| **SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | | **DEPARTMENT OF COMPUTER SCIENCE ENGINEERING** | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ProgramName:**B. Tech | | | | **Assignment Type: Lab** | | | **AcademicYear:**2025-2026 | | |
| **CourseCoordinatorName** | | | | Venkataramana Veeramsetty | | | | | |
| **Instructor(s)Name** | | | | | Dr. V. Venkataramana (Co-ordinator) | | --- | | Dr. T. Sampath Kumar | | Dr. Pramoda Patro | | Dr. Brij Kishor Tiwari | | Dr.J.Ravichander | | Dr. Mohammand Ali Shaik | | Dr. Anirodh Kumar | | Mr. S.Naresh Kumar | | Dr. RAJESH VELPULA | | Mr. Kundhan Kumar | | Ms. Ch.Rajitha | | Mr. M Prakash | | Mr. B.Raju | | Intern 1 (Dharma teja) | | Intern 2 (Sai Prasad) | | Intern 3 (Sowmya) | | NS\_2 ( Mounika) | | | | | | |
| **CourseCode** | | | 24CS002PC215 | **CourseTitle** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | | Week2 - Wednesday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:4.3**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***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 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) | 0.5 | | One Shot (Task#2) | 0.5 | | Few Shot (Task#3 & Task #5) | 1.0 | | Comparison (Task#4) | 0.5 | | **Total** | **2.5 Marks** | | | | | | | Week2 - Wednesday |  |

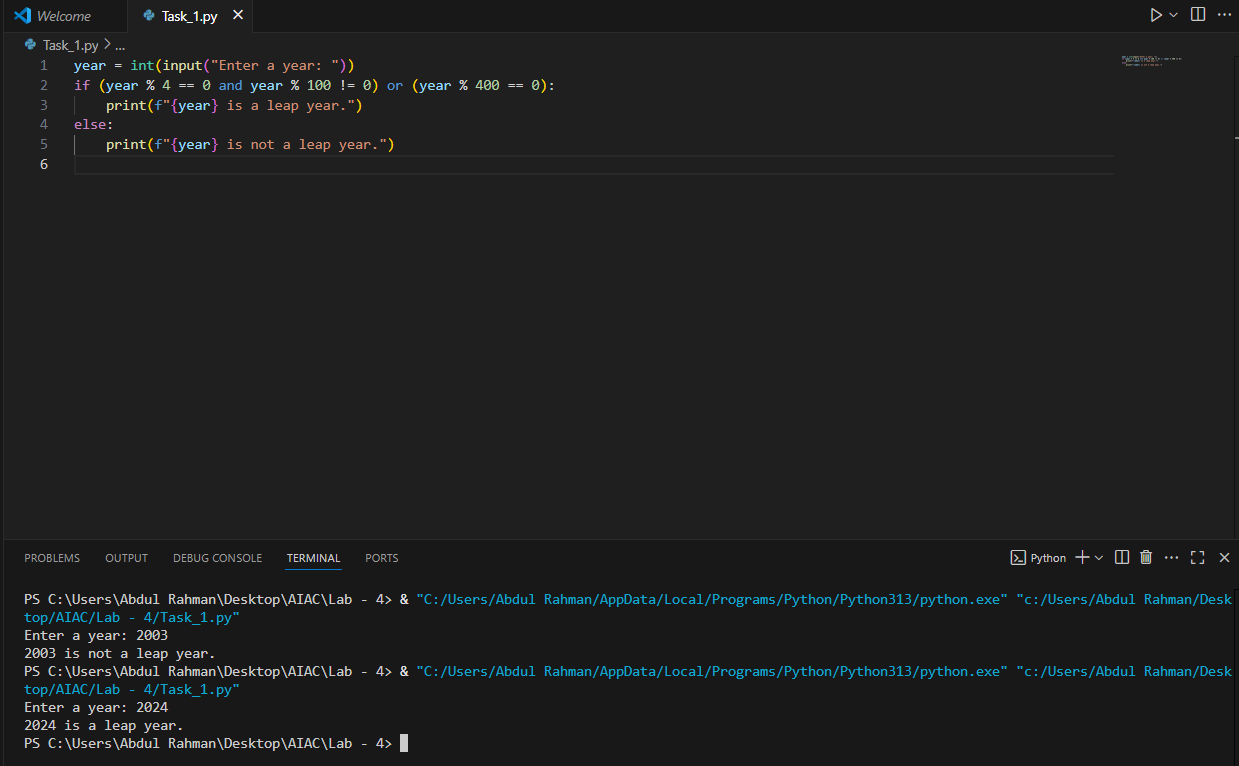
**Task Description#1**

* 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

Prompt: Write a simple python programme which checks whether a given year is leap year or not , and takes input from console.





**Task Description#2**

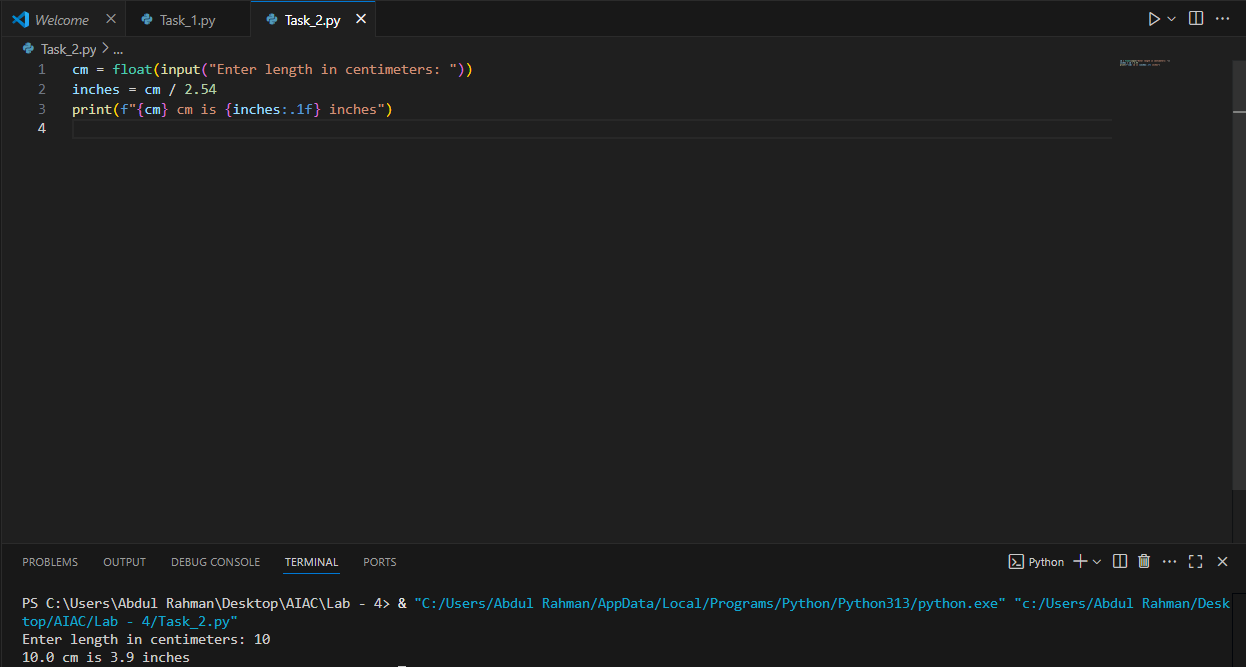
* One-shot: Give one input-output example to guide AI in writing a function that converts centimeters to inches.

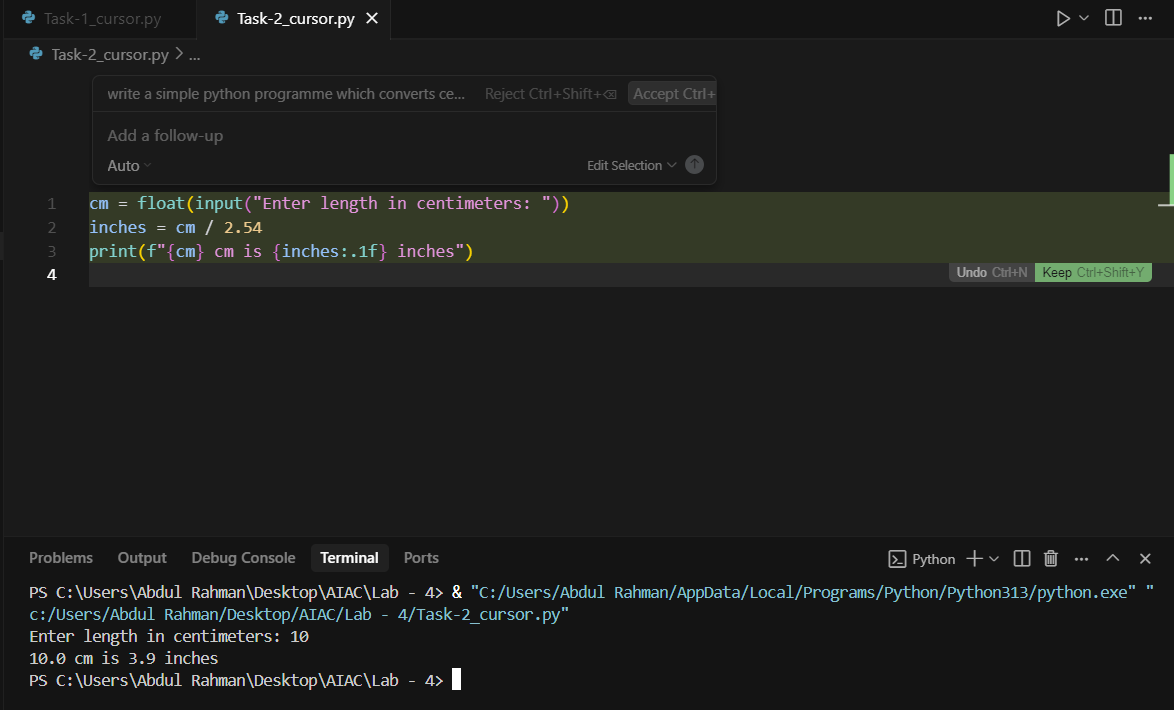
**Expected Output#2**

* Function with correct conversion logic

Prompt : Write a simple python programme which converts cms into inches , takes input from the console .

Example : 10 cms is 3.9 inches.





**Task Description#3**

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

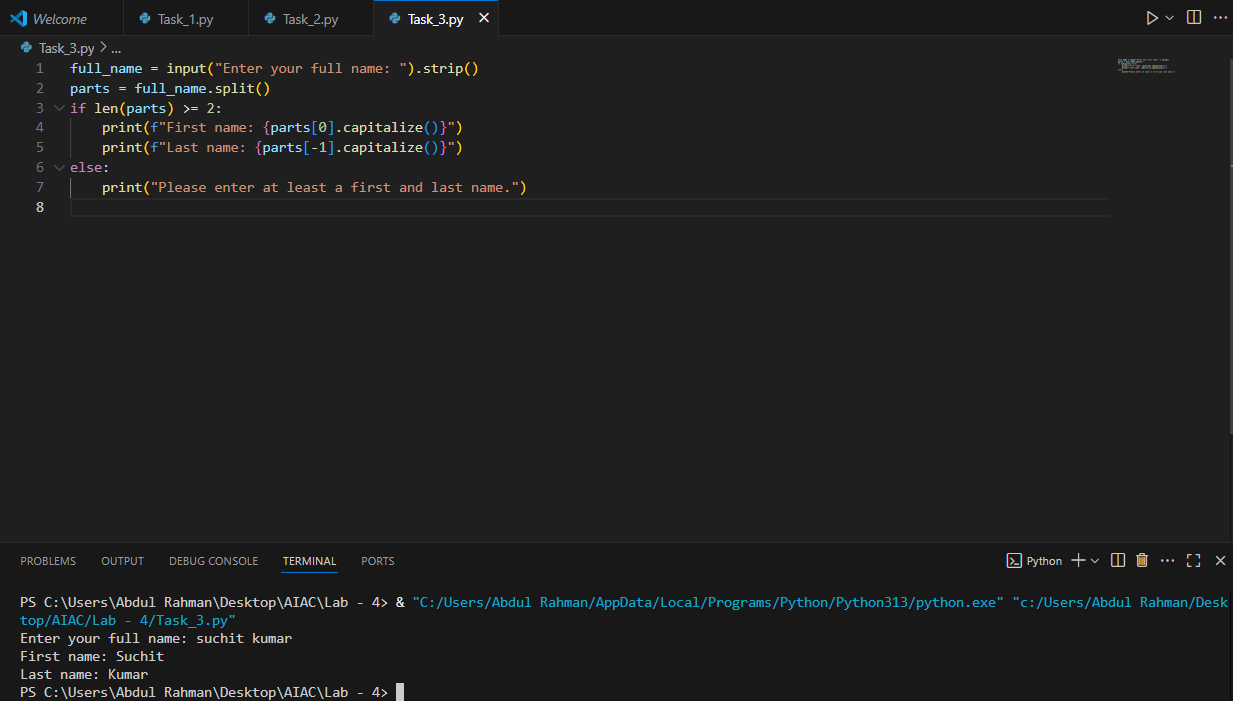
**Expected Output#3**

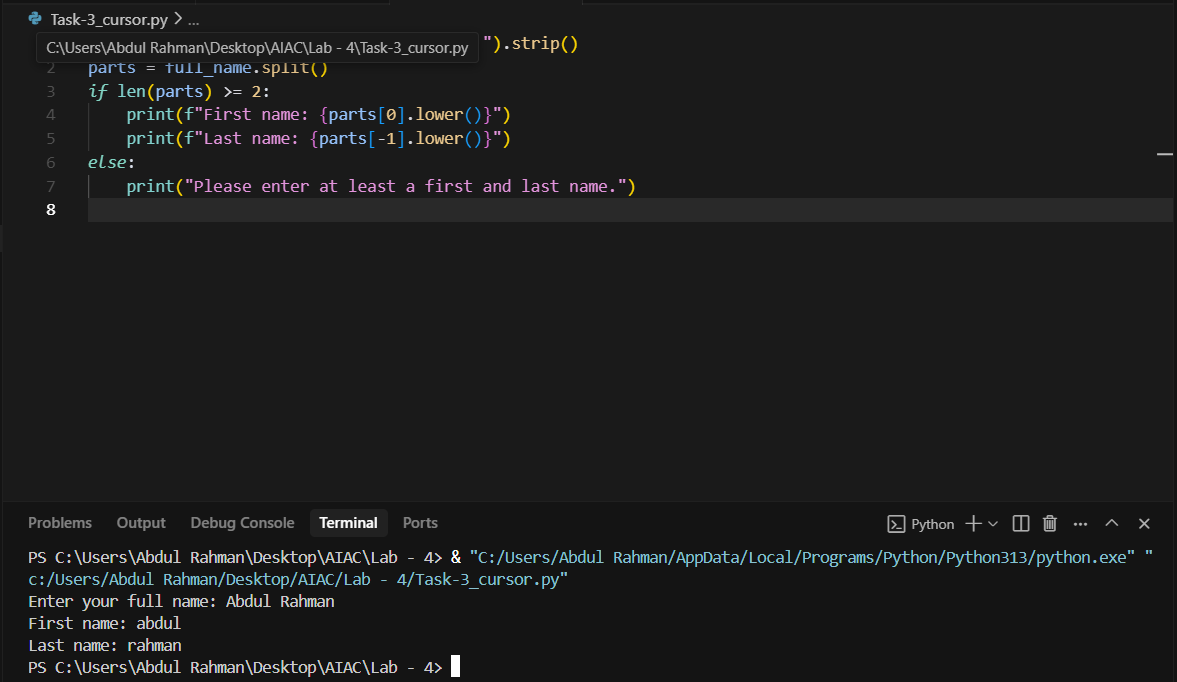
* Well-structured function respecting the examples

Prompt : Write a python programme which divides the full name into two parts first name and last name , take input from console.

Example : Abdul Rahman–first name : Abdul , last name : Rahman.

Example : Albert Einstein –first name : Albert , last name : Einstein.





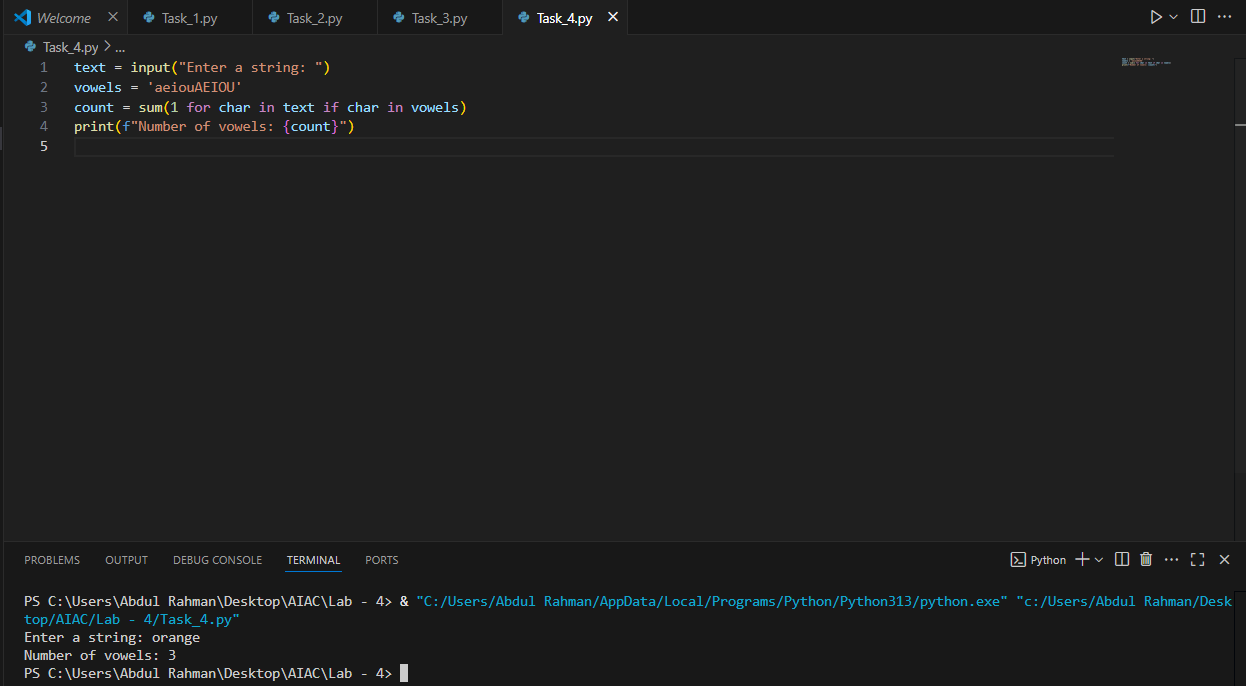
**Task Description#4**

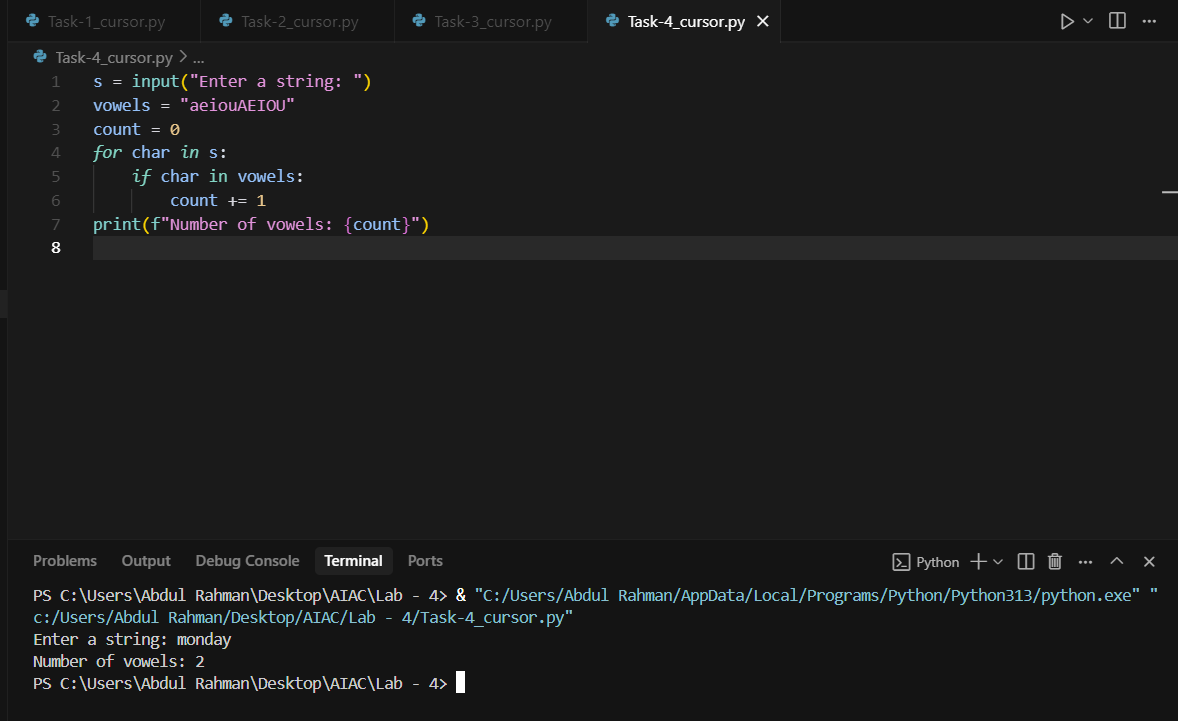
* Compare zero-shot and few-shot prompts for writing a function that counts the number of vowels in a string.

**Expected Output#4**

* Functional output and comparative reflection

Prompt : Write a simple python programme which counts number of vowels in string , taking input string from console.



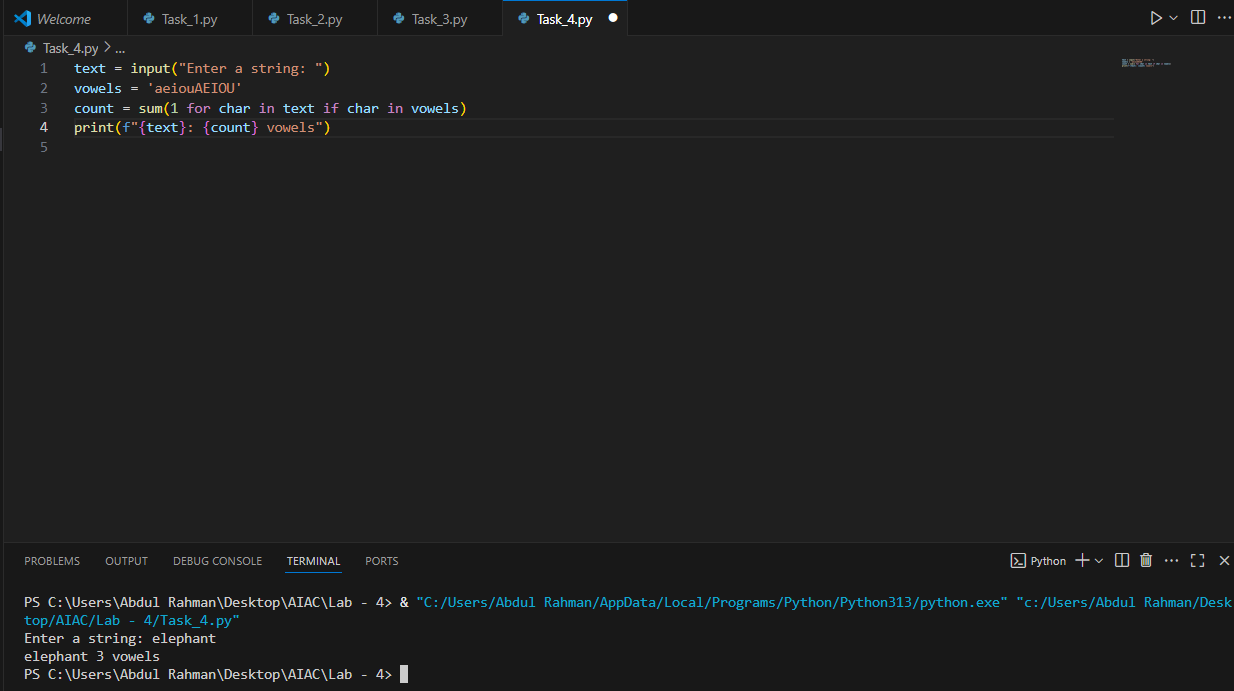


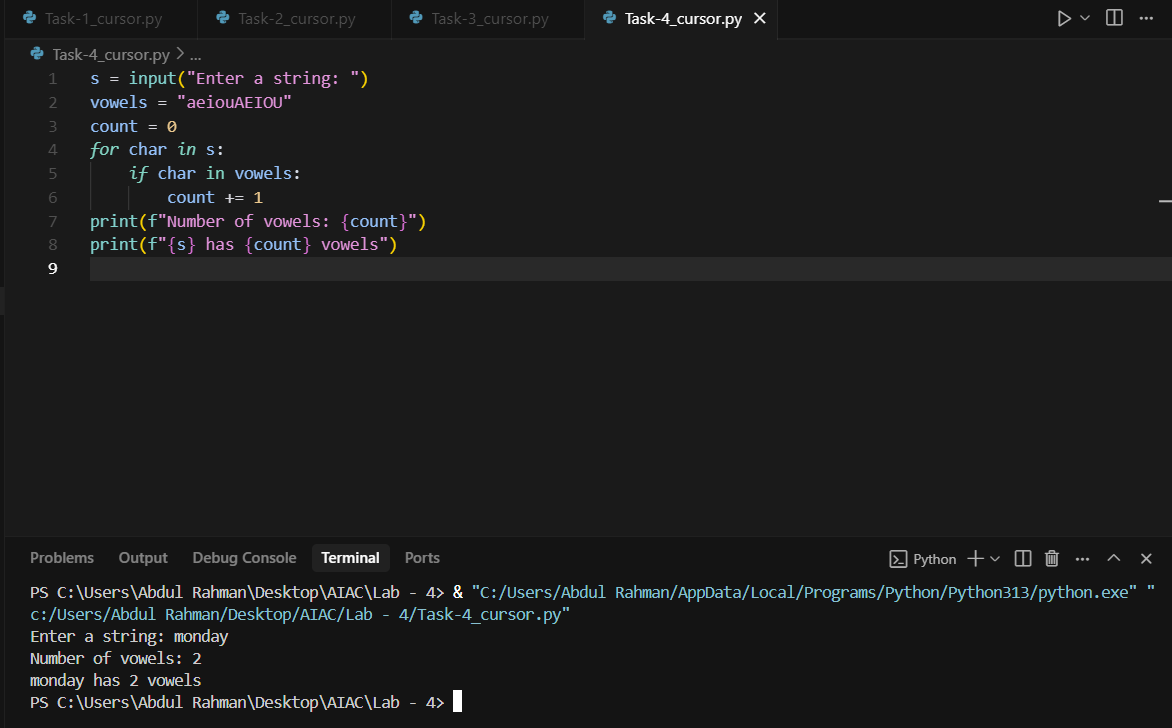
Prompt : Write a simple python programme which counts number of vowels in string , taking input string from console.

Example : maths has 1 vowels .

Mario has 3 vowels .

You has 3 vowels .





Comparative reflection :

In **zero-shot**, the model generates output without seeing any examples beforehand, relying only on its prior knowledge. In **one-shot**, the model is given a single example before generating output, helping it align more closely with the desired format or style. Zero-shot is faster but riskier, while one-shot provides more guidance and usually better accuracy.

**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

Prompt : Write a python programme which reads a file hw.txt and returns the number of lines in it after counting

Example : ‘hw.txt’ has 5 lines of info then the output would be ‘The file has 5 lines’.

‘hw.txt’ has 9 lines of info then the output would be ‘The file has 9 lines’.

