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	Week1 - Tuesday	**Time(s)**	
Duration	2 Hours	**Applicableto Batches**	24CSBTB01 To 24CSBTB39
AssignmentNumber:1.2(Present assignment number)/**24**(Total number of assignments)			

Q.No.	**Question**	***ExpectedTime to complete***
1	Lab 1: Environment Setup – GitHub Copilot and VS Code Integration **Lab Objectives:** ● To install and configure GitHub Copilot in Visual Studio Code. ● To explore AI-assisted code generation using GitHub Copilot.	Week1 - wednesday

- To analyze the accuracy and effectiveness of Copilot's code suggestions.
- To understand prompt-based programming using comments and code context

Lab Outcomes (LOs):
After completing this lab, students will be able to:

- Set up GitHub Copilot in VS Code successfully.
- Use inline comments and context to generate code with Copilot.
- Evaluate AI-generated code for correctness and readability.
- Compare code suggestions based on different prompts and programming styles.

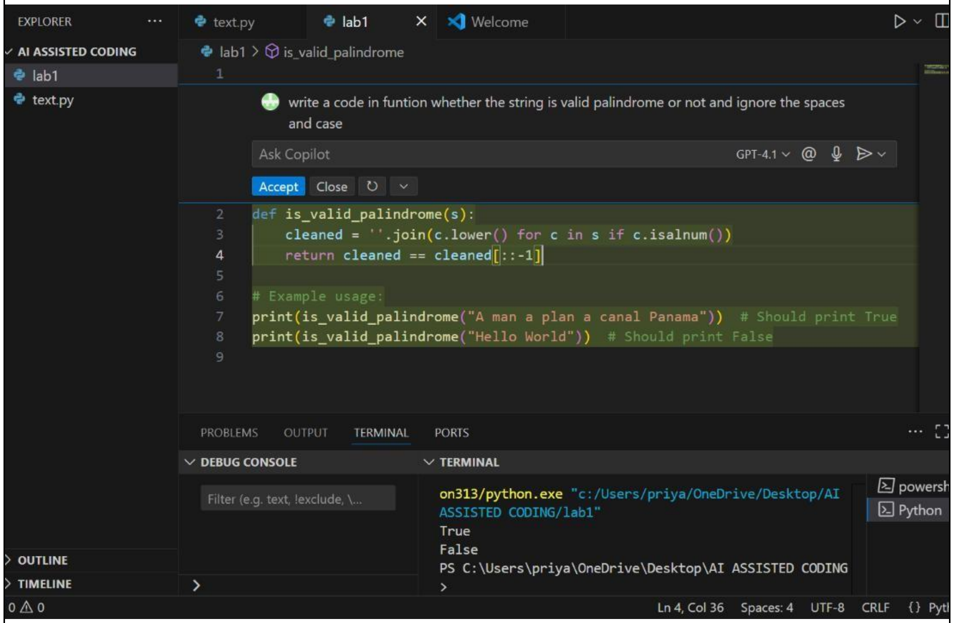
Task Description#1

- Write a comment: # Function to check if a string is a valid palindrome (ignoring spaces and case) and allow Copilot to complete it.

Expected Output#1

- A function that correctly returns True for phrases like "A man a plan a canal Panama"

PROMPT-1
WRITE A CODE IN FUNCTION WHETHER THE STRING IS VALID PALINDROME OR NOT AND IGNORE THE SPACES AND CASE

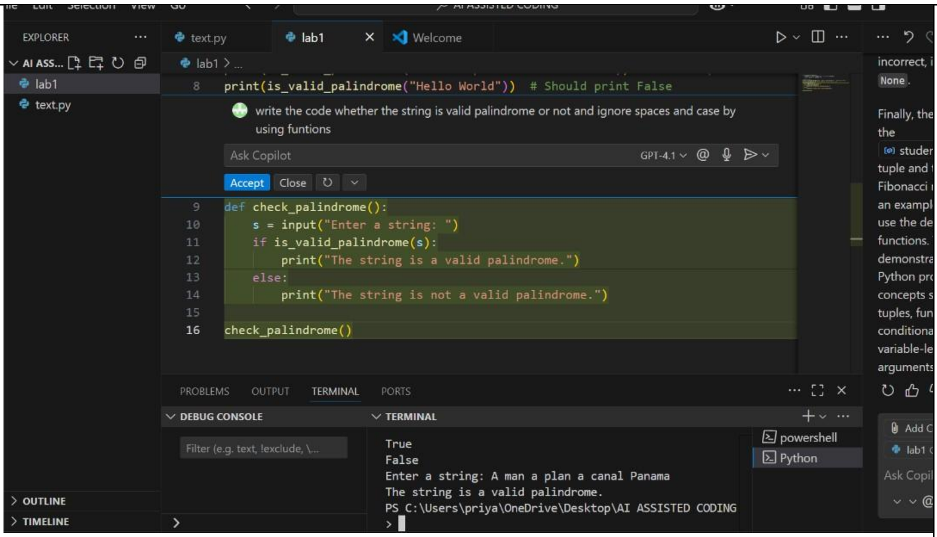


write a code in funtion whether the string is valid palindrome or not and ignore the spaces and case

```python
def is_valid_palindrome(s):
    cleaned = ''.join(c.lower() for c in s if c.isalnum())
    return cleaned == cleaned[::-1]

# Example usage:
print(is_valid_palindrome("A man a plan a canal Panama"))  # Should print True
print(is_valid_palindrome("Hello World"))  # Should print False
```

```
on313/python.exe "c:/Users/priya/OneDrive/Desktop/AI
ASSISTED CODING/lab1"
True
False
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
>
```

PROMPT-2
WRITE THE CODE WHETHER THE STRING IS VALID PALINDROME OR NOT AND IGNORE SPACES AND CASE BY USING FUNCTIONS



```python
print(is_valid_palindrome("Hello World"))  # Should print False
```

write the code whether the string is valid palindrome or not and ignore spaces and case by using funtions

```python
def check_palindrome():
    s = input("Enter a string: ")
    if is_valid_palindrome(s):
        print("The string is a valid palindrome.")
    else:
        print("The string is not a valid palindrome.")

check_palindrome()
```

```
True
False
Enter a string: A man a plan a canal Panama
The string is a valid palindrome.
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
>
```

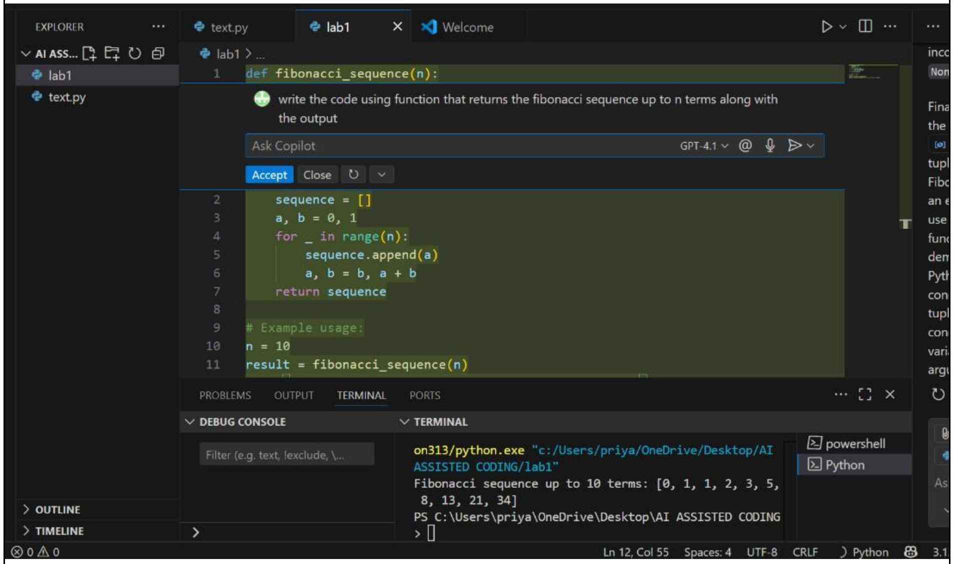
Task Description#2

- Generate a Python function that returns the Fibonacci sequence up to n terms. Prompt with only a function header and docstring

Expected Output#2

- AI completes the function logic using loop or recursion with accurate output

PROMPT-1
WRITE A CODE USING FUNCTIONS THAT RETURNS THE FIBONACCI SEQUENCE UP TO N TERMS ALONG WITH THE OUTPUT



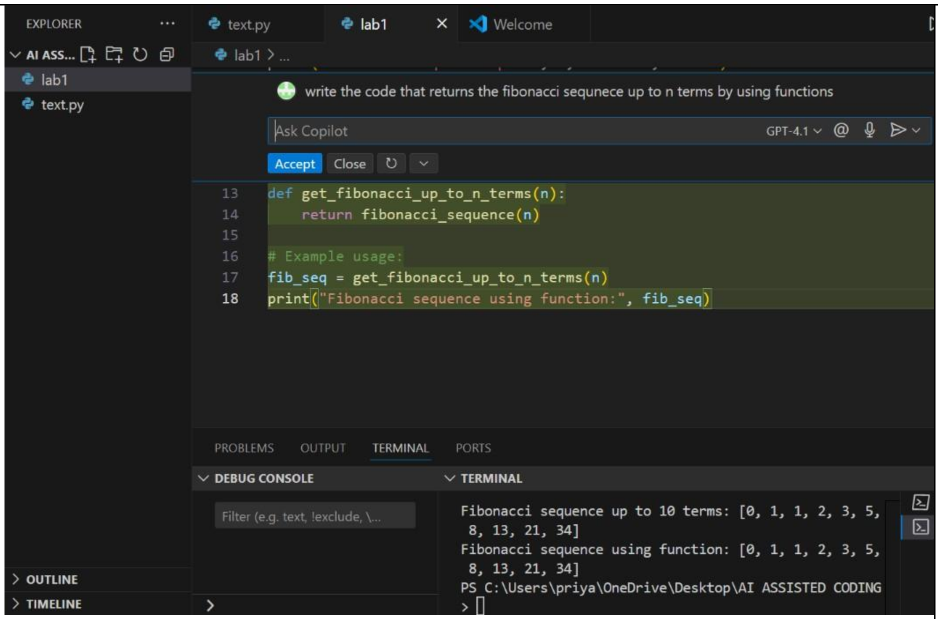
```python
def fibonacci_sequence(n):
```

write the code using function that returns the fibonacci sequence up to n terms along with the output

```python
    sequence = []
    a, b = 0, 1
    for _ in range(n):
        sequence.append(a)
        a, b = b, a + b
    return sequence

# Example usage:
n = 10
result = fibonacci_sequence(n)
```

```
on313/python.exe "c:/Users/priya/OneDrive/Desktop/AI
ASSISTED CODING/lab1"
Fibonacci sequence up to 10 terms: [0, 1, 1, 2, 3, 5,
 8, 13, 21, 34]
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
>
```

PROMPT-2
WRITE THE CODE THAT RETURNS THE FIBONACCI SEQUNECE UP TO N TERMS BY USING FUNCTIONS



write the code that returns the fibonacci sequnece up to n terms by using functions

```python
def get_fibonacci_up_to_n_terms(n):
    return fibonacci_sequence(n)

# Example usage:
fib_seq = get_fibonacci_up_to_n_terms(n)
print("Fibonacci sequence using function:", fib_seq)
```

```
Fibonacci sequence up to 10 terms: [0, 1, 1, 2, 3, 5,
 8, 13, 21, 34]
Fibonacci sequence using function: [0, 1, 1, 2, 3, 5,
 8, 13, 21, 34]
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
>
```

Task Description#3

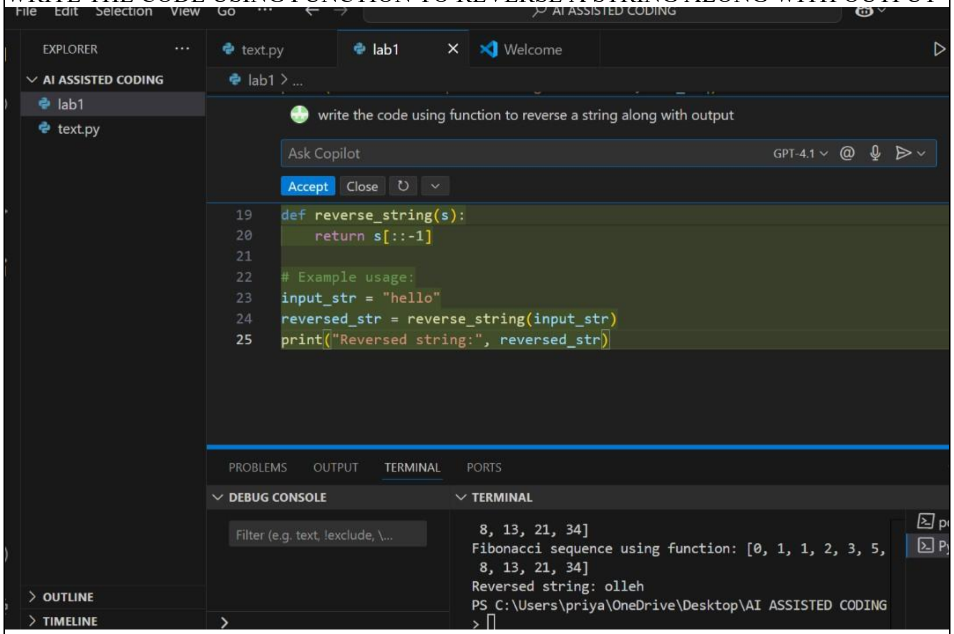
- Write a comment like # Function to reverse a string and use Copilot to generate the function.

Expected Output#3

- Auto-completed reverse function

PROMPT-1

WRITE THE CODE USING FUNCTION TO REVERSE A STRING ALONG WITH OUTPUT



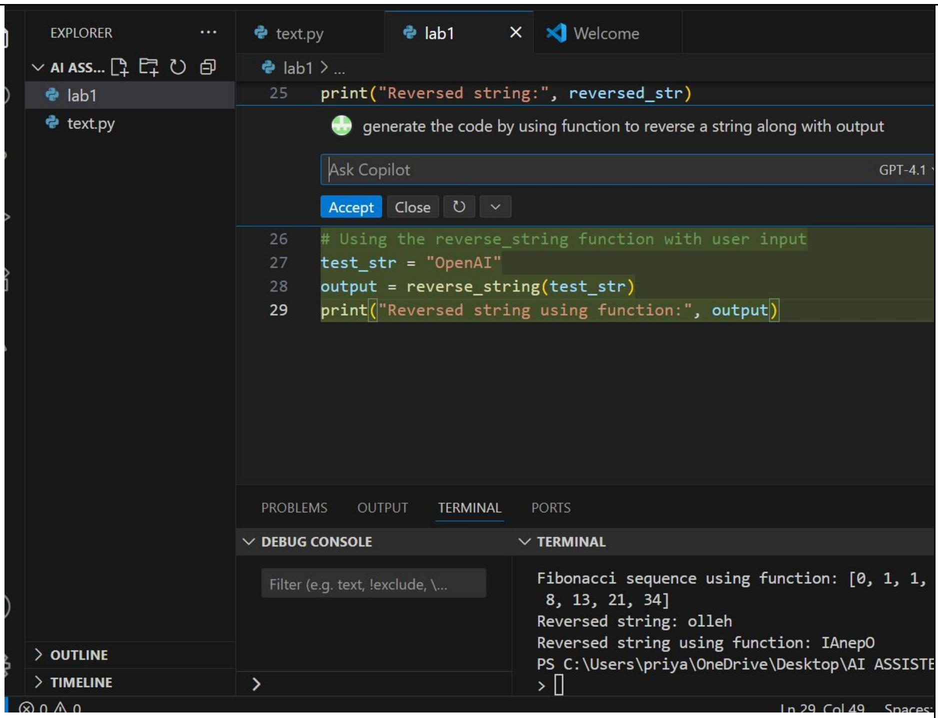
write the code using function to reverse a string along with output

```python
def reverse_string(s):
    return s[::-1]

# Example usage:
input_str = "hello"
reversed_str = reverse_string(input_str)
print("Reversed string:", reversed_str)
```

```
 8, 13, 21, 34]
Fibonacci sequence using function: [0, 1, 1, 2, 3, 5,
 8, 13, 21, 34]
Reversed string: olleh
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
>
```

PROMPT-2

GENERATE THE CODE BY USING FUNCTION TO REVERSE A STRING ALONG WITH OUTPUT



```python
print("Reversed string:", reversed_str)
```

generate the code by using function to reverse a string along with output

```python
# Using the reverse_string function with user input
test_str = "OpenAI"
output = reverse_string(test_str)
print("Reversed string using function:", output)
```

```
Fibonacci sequence using function: [0, 1, 1,
 8, 13, 21, 34]
Reversed string: olleh
Reversed string using function: IAnepO
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTE
> 
```

Task Description#4

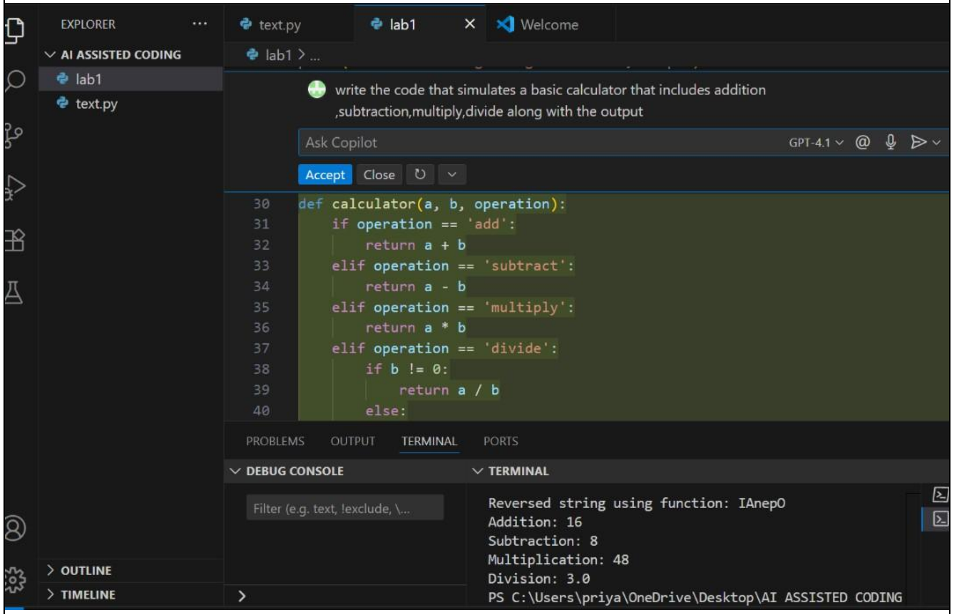
- Generate a program that simulates a basic calculator (add, subtract, multiply, divide). Write the comment: # Simple calculator with 4 operations and let AI complete it.

Expected Output#4

- Fully working calculator with input/output and operator selection logic

PROMPT-1

WRITE THE CODE THAT SIMULATES A BASIC CALCULATOR THAT INCLUDES ADDITION ,SUBTRACTION,MULTIPLY,DIVIDE ALONG WITH OUTPUT



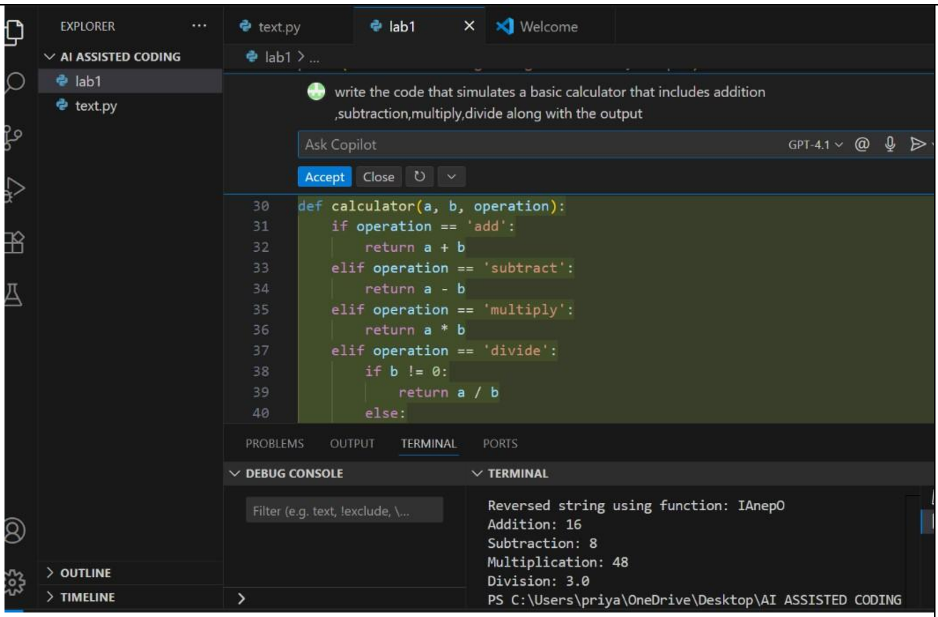
write the code that simulates a basic calculator that includes addition ,subtraction,multiply,divide along with the output

```python
def calculator(a, b, operation):
    if operation == 'add':
        return a + b
    elif operation == 'subtract':
        return a - b
    elif operation == 'multiply':
        return a * b
    elif operation == 'divide':
        if b != 0:
            return a / b
        else:
```

```
Reversed string using function: IAnepO
Addition: 16
Subtraction: 8
Multiplication: 48
Division: 3.0
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
```

PROMPT-2

GENERATE A BASIC CALCULATOR WITH BASIC OPERATION SUCH AS ADDITION,SUBTRACTION,MULTIPLY,DIVISON



write the code that simulates a basic calculator that includes addition ,subtraction,multiply,divide along with the output

```
def calculator(a, b, operation):
    if operation == 'add':
        return a + b
    elif operation == 'subtract':
        return a - b
    elif operation == 'multiply':
        return a * b
    elif operation == 'divide':
        if b != 0:
            return a / b
        else:
```

```
Reversed string using function: IAnepO
Addition: 16
Subtraction: 8
Multiplication: 48
Division: 3.0
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
```

Task Description#5

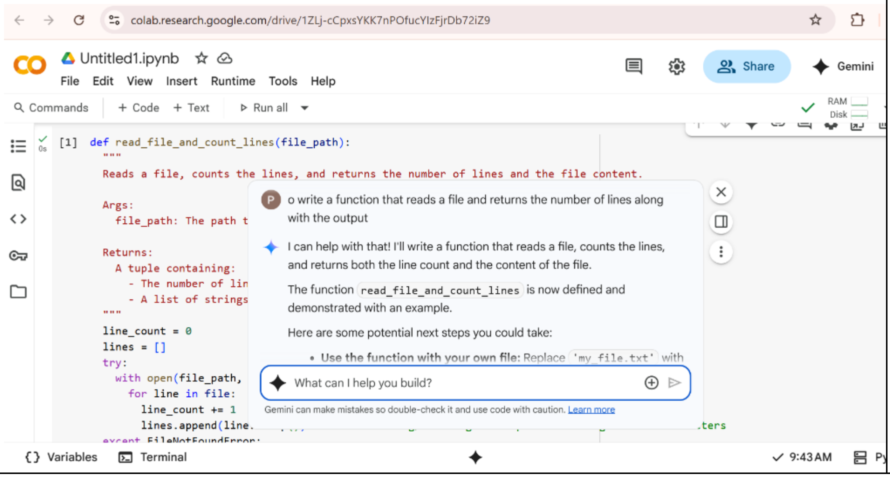
- Use a comment to instruct AI to write a function that reads a file and returns the number of lines..

Expected Output#5

- Functional implementation using open() or with open() and readlines()

PROMPT-1

WRITE A FUNCTION THAT READS A FILE AND RETURNS THE NUMBER OF LINES ALONG WITH THE OUTPUT



```
def read_file_and_count_lines(file_path):
  """
  Reads a file, counts the lines, and returns the number of lines and the file content.

  Args:
    file_path: The path t

  Returns:
    A tuple containing:
      - The number of lin
      - A list of strings
  """
  line_count = 0
  lines = []
  try:
    with open(file_path,
      for line in file:
        line_count += 1
        lines.append(line.
```

o write a function that reads a file and returns the number of lines along with the output

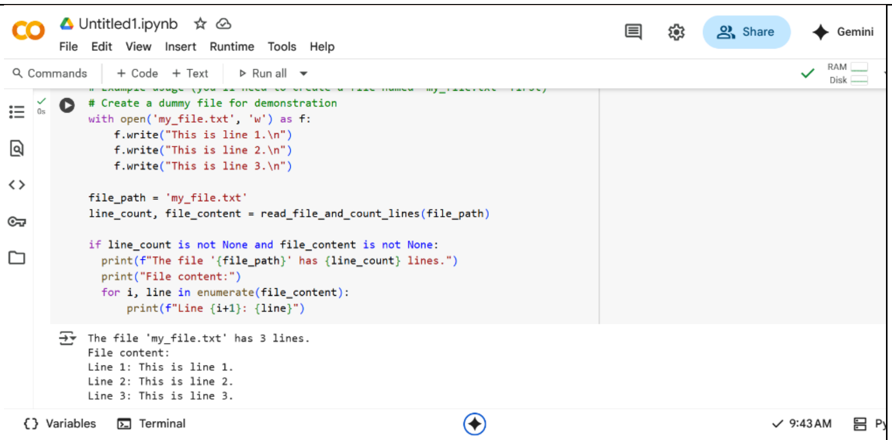
I can help with that! I'll write a function that reads a file, counts the lines, and returns both the line count and the content of the file.

The function `read_file_and_count_lines` is now defined and demonstrated with an example.

Here are some potential next steps you could take:

- **Use the function with your own file:** Replace `'my_file.txt'` with

Gemini can make mistakes so double-check it and use code with caution. Learn more

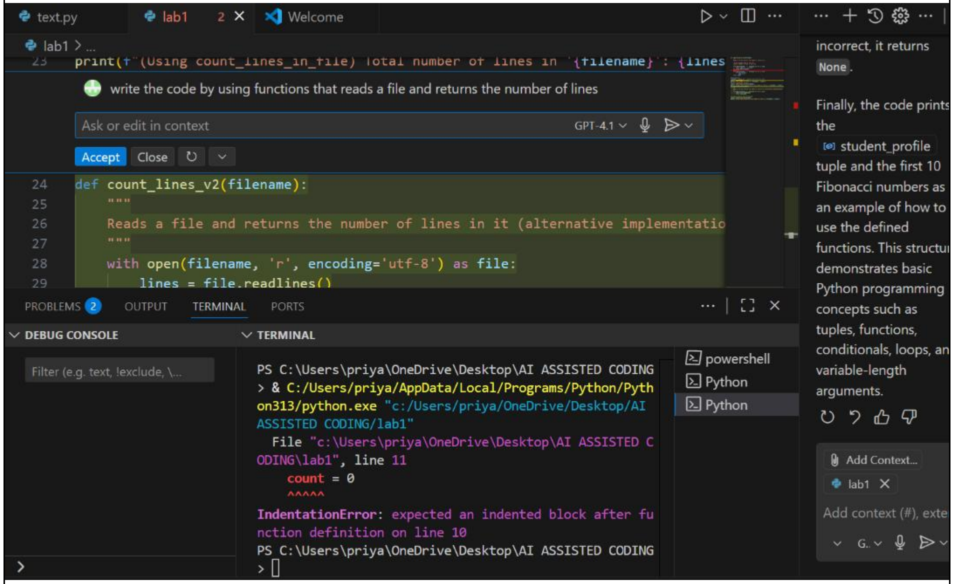


```python
# Create a dummy file for demonstration
with open('my_file.txt', 'w') as f:
    f.write("This is line 1.\n")
    f.write("This is line 2.\n")
    f.write("This is line 3.\n")

file_path = 'my_file.txt'
line_count, file_content = read_file_and_count_lines(file_path)

if line_count is not None and file_content is not None:
  print(f"The file '{file_path}' has {line_count} lines.")
  print("File content:")
  for i, line in enumerate(file_content):
      print(f"Line {i+1}: {line}")
```

```
The file 'my_file.txt' has 3 lines.
File content:
Line 1: This is line 1.
Line 2: This is line 2.
Line 3: This is line 3.
```

PROMPT-2

WRITE THE CODE BY USING FUNCTIONS THAT READS A FILE AND RETURN THE NUMBER OF LINES



write the code by using functions that reads a file and returns the number of lines

```python
print(f"(Using count_lines_in_file) Total number of lines in '{filename}': {lines
def count_lines_v2(filename):
    """
    Reads a file and returns the number of lines in it (alternative implementatio
    """
    with open(filename, 'r', encoding='utf-8') as file:
        lines = file.readlines()
```

```
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
> & C:/Users/priya/AppData/Local/Programs/Python/Pyth
on313/python.exe "c:/Users/priya/OneDrive/Desktop/AI
ASSISTED CODING/lab1"
  File "c:\Users\priya\OneDrive\Desktop\AI ASSISTED C
ODING\lab1", line 11
    count = 0
    ^^^^^
IndentationError: expected an indented block after fu
nction definition on line 10
PS C:\Users\priya\OneDrive\Desktop\AI ASSISTED CODING
> 
```

incorrect, it returns None.

Finally, the code prints the student_profile tuple and the first 10 Fibonacci numbers as an example of how to use the defined functions. This structu demonstrates basic Python programming concepts such as tuples, functions, conditionals, loops, an variable-length arguments.

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
Task #1	0.5
Task #2	0.5
Task #3	0.5
Task #4	0.5
Task #5	0.5
Total	**2.5 Marks**