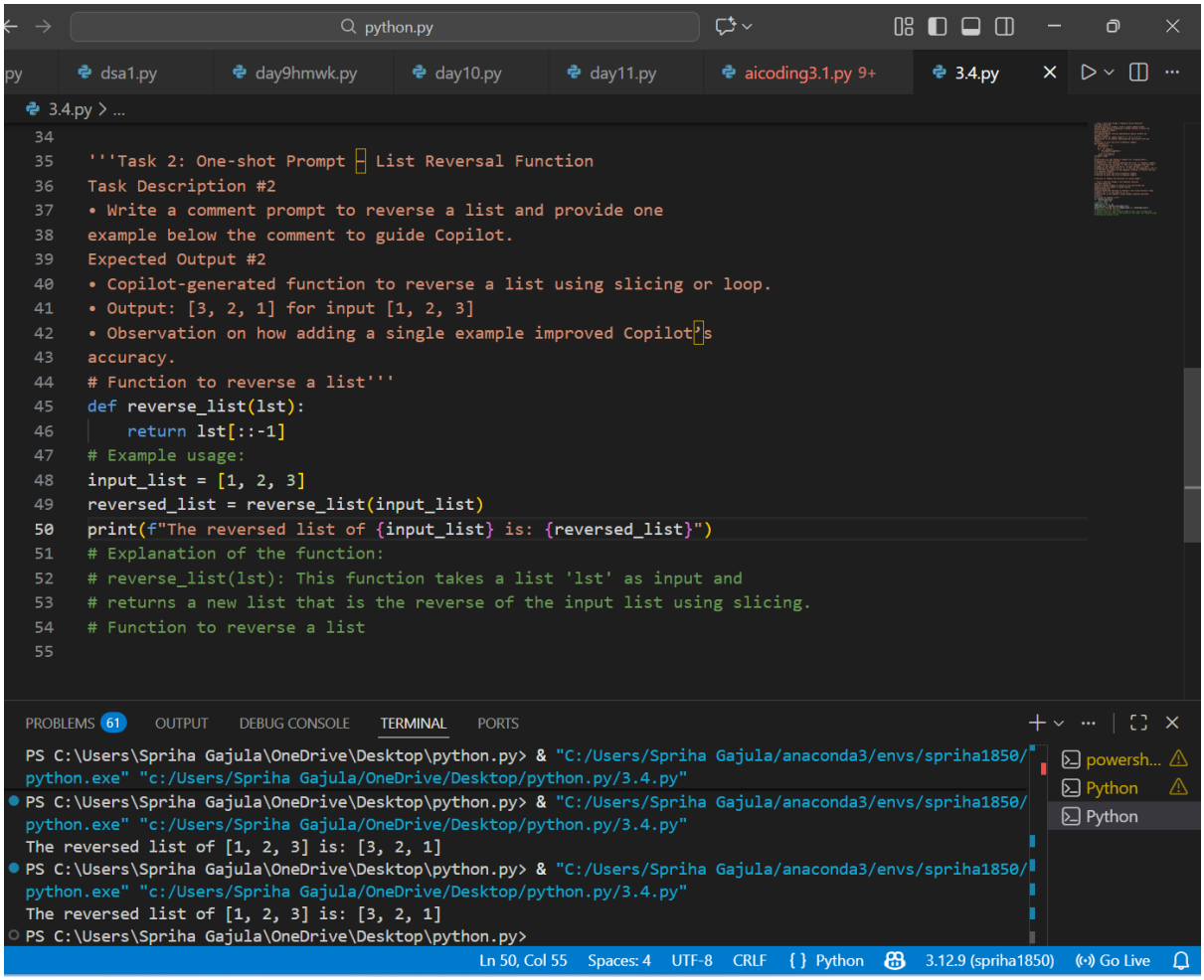
Expected Output #2

• Copilot-generated function to reverse a list using slicing or loop.

• Output: [3, 2, 1] for input [1, 2, 3]

• Observation on how adding a single example improved Copilot's accuracy.

CODE:



JUSTIFICATION:

reverse_list(lst): This function takes a list 'lst' as input and

returns a new list that is the reverse of the input list using slicing.

Function to reverse a list Function to compute the factorial of a given number

Task 3: Few-shot Prompt – String Pattern Matching

Task Description #3

• Write a comment with 2–3 examples to help Copilot understand

how to check if a string starts with a capital letter and ends with a

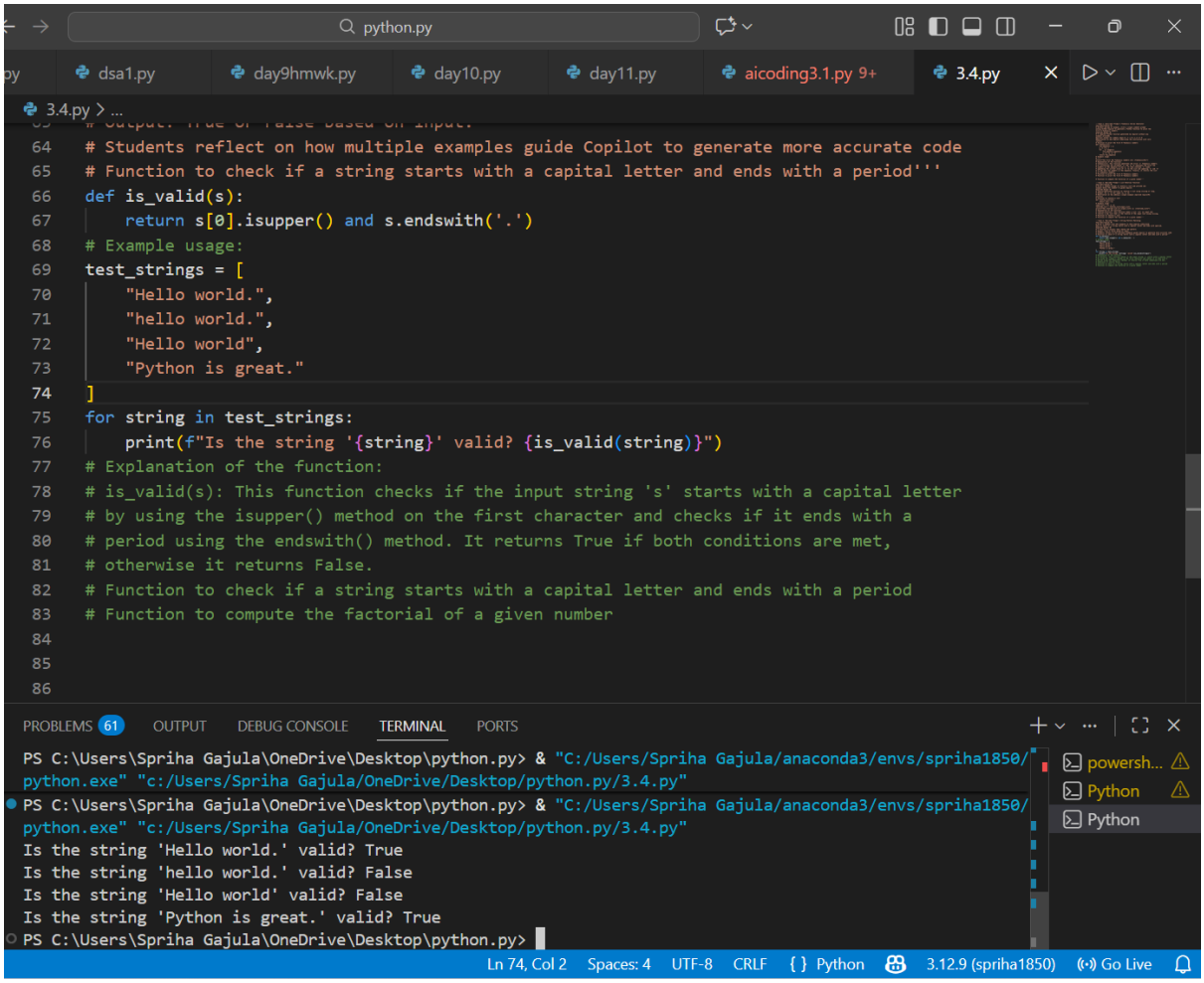
period.

Expected Output #3

• A function is_valid() that checks the pattern.

• Output: True or False based on input.

• Students reflect on how multiple examples guide Copilot to generate more accurate code.



JUSTIFICATION :

is_valid(s): This function checks if the input string 's' starts with a capital letter by using the isupper() method on the first character and checks if it ends with a period using the endswith() method. It returns True if both conditions are met, otherwise it returns False.

Function to check if a string starts with a capital letter and ends with a period

Function to compute the factorial of a given number

Task 4: Zero-shot vs Few-shot – Email Validator

Task Description #4

• First, prompt Copilot to write an email validation function using zero-shot (just the task in comment).

• Then, rewrite the prompt using few-shot examples.

Expected Output #4

• Compare both outputs:

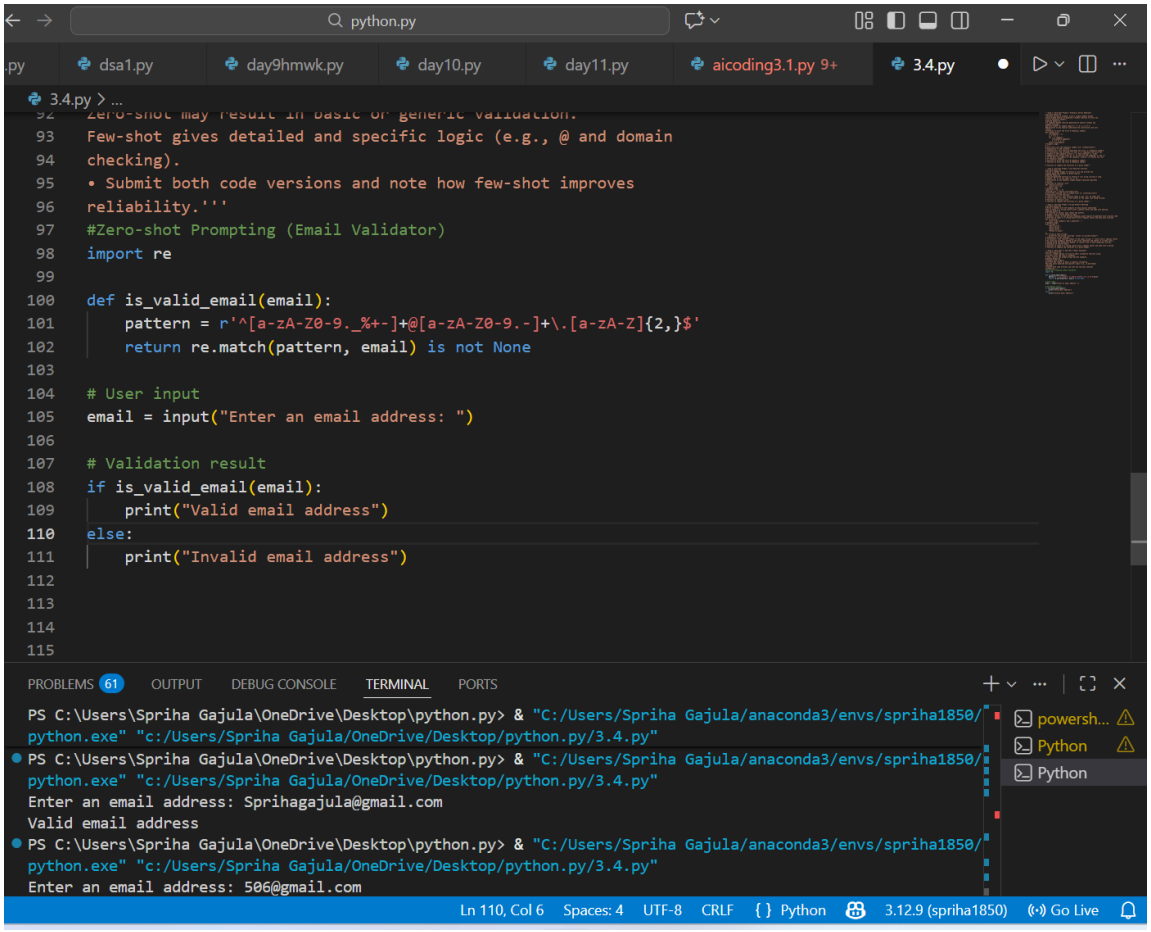
Zero-shot may result in basic or generic validation.

Few-shot gives detailed and specific logic (e.g., @ and domain checking).

• Submit both code versions and note how few-shot improves reliability.

CODE:

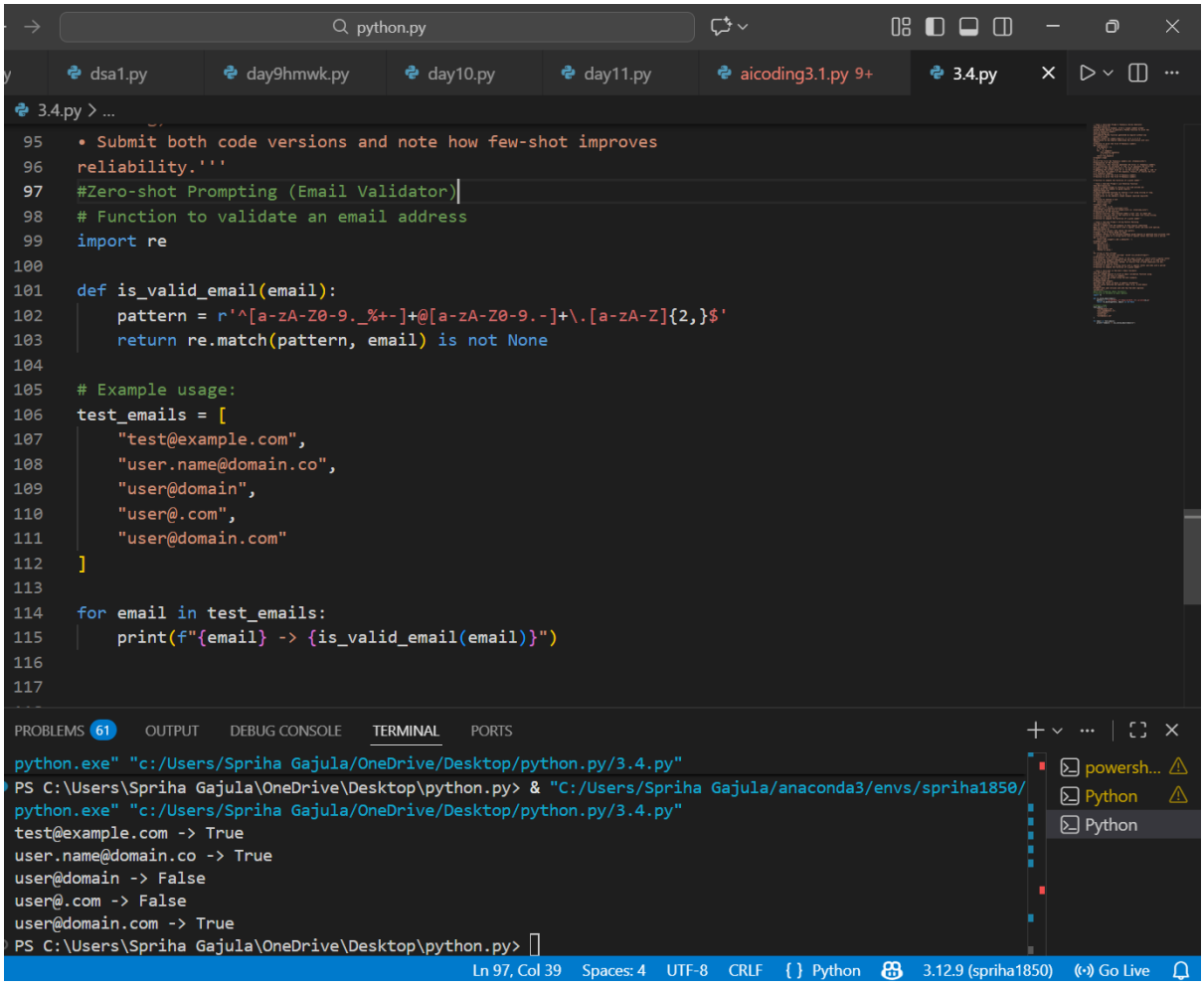
Zero-shot Prompting (Email Validator)



JUSTIFICATION :

This program uses regular expressions to validate the structure of an email address.
It checks for the presence of valid characters, an @ symbol, and a proper domain format.
User input makes the program flexible and reusable for different email addresses.
The function returns a Boolean value, ensuring clear and efficient validation.

Few-shot Prompting (Email Validator)



JUSTIFICATION :

This program checks whether an email address is valid using a regular expression.
The is_valid_email() function defines a pattern that allows letters, numbers, and special characters before @, followed by a valid domain name and extension.

A list of sample email addresses is provided to test different cases, including valid and invalid formats.
Each email is passed to the function, and the result (True or False) is printed to show whether the email is valid.

Task 5: Prompt Tuning – Summing Digits of a Number

Task Description #5

• Experiment with 2 different prompt styles to generate a function

that returns the sum of digits of a number.

Style 1: Generic task prompt

Style 2: Task + Input/Output example

Expected Output #5

• Two versions of the sum_of_digits() function.

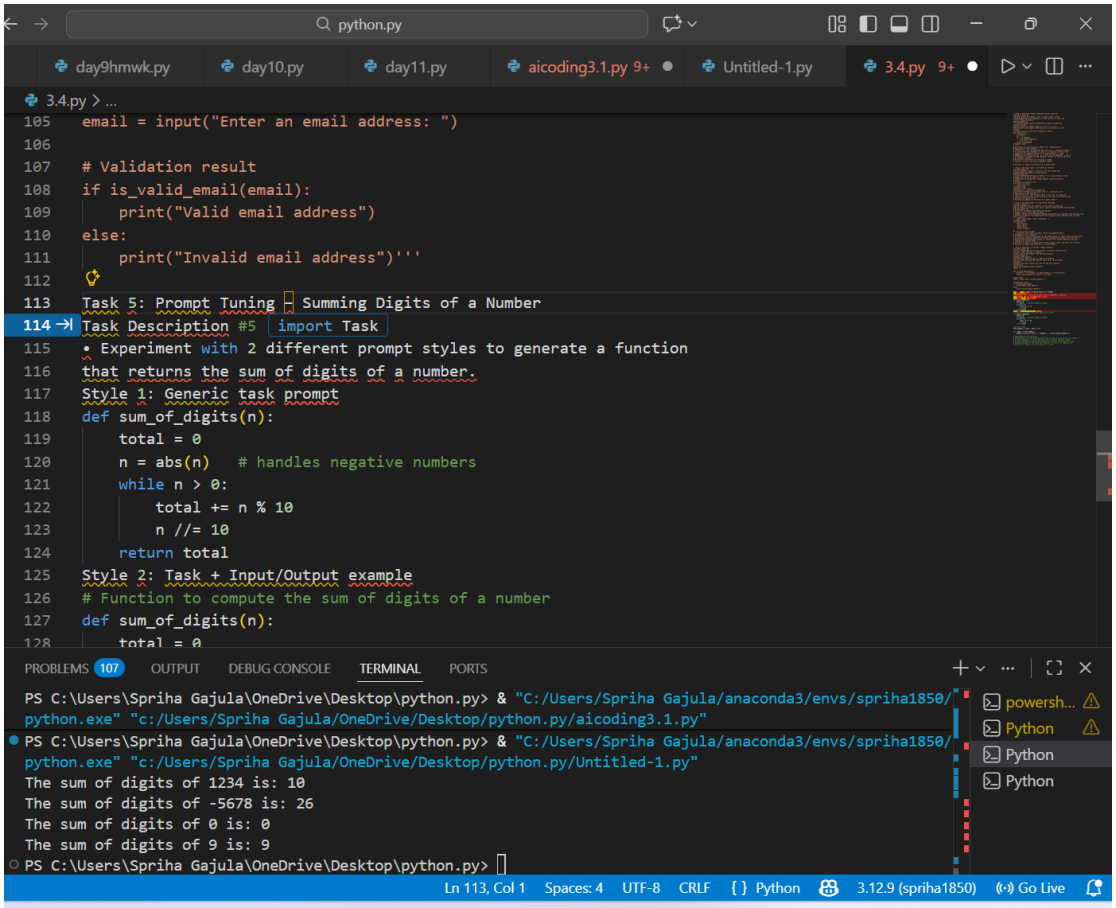
• Example Output: sum_of_digits(123) → 6

• Short analysis: which prompt produced cleaner or more

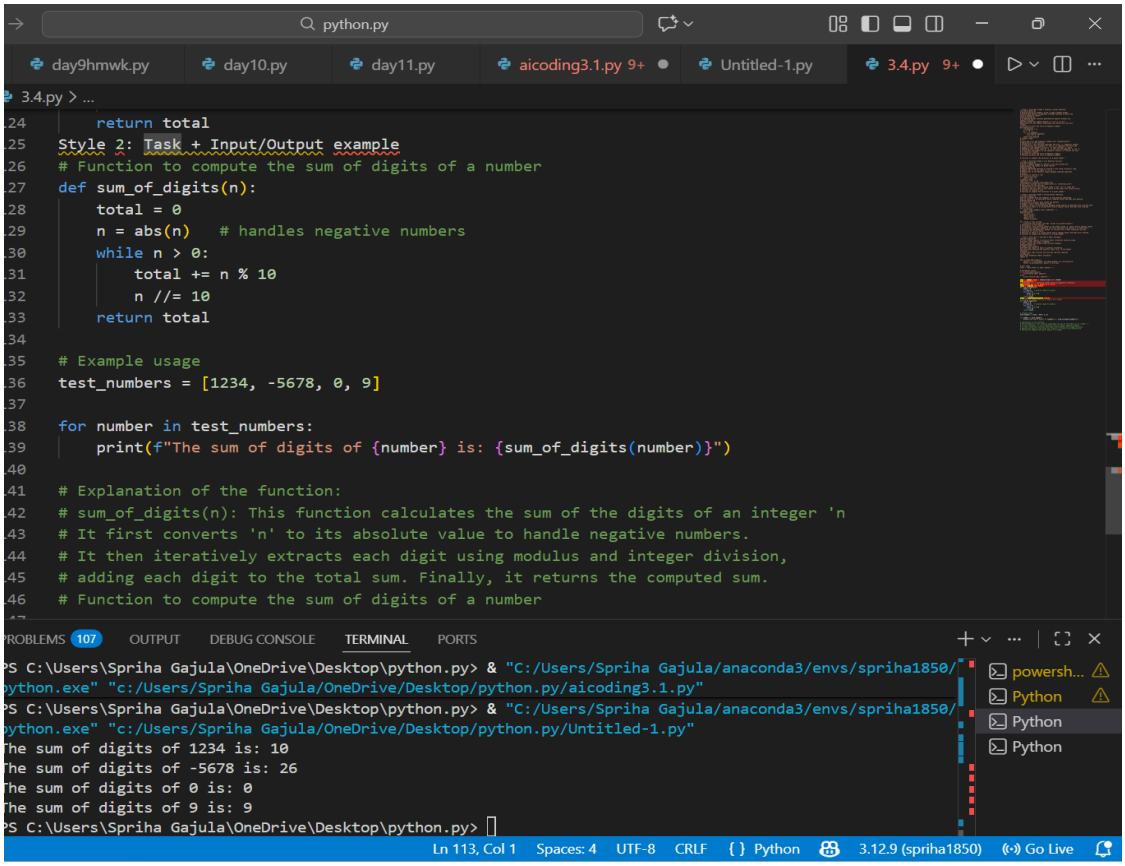
optimized code and why

CODE:

Style 1: Generic task prompt



Style 2: Task + Input/Output example



JUSTIFICATION:

The sum_of_digits function calculates the total of all digits in a number. First, it takes the absolute value so that negative numbers are handled correctly. It then looks at each digit one by one, starting from the last digit, and adds it to a running total. After all digits have been added, the function gives back the total sum. Finally, the code tests this function with different numbers and prints the results.