SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: M.Tech. and MCA		**Assignment Type: Lab**	**AcademicYear:**2025-2026
Course Coordinator Name		Venkataramana Veeramsetty	
Course Code		**Course Title**	AI Assisted Problem Solving Using Python
Year/Sem	I/I	**Regulation**	R24
Date and Day of Assignment	Week3 - Monday	**Time(s)**	
Duration	2 Hours	**Applicable to Batches**	M.Tech. and MCA
AssignmentNumber:4.3(Present assignment number)/**24**(Total number of assignments)			

Q.No.	Question	*Expected Time to complete*
1	Lab 4: Advanced Prompt Engineering – Zero-shot, One-shot, and Few-shot Techniques **Lab Objectives:** ● To explore and apply different levels of prompt examples in AI-assisted code generation. ● To understand how zero-shot, one-shot, and few-shot prompting affect AI output quality. ● To evaluate the impact of context richness and example quantity on AI performance. ● To build awareness of prompt strategy effectiveness for different problem types. **Lab Outcomes (LOs):** After completing this lab, students will be able to: ● Use zero-shot prompting to instruct AI with minimal context. ● Use one-shot prompting with a single example to guide AI code generation. ● Apply few-shot prompting using multiple examples to improve AI	Week3 - Monday

responses.

- Compare AI outputs across the three prompting strategies.

Task Description#1

- Zero-shot: Prompt AI to write a function that checks whether a given year is a leap year.

Expected Output#1

- AI-generated function with no examples provided

Task Description#2

- One-shot: Give one input-output example to guide AI in writing a function that converts centimeters to inches.

Expected Output#2

- Function with correct conversion logic

Task Description#3

- Few-shot: Provide 2–3 examples to generate a function that formats full names as "Last, First".

Expected Output#3

- Well-structured function respecting the examples

Task Description#4

- Compare zero-shot and few-shot prompts for writing a function that counts the number of vowels in a string.

Expected Output#4

- Functional output and comparative reflection

Task Description#5

- Use few-shot prompting to generate a function that reads a .txt file and returns the number of lines.

Expected Output#5

- Working file-processing function with AI-guided logic

Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots

Evaluation Criteria:

	Criteria	Max Marks		
	Zero Shot (Task #1)	2.5		
	One Shot (Task#2)	2.5		
	Few Shot (Task#3 & Task #5)	2.5		
	Comparison (Task#4)	2.5		
	Total	**10 Marks**		

Task Description#1

- Zero-shot: Prompt AI to write a function that checks whether a given year is a leap year.

PROMPT: Write a program that takes a year from the user and prints whether it is a leap year or not.

```
C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 4 > TASK 1 > TASK 1.py > ...
def is_leap(year: int) -> bool:
    """ Return True if year is a leap year, otherwise False.
    Leap year rule:
      - divisible by 4 and not divisible by 100, OR
      - divisible by 400 """
    return (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0)

def get_year_from_user(prompt: str = "Enter a year: ") -> int:
    """Prompt the user until a valid integer year is entered and return it."""
    while True:
        s = input(prompt).strip()
        try:
            year = int(s)
            return year
        except ValueError:
            print("Invalid input. Please enter a valid integer year.")


def main():
    year = get_year_from_user()
    if is_leap(year):
        print(f"{year} is a leap year.")
    else:
        print(f"{year} is not a leap year.")


if __name__ == "__main__":
    main()
```

Expected Output#1

- AI-generated function with no examples provided

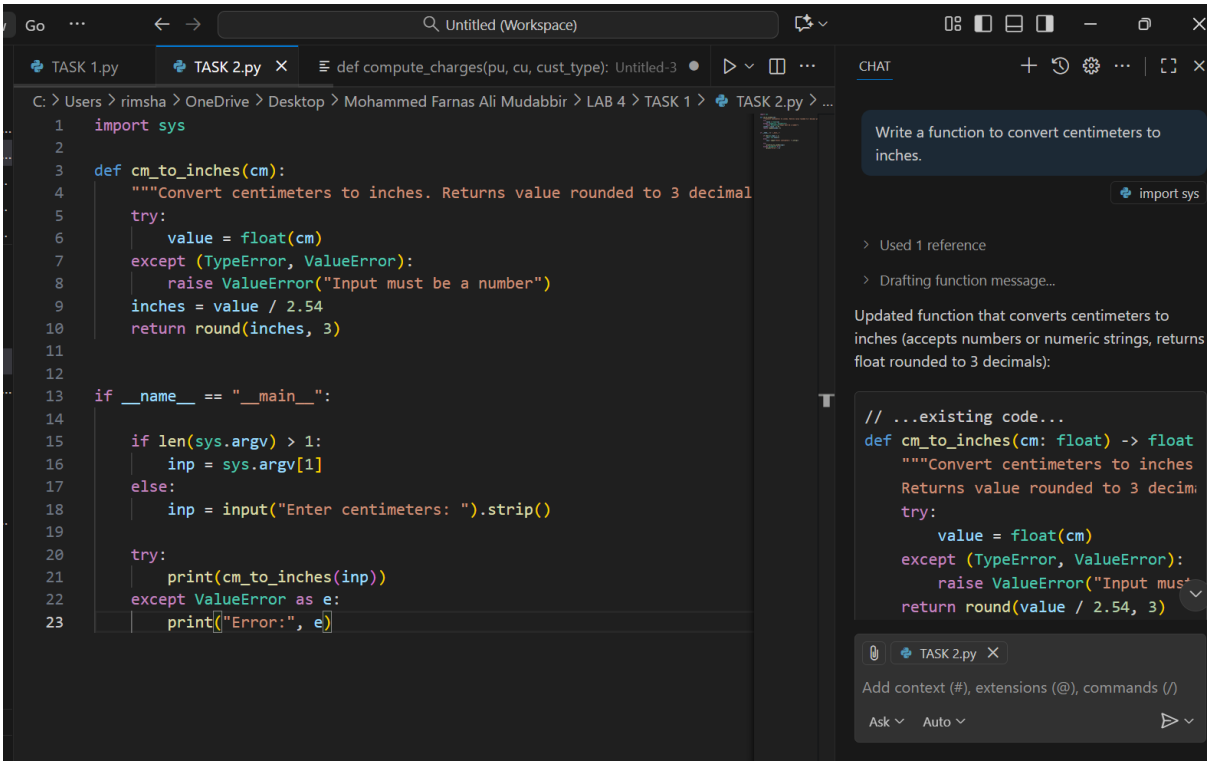
Practical output:

```
Enter a year: 2012
2012 is a leap year.
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#2

- One-shot: Give one input-output example to guide AI in writing a function that converts centimeters to inches.

PROMPT: Write a function that converts a value in centimeters to inches**.**



Expected Output#2

- Function with correct conversion logic

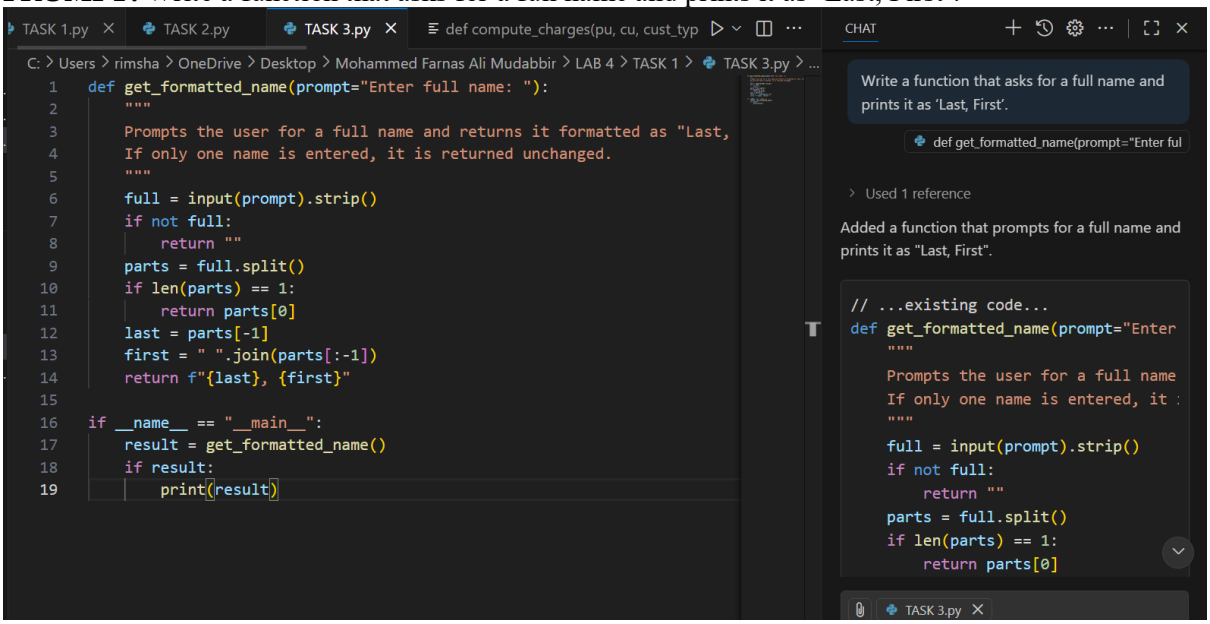
Practical output:

```
Enter centimeters: 20
7.874
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#3

Few-shot: Provide 2–3 examples to generate a function that formats full names as “Last, First”.

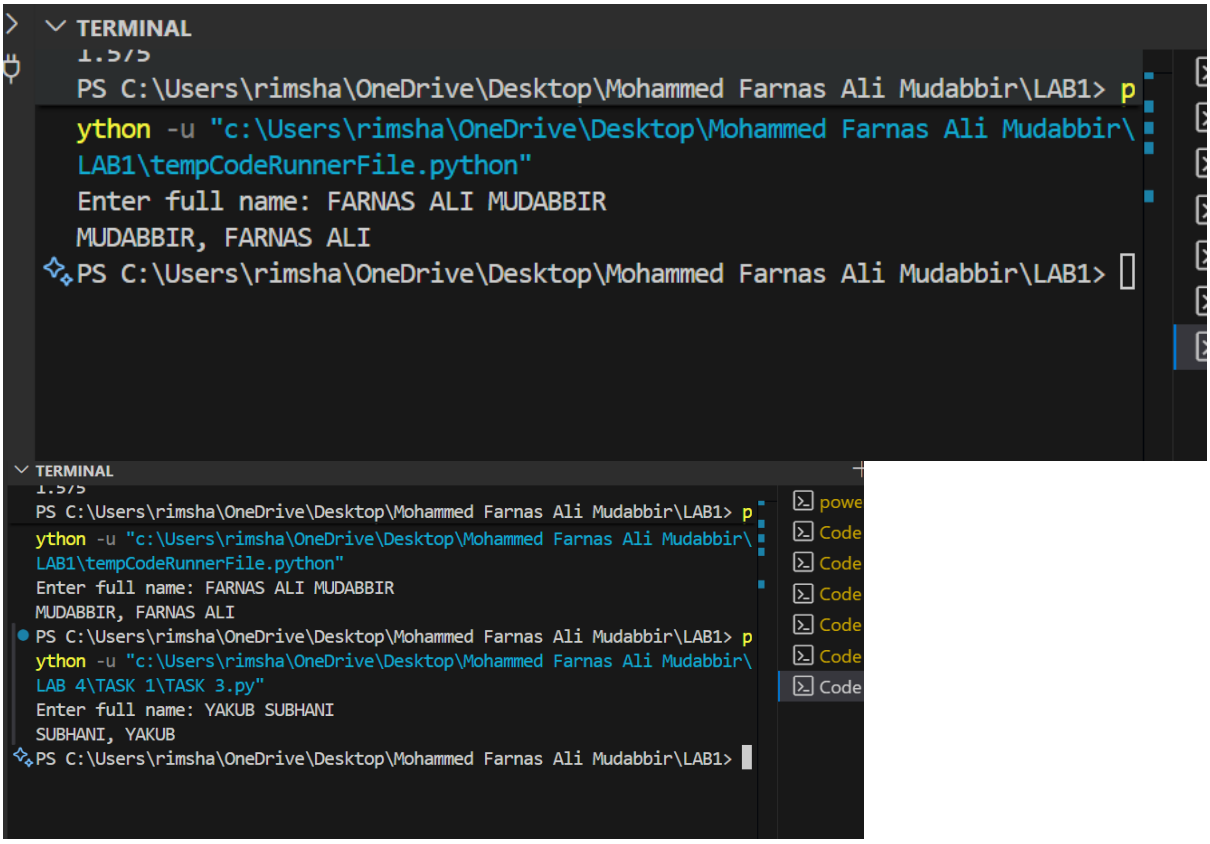
PROMPT: Write a function that asks for a full name and prints it as ‘Last, First’.



Expected Output#3

- Well-structured function respecting the examples

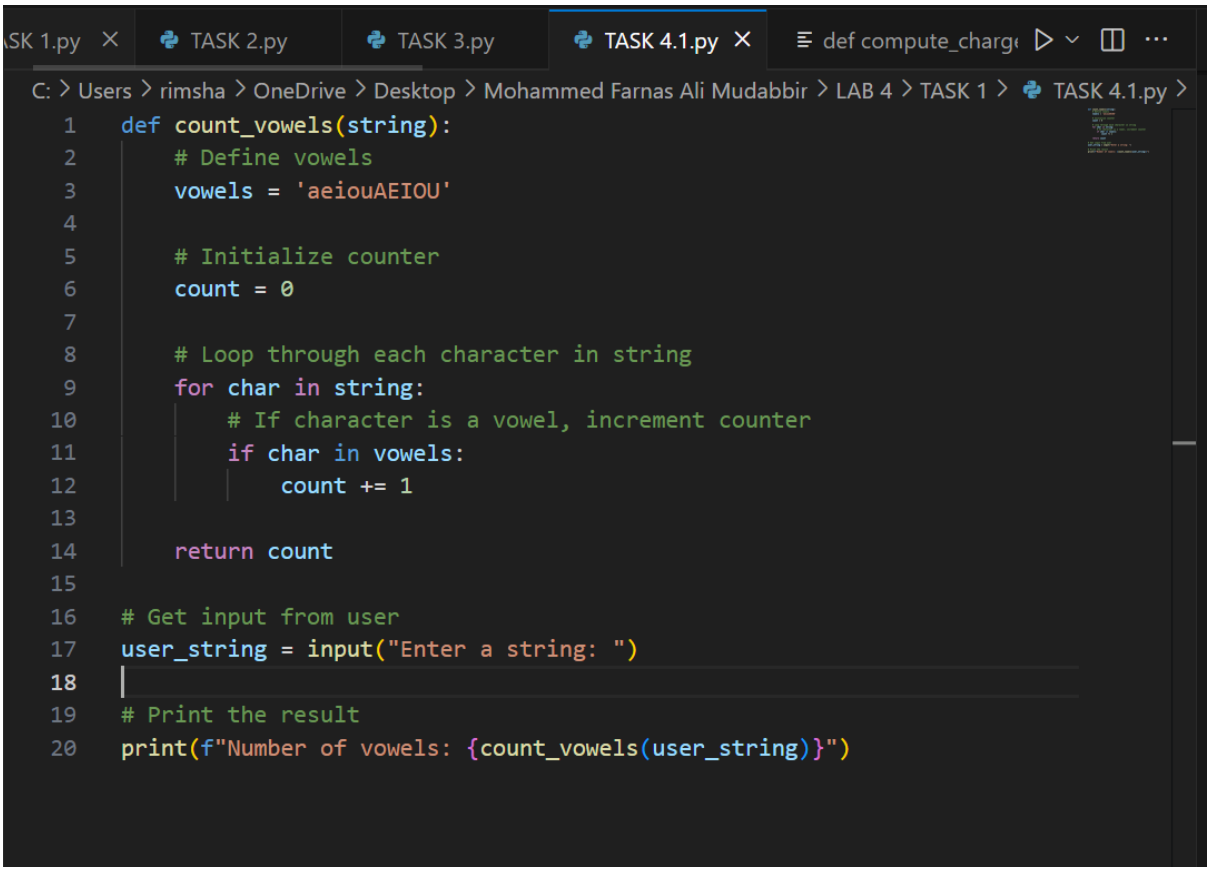
Practical output:



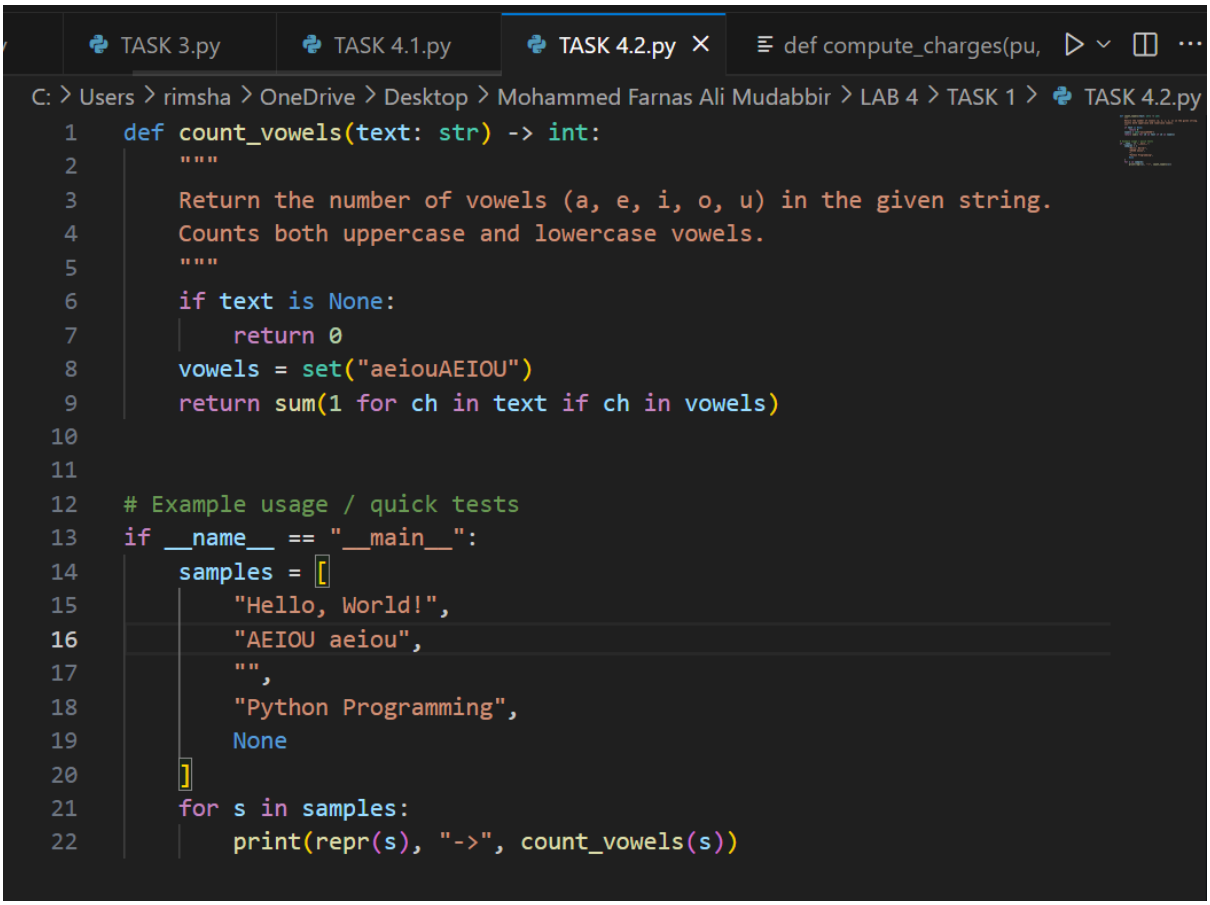
Task Description#4

Compare zero-shot and few-shot prompts for writing a function that counts the number of vowels in a string.

ZERO-SHOT PROMPT: Write a function that takes a string as input and returns the number of vowels in it.



FEW-SHOT PROMPT: Write a user input function that reads a string and prints how many vowels it has.



Difference Between Zero-Shot and Few-Shot Prompts

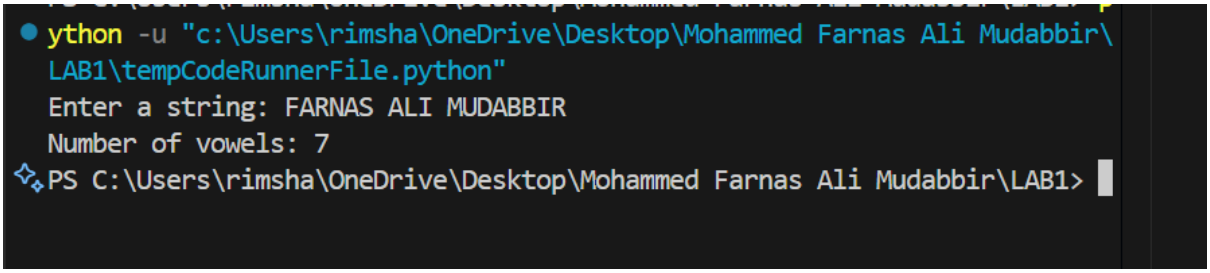
- Zero-Shot Prompt:
  The model is given only the instruction (no examples).
  Example*:* Write a function to count vowels.
- Few-Shot Prompt:
  The model is given one or more examples along with the instruction.
  Example*:* Showing sample input-output before asking to write the function.

Expected Output#4

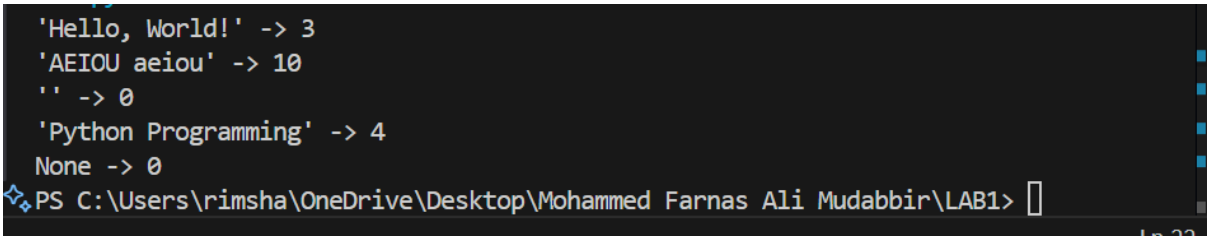
- Functional output and comparative reflection

Practical output:

ZERO-SHOT:

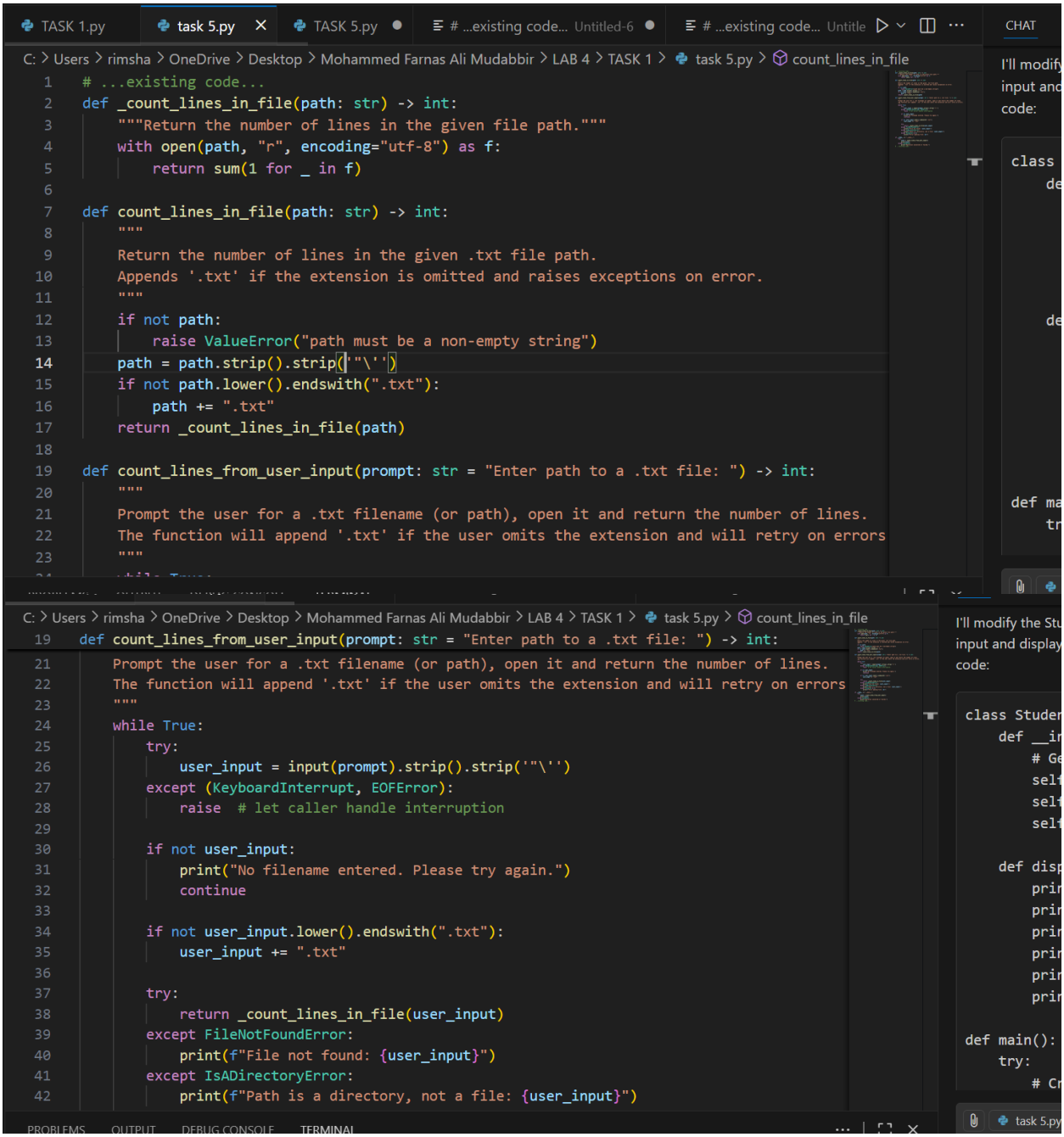


FEW-SHOT:



Task Description#5

- Use few-shot prompting to generate a function that reads a .txt file and returns the number of lines.



TASK 1.py | task 5.py | TASK 5.py | # ...existing code... Untitled-6 | # ...existing code... Untitle

C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 4 > TASK 1 > task 5.py > count_lines_in_file

```
19  def count_lines_from_user_input(prompt: str = "Enter path to a .txt file: ") -> int:
41              except IsADirectoryError:
42                  print(f"Path is a directory, not a file: {user_input}")
43              except Exception as e:
44                  print(f"Error opening file: {e}")
45
46  if __name__ == "__main__":
47      try:
48          count = count_lines_from_user_input()
49          print(count)
50      except Exception:
51          print("Operation cancelled or failed.")
52  # ...existing code...
```

Expected output:

```
Enter path to a .txt file: "C:\Users\rimsha\OneDrive\Desktop\SAMPLE_txt.txt"
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Enter path to a .txt file: "C:\Users\rimsha\OneDrive\Desktop\SAMPLE_txt.txt"
Enter path to a .txt file: "C:\Users\rimsha\OneDrive\Desktop\SAMPLE_txt.txt"
7
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```