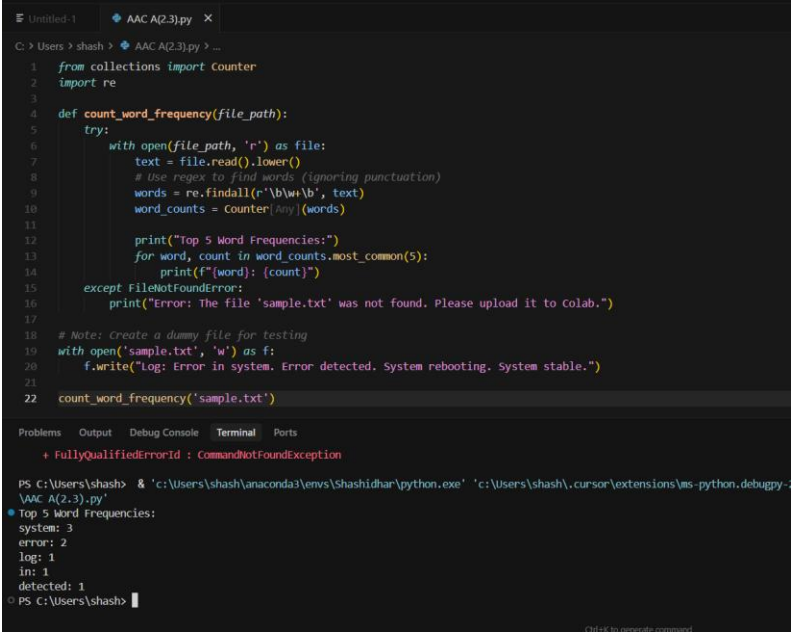


NAME:O.ISRAEL H.NO:2303A51825 BATCH:26

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING																		
Program Name: B. Tech		Assignment Type: Lab	Academic Year: 2025-2026																	
Course Coordinator Name		Dr. Rishabh Mittal																		
Instructor(s) Name		<table border="1"> <tr><td>Mr. S Naresh Kumar</td></tr> <tr><td>Ms. B. Swathi</td></tr> <tr><td>Dr. Sasanko Shekhar Gantayat</td></tr> <tr><td>Mr. Md Sallauddin</td></tr> <tr><td>Dr. Mathivanan</td></tr> <tr><td>Mr. Y Srikanth</td></tr> <tr><td>Ms. N Shilpa</td></tr> <tr><td>Dr. Rishabh Mittal (Coordinator)</td></tr> <tr><td>Dr. R. Prashant Kumar</td></tr> <tr><td>Mr. Ankushavali MD</td></tr> <tr><td>Mr. B Viswanath</td></tr> <tr><td>Ms. Sujitha Reddy</td></tr> <tr><td>Ms. A. Anitha</td></tr> <tr><td>Ms. M.Madhuri</td></tr> <tr><td>Ms. Katherashala Swetha</td></tr> <tr><td>Ms. Velpula sumalatha</td></tr> <tr><td>Mr. Bingi Raju</td></tr> </table>		Mr. S Naresh Kumar	Ms. B. Swathi	Dr. Sasanko Shekhar Gantayat	Mr. Md Sallauddin	Dr. Mathivanan	Mr. Y Srikanth	Ms. N Shilpa	Dr. Rishabh Mittal (Coordinator)	Dr. R. Prashant Kumar	Mr. Ankushavali MD	Mr. B Viswanath	Ms. Sujitha Reddy	Ms. A. Anitha	Ms. M.Madhuri	Ms. Katherashala Swetha	Ms. Velpula sumalatha	Mr. Bingi Raju
Mr. S Naresh Kumar																				
Ms. B. Swathi																				
Dr. Sasanko Shekhar Gantayat																				
Mr. Md Sallauddin																				
Dr. Mathivanan																				
Mr. Y Srikanth																				
Ms. N Shilpa																				
Dr. Rishabh Mittal (Coordinator)																				
Dr. R. Prashant Kumar																				
Mr. Ankushavali MD																				
Mr. B Viswanath																				
Ms. Sujitha Reddy																				
Ms. A. Anitha																				
Ms. M.Madhuri																				
Ms. Katherashala Swetha																				
Ms. Velpula sumalatha																				
Mr. Bingi Raju																				
CourseCode	23CS002PC304	Course Title	AI Assisted Coding																	
Year/Sem	III/II	Regulation	R23																	
Date and Day of Assignment	Week1 – Wednesday	Time(s)	23CSBTB01 To 23CSBTB52																	
Duration	2 Hours	Applicable to Batches	All batches																	
Assignment Number: 1.3(Present assignment number)/24(Total number of assignments)																				
Q.No.	Question	Expected Time to complete																		
1	Lab 2: Exploring Additional AI Coding Tools beyond Copilot – Gemini (Colab) and Cursor AI Lab Objectives: ❖ To explore and evaluate the functionality of Google Gemini for AI-	Week1 - Monday																		

	<p>assisted coding within Google Colab.</p> <ul style="list-style-type: none"> ❖ To understand and use Cursor AI for code generation, explanation, and refactoring. ❖ To compare outputs and usability between Gemini, GitHub Copilot, and Cursor AI. ❖ To perform code optimization and documentation using AI tools. <p>Lab Outcomes (LOs): After completing this lab, students will be able to:</p> <ul style="list-style-type: none"> ❖ Generate Python code using Google Gemini in Google Colab. ❖ Analyze the effectiveness of code explanations and suggestions by Gemini. ❖ Set up and use Cursor AI for AI-powered coding assistance. ❖ Evaluate and refactor code using Cursor AI features. ❖ Compare AI tool behavior and code quality across different platforms. 	
	<p>Task 1: Word Frequency from Text File</p> <ul style="list-style-type: none"> ❖ Scenario: You are analyzing log files for keyword frequency. ❖ Task: Use Gemini to generate Python code that reads a text file and counts word frequency, then explains the code. ❖ Expected Output: <ul style="list-style-type: none"> ➤ Working code ➤ Explanation ➤ Screenshot  <p>The screenshot displays a code editor with a Python script named 'AAC A(2.3).py'. The code defines a function 'count_word_frequency' that takes a file path as input. It opens the file, reads its content, and uses regular expressions to find all words, ignoring punctuation. It then uses the 'Counter' class from the 'collections' module to count the frequency of each word and prints the top 5 most common words. The code also includes an exception handler for 'FileNotFoundError' and a note to create a dummy file for testing. The terminal output shows the command being executed and the resulting word frequencies: 'system: 3', 'error: 2', 'log: 1', 'in: 1', and 'detected: 1'.</p>	

Task 2: File Operations Using Cursor AI

❖ **Scenario:**

You are automating basic file operations.

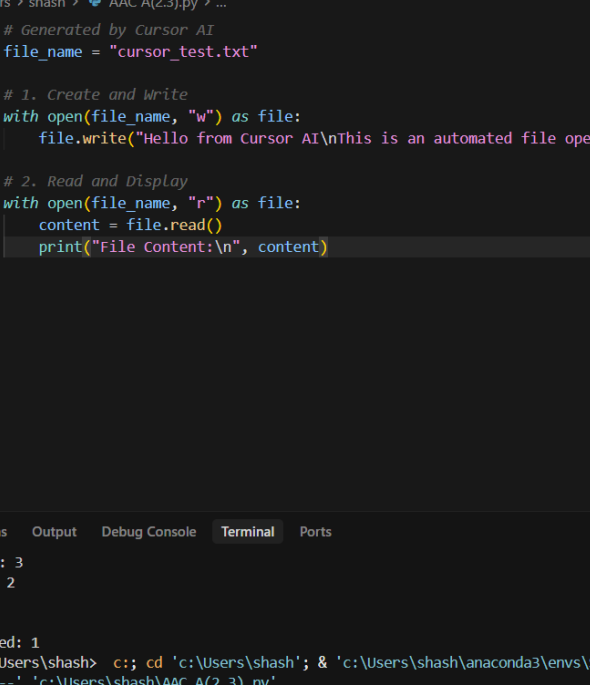
❖ Task:

Use Cursor AI to generate a program that:

- Creates a text file
- Writes sample text
- Reads and displays the content

❖ **Expected Output:**

- Functional code
- Cursor AI screenshots



The screenshot displays a Jupyter Notebook environment. At the top, a menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. Below the menu, a toolbar shows icons for saving, undo, redo, and navigation. The notebook has two tabs: 'Untitled-1' and 'AAC A(2.3).py'. The active tab shows a Python script with the following code:

```

1  # Generated by Cursor AI
2  file_name = "cursor_test.txt"
3
4  # 1. Create and Write
5  with open(file_name, "w") as file:
6      file.write("Hello from Cursor AI\nthis is an automated file operation test")
7
8  # 2. Read and Display
9  with open(file_name, "r") as file:
10     content = file.read()
11     print("File Content:\n", content)

```

Below the code editor, the 'Terminal' tab is active, showing the execution output:

```

system: 3
error: 2
log: 1
in: 1
detected: 1
● PS C:\Users\shash> c:; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashidhar\
432' '--' 'c:\Users\shash\AAC A(2.3).py'
File Content:
    Hello from Cursor AI
    This is an automated file operation test.
○ PS C:\Users\shash>

```

Task 3: CSV Data Analysis

❖ **Scenario:**

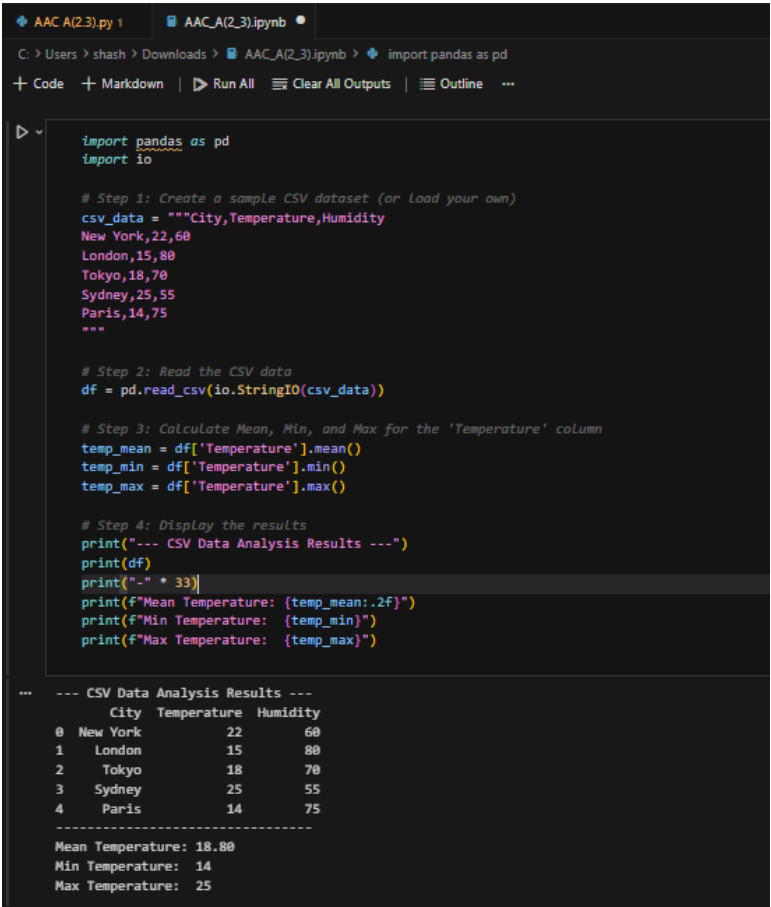
You are processing structured data from a CSV file.

❖ **Task:**

Use Gemini in Colab to read a CSV file and calculate mean, min, and max.

❖ **Expected Output:**

- Correct output
- Screenshot



The screenshot shows a Jupyter Notebook interface with a file named 'AAC_A(2,3).ipynb'. The code in the notebook performs the following steps:

- Imports pandas as pd and io.
- Creates a sample CSV dataset (or loads an existing one) with the following data:

```
csv_data = """City,Temperature,Humidity
New York,22,60
London,15,80
Tokyo,18,70
Sydney,25,55
Paris,14,75
"""
```
- Reads the CSV data into a DataFrame using `pd.read_csv(io.StringIO(csv_data))`.
- Calculates the Mean, Min, and Max for the 'Temperature' column:

```
temp_mean = df['Temperature'].mean()
temp_min = df['Temperature'].min()
temp_max = df['Temperature'].max()
```
- Displays the results using `print` statements.

The output of the notebook shows the CSV data analysis results, including the DataFrame and the calculated statistics:

```
--- CSV Data Analysis Results ---
  City  Temperature  Humidity
0  New York         22       60
1   London         15       80
2   Tokyo          18       70
3  Sydney          25       55
4   Paris          14       75
-----
Mean Temperature: 18.80
Min Temperature:  14
Max Temperature:  25
```

Task 4: Sorting Lists – Manual vs Built-in

❖ **Scenario:**

You are reviewing algorithm choices for efficiency.

❖ **Task:**

Use **Gemini** to generate:

- Bubble sort
- Python's built-in `sort()`
- Compare both implementations.

❖ **Expected Output:**

- Two versions of code
- Short comparison

```
File Edit Selection View Go Run Terminal Help
AAC A(2.3).py X
C: > Users > shash > AAC A(2.3).py > ...
1 def bubble_sort(arr):
2     n = len(arr)
3     # Outer Loop to traverse through all array elements
4     for i in range(n):
5         # Last i elements are already in place, so we ignore them
6         for j in range(0, n - i - 1):
7             # Swap if the element found is greater than the next
8             if arr[j] > arr[j + 1]:
9                 arr[j], arr[j + 1] = arr[j + 1], arr[j]
10    return arr
11
12 # Example usage
13 data = [64, 34, 25, 12, 22, 11, 90]
14 print(f"Manual Bubble Sort: {bubble_sort(data.copy())}")

Problems Output Debug Console Terminal Ports
import pandas as pd
ModuleNotFoundError: No module named 'pandas'
PS C:\Users\shash> c:; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashid
'c:\Users\shash\AAC A(2.3).py'
Traceback (most recent call last):
  File "c:\Users\shash\AAC A(2.3).py", line 1, in <module>
    import pandas as pd
ModuleNotFoundError: No module named 'pandas'
PS C:\Users\shash> c:; cd 'c:\Users\shash'; & 'c:\Users\shash\anaconda3\envs\Shashid
'c:\Users\shash\AAC A(2.3).py'
Manual Bubble Sort: [11, 12, 22, 25, 34, 64, 90]
PS C:\Users\shash>
```

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