

ASSIGNMENT – 2

NAME : M.SINDHUJA HALL TICKET NO : 2403A52060 BATCH NO :

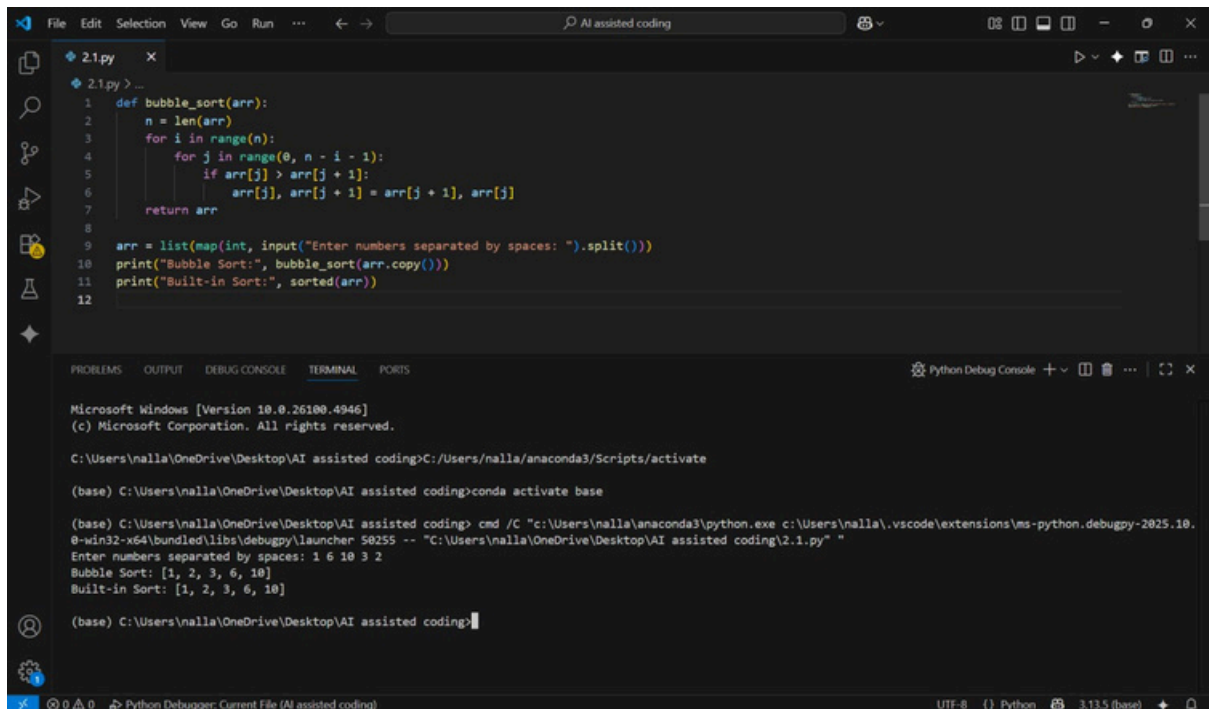
AIB03 TASK 1 : Open Google Colab and use Google Gemini to generate Python code

that performs sorting of a list using both the bubble sort algorithm and Python's built-in sort() function. Compare the two implementations

PROMPT :

Generate Python code that performs sorting of a list using both the bubble sort algorithm and Python's built-in sort() function. Compare the two implementations.

CODE :



```
2.1.py
def bubble_sort(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n - i - 1):
            if arr[j] > arr[j + 1]:
                arr[j], arr[j + 1] = arr[j + 1], arr[j]
    return arr

arr = list(map(int, input("Enter numbers separated by spaces: ").split()))
print("Bubble Sort:", bubble_sort(arr.copy()))
print("Built-in Sort:", sorted(arr))
```

Microsoft Windows [Version 10.0.26100.4946]
(c) Microsoft Corporation. All rights reserved.

C:\Users\nalla\OneDrive\Desktop\AI assisted coding>C:\Users\nalla\anaconda3\Scripts\activate

(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>conda activate base

(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>cmd /C "c:\Users\nalla\anaconda3\python.exe c:\Users\nalla\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher 50255 -- "C:\Users\nalla\OneDrive\Desktop\AI assisted coding\2.1.py" "

Enter numbers separated by spaces: 1 6 10 3 2

Bubble Sort: [1, 2, 3, 6, 10]

Built-in Sort: [1, 2, 3, 6, 10]

(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>

OBSERVATION :

This program sorts a list of numbers using ***bubble sort***.
It defines a function `bubble_sort(arr)` that takes a list as input.
Inside the function, nested loops compare each pair of adjacent elements and swap them if they are in the wrong order, repeatedly pushing the largest numbers to the end.
The program asks the user to enter numbers separated by spaces and converts them into a list.
Finally, it prints the sorted list using both the custom `bubble_sort` function and Python's built-in `sorted()` function for comparison.

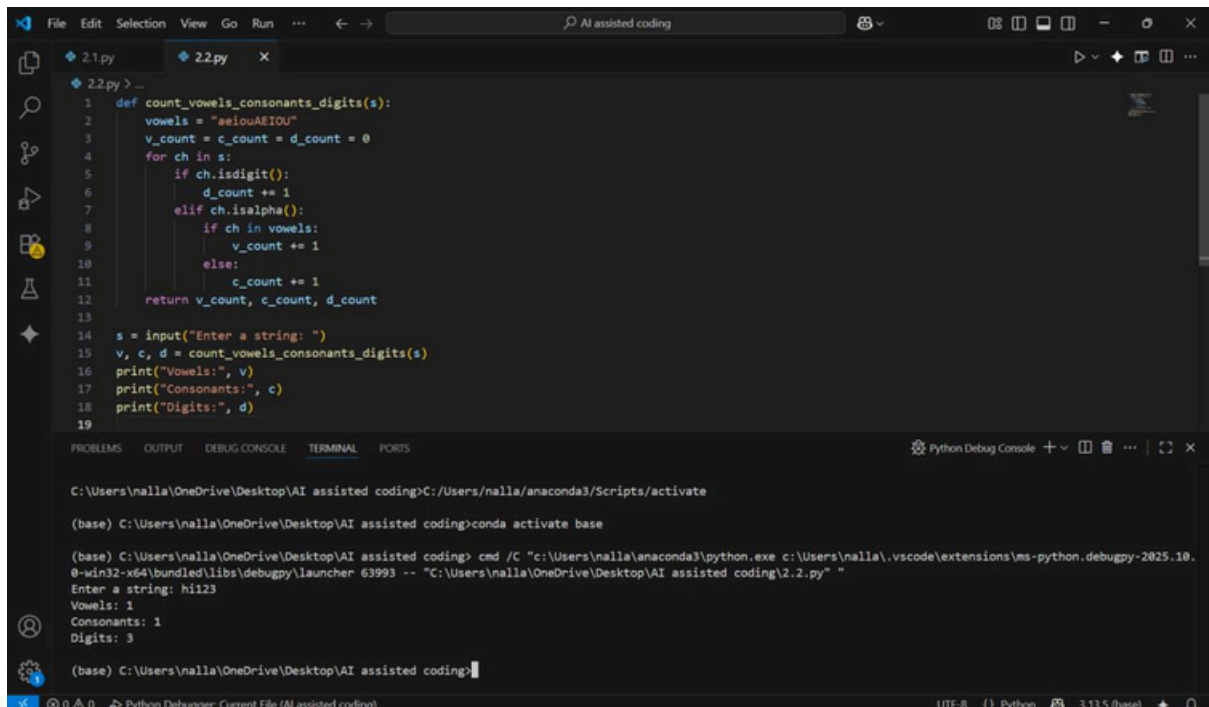
TASK 2:

In Colab, use Google Gemini to generate a Python function that takes a string and returns: The number of vowels, The number of consonants, The number of digits in the string

PROMPT :

Generate a Python function that takes a string and returns:
The number of vowels, The number of consonants, The number of digits in the string

CODE :



The screenshot shows a VS Code editor with a Python file named 22.py. The code defines a function `count_vowels_consonants_digits(s)` that counts vowels, consonants, and digits in a string. The function uses `isalpha()` to check for vowels and `isdigit()` for digits. The main program prompts the user to enter a string, calls the function, and prints the counts. The terminal shows the execution of the script, with the input string "hi123" and the output: Vowels: 1, Consonants: 1, Digits: 3.

```
1 def count_vowels_consonants_digits(s):
2     vowels = "aeiouAEIOU"
3     v_count = c_count = d_count = 0
4     for ch in s:
5         if ch.isdigit():
6             d_count += 1
7         elif ch.isalpha():
8             if ch in vowels:
9                 v_count += 1
10            else:
11                c_count += 1
12    return v_count, c_count, d_count
13
14 s = input("Enter a string: ")
15 v, c, d = count_vowels_consonants_digits(s)
16 print("Vowels:", v)
17 print("Consonants:", c)
18 print("Digits:", d)
19
```

```
C:\Users\nalla\OneDrive\Desktop\AI assisted coding>C:\Users\nalla\anaconda3\Scripts\activate
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>conda activate base
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>cmd /C "c:\Users\nalla\anaconda3\python.exe c:\Users\nalla\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher 63993 -- "C:\Users\nalla\OneDrive\Desktop\AI assisted coding\22.py" "
Enter a string: hi123
Vowels: 1
Consonants: 1
Digits: 3
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>
```

OBSERVATION:

This program counts the number of vowels, consonants, and digits in a string. It defines a function `count_vowels_consonants_digits(s)` that takes a string as input. Inside the function, it loops through each character and checks if it is a digit, a vowel, or a consonant, updating separate counters for each. The program asks the user to enter a string and calls the function with that input. Finally, it prints the counts of vowels, consonants, and digits in the string.

TASK 3 :

Install and set up Cursor AI. Use it to generate a Python program that performs file handling:

Create a text file

Write sample text

Read and display the content

PROMPT :

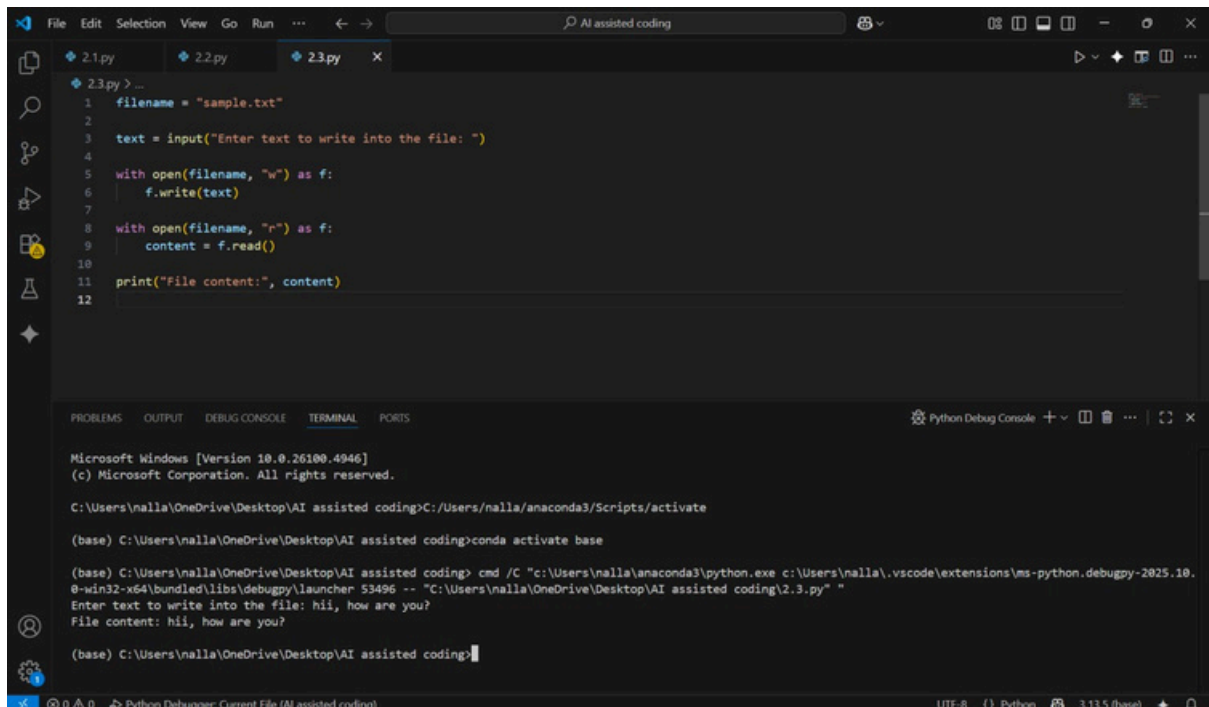
generate a Python program that performs file handling:

Create a text file

Write sample text

Read and display the content

CODE :



The screenshot shows a Visual Studio Code editor window with a Python file named 23.py. The code in the file is as follows:

```
1 filename = "sample.txt"
2
3 text = input("Enter text to write into the file: ")
4
5 with open(filename, "w") as f:
6     f.write(text)
7
8 with open(filename, "r") as f:
9     content = f.read()
10
11 print("File content:", content)
12
```

Below the editor, the Python Debug Console and Terminal are visible. The terminal shows the following commands and output:

```
Microsoft Windows [Version 10.0.26100.4946]
(c) Microsoft Corporation. All rights reserved.

C:\Users\nalla\OneDrive\Desktop\AI assisted coding>C:\Users\nalla\anaconda3\Scripts\activate
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>conda activate base
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>cmd /C "c:\Users\nalla\anaconda3\python.exe c:\Users\nalla\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher 53496 -- "C:\Users\nalla\OneDrive\Desktop\AI assisted coding\23.py" "
Enter text to write into the file: hii, how are you?
File content: hii, how are you?

(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>
```

OBSERVATION :

This program allows the user to write text into a file and then read it back. It first defines a file named "sample.txt" and asks the user to enter some text. Using the with open statement in write mode ("w"), the program writes the entered text into the file. After that, it opens the same file in read mode ("r") to read its contents. Finally, the program displays the text from the file, showing that the writing and reading operations were successful.

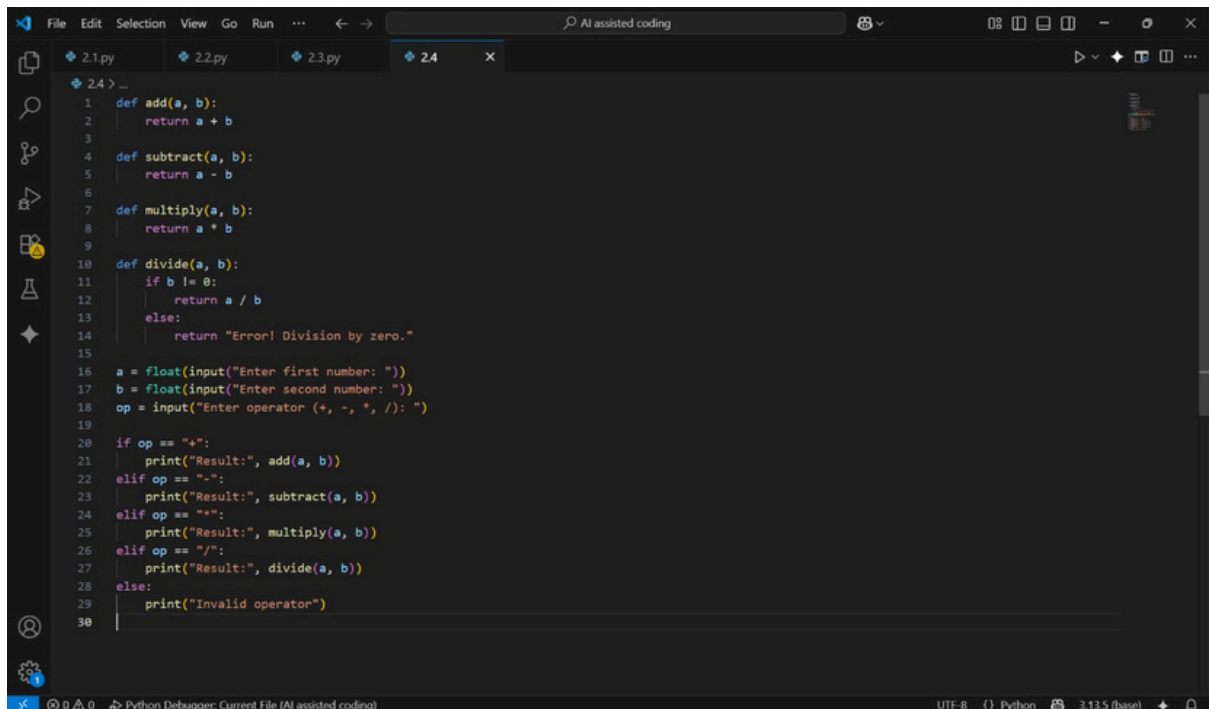
TASK 4 :

Ask Google Gemini to generate a Python program that implements a simple calculator using functions (add, subtract, multiply, divide). Then, ask Gemini to explain how the code works

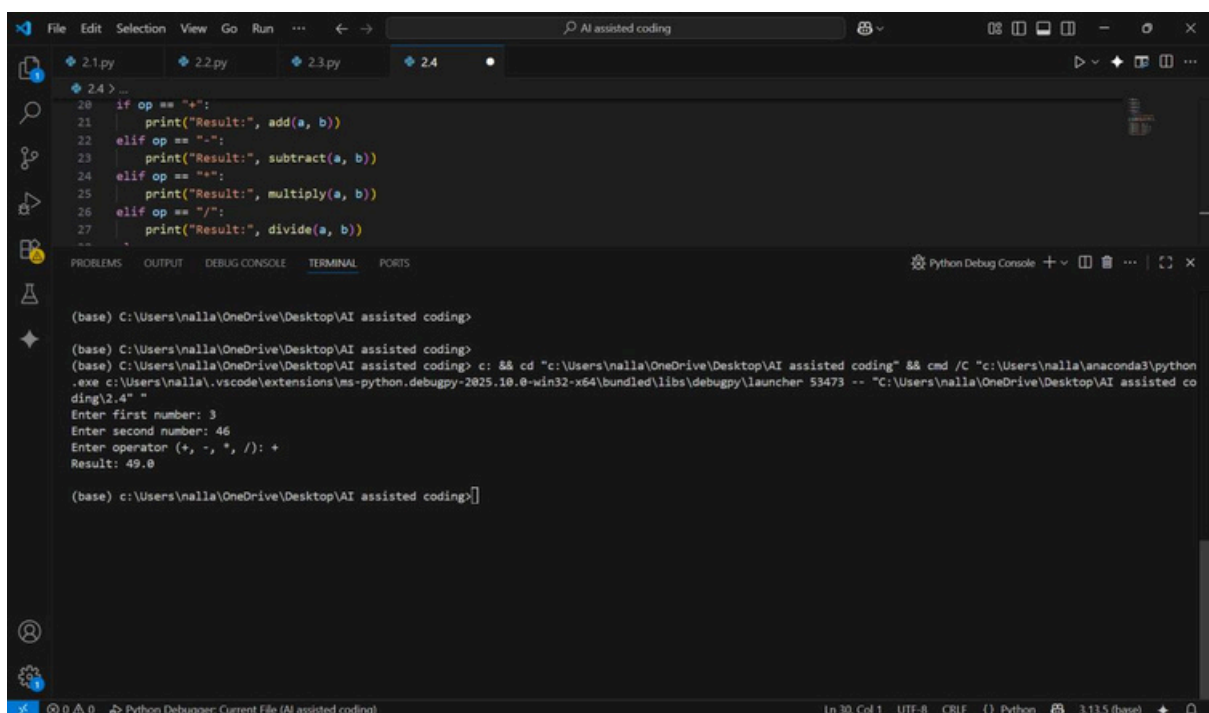
PROMPT :

Generate a Python program that implements a simple calculator using functions (add, subtract, multiply, divide).

CODE :



```
1 def add(a, b):
2     return a + b
3
4 def subtract(a, b):
5     return a - b
6
7 def multiply(a, b):
8     return a * b
9
10 def divide(a, b):
11     if b != 0:
12         return a / b
13     else:
14         return "Error! Division by zero."
15
16 a = float(input("Enter first number: "))
17 b = float(input("Enter second number: "))
18 op = input("Enter operator (+, -, *, /): ")
19
20 if op == "+":
21     print("Result:", add(a, b))
22 elif op == "-":
23     print("Result:", subtract(a, b))
24 elif op == "*":
25     print("Result:", multiply(a, b))
26 elif op == "/":
27     print("Result:", divide(a, b))
28 else:
29     print("Invalid operator")
30
```



```
20 if op == "+":
21     print("Result:", add(a, b))
22 elif op == "-":
23     print("Result:", subtract(a, b))
24 elif op == "*":
25     print("Result:", multiply(a, b))
26 elif op == "/":
27     print("Result:", divide(a, b))
--
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding> c: && cd "c:\Users\nalla\OneDrive\Desktop\AI assisted coding" && cmd /C "c:\Users\nalla\anaconda3\python.exe c:\Users\nalla\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher 53473 -- "C:\Users\nalla\OneDrive\Desktop\AI assisted coding\2.4" "
Enter first number: 3
Enter second number: 46
Enter operator (+, -, *, /): +
Result: 49.0
(base) c:\Users\nalla\OneDrive\Desktop\AI assisted coding>]
```

OBSERVATION :

This program is a simple calculator that performs addition, subtraction, multiplication, or division based on user input. It defines four separate

functions: add, subtract, multiply, and divide, each taking two numbers as arguments. The divide function also checks for division by zero to avoid errors. The program then asks the user to enter two numbers and an operator (+, -, *, or /). Based on the operator entered, it calls the corresponding function and prints the result, or displays an error message if the operator is invalid.

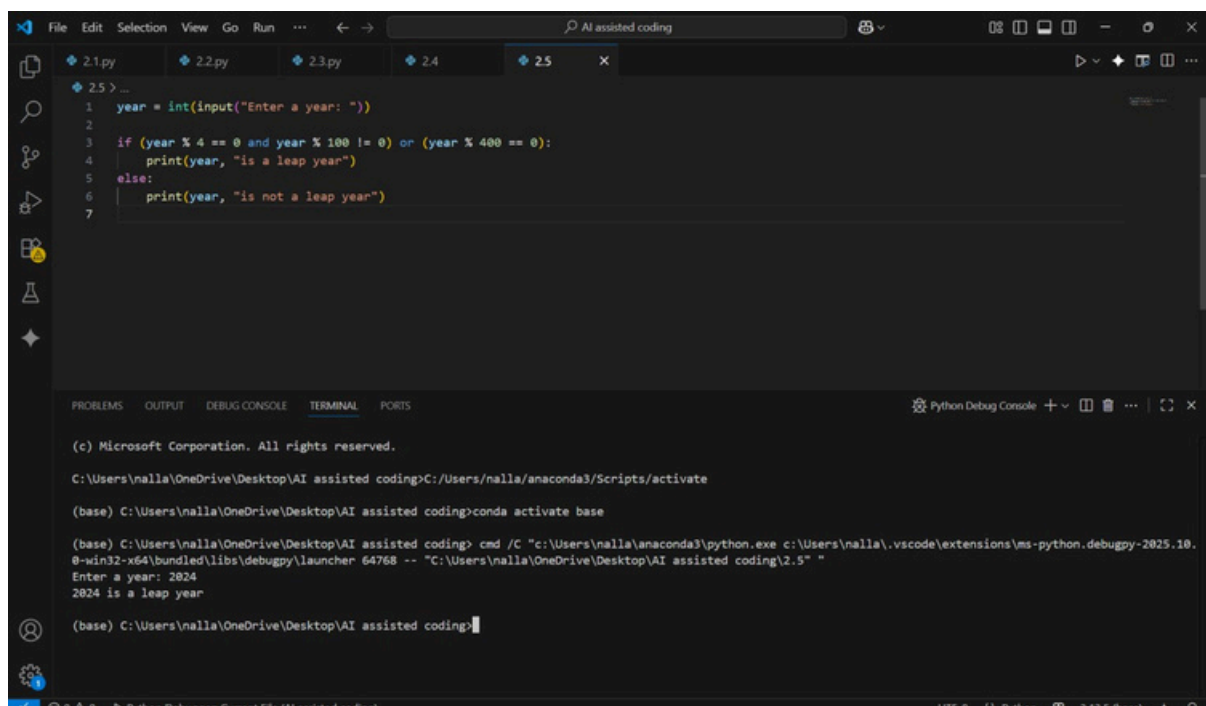
TASK 5 :

Use Cursor AI to create a Python program that checks if a given year is a leap year or not. Try different prompt styles and see how Cursor modifies its code suggestions

PROMPT :

create a Python program that checks if a given year is a leap year or not.

CODE :



```
File Edit Selection View Go Run ... AI assisted coding
2.1.py 2.2.py 2.3.py 2.4 2.5 x
2.5 > ...
1 year = int(input("Enter a year: "))
2
3 if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
4     print(year, "is a leap year")
5 else:
6     print(year, "is not a leap year")
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python Debug Console
(c) Microsoft Corporation. All rights reserved.
C:\Users\nalla\OneDrive\Desktop\AI assisted coding>C:\Users\nalla\anaconda3\Scripts\activate
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>conda activate base
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>cmd /C "c:\Users\nalla\anaconda3\python.exe c:\Users\nalla\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher 64768 -- "C:\Users\nalla\OneDrive\Desktop\AI assisted coding\2.5" "
Enter a year: 2024
2024 is a leap year
(base) C:\Users\nalla\OneDrive\Desktop\AI assisted coding>
```

OBSERVATION:

This program checks whether a given year is a leap year or not. It first asks the user to enter a year. Then it uses a condition that checks if the

year is divisible by 4 but not by 100, or if it is divisible by 400. If either condition is true, the program prints that the year is a leap year; otherwise, it prints that it is not a leap year.