AI assistant 1.2 assignment

Name:balaji

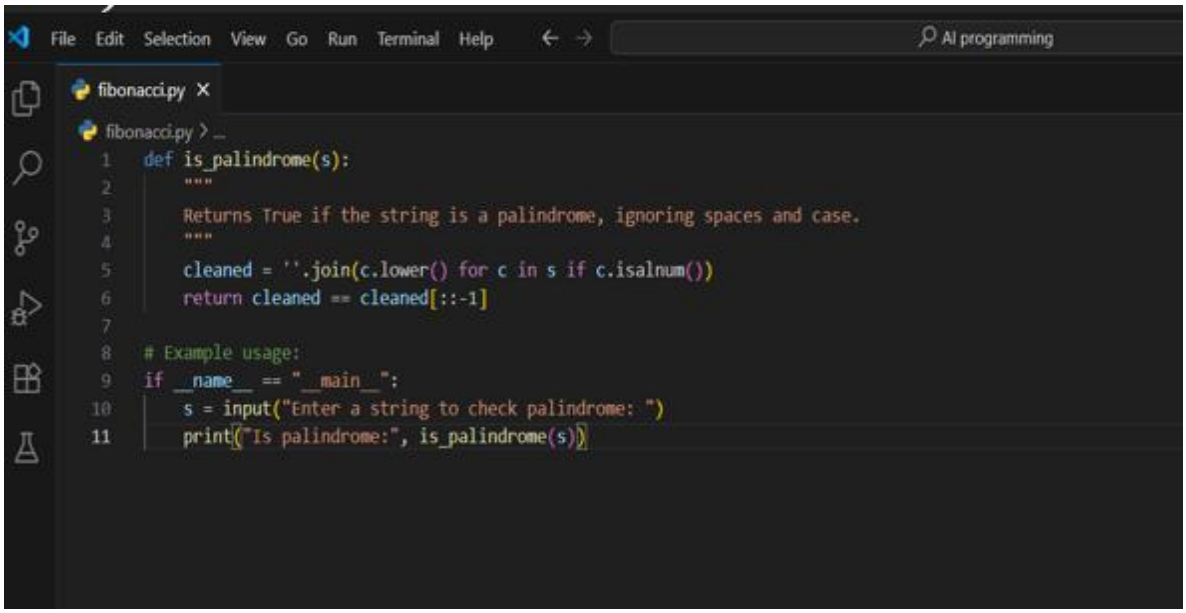
Roll no:2403A51102

Batch:06

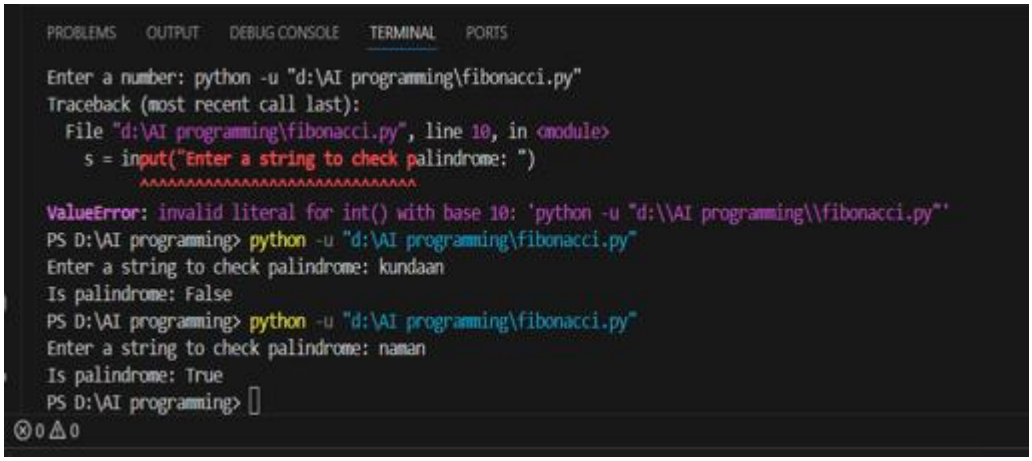
task Description#1

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

Code:



Output:



Observation

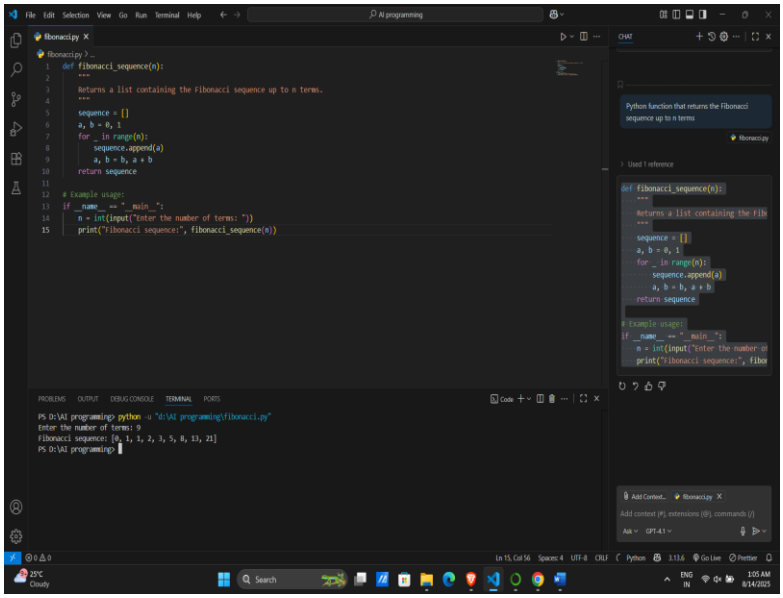
The function first removes all spaces from the input string and converts it to lowercase to ensure the comparison is case-insensitive and whitespace-insensitive. It then compares the cleaned string to its reverse to determine if it's a palindrome.

Task:**2**

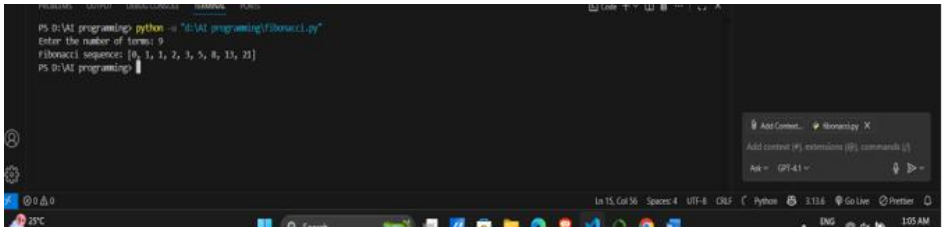
- Generate a Python function that returns the Fibonacci sequence up to n terms. Prompt with only a function header and docstring

Prompt: Python function that returns the Fibonacci sequence up to n terms

Code:



Output:



Observation:

The Python program successfully generates the Fibonacci sequence up to a user-defined number of terms using an iterative approach.

- **The function fibonacci_sequence(n) takes an integer input n and returns a list containing the first n terms of the Fibonacci sequence.**

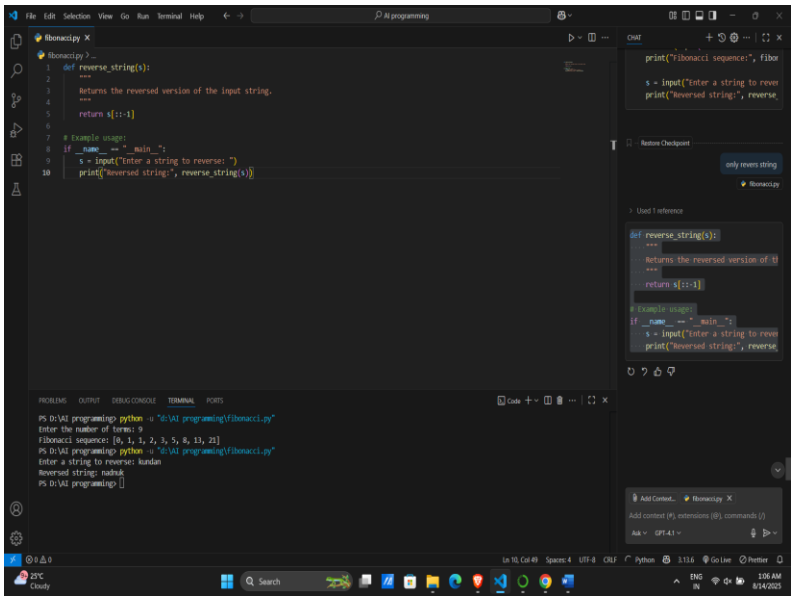
- **The variables a and b are initialized to 0 and 1, representing the first two terms of the Fibonacci sequence.**
- **A for loop is used to generate the sequence:**
  - **In each iteration, the current value of a is appended to the list.**
  - **The values of a and b are then updated as a = b and b = a + b, ensuring the next term follows the Fibonacci pattern.**
- **The program uses input() to take the number of terms from the user, converts it to an integer using int(), and prints the resulting sequence using print().**

Task:3

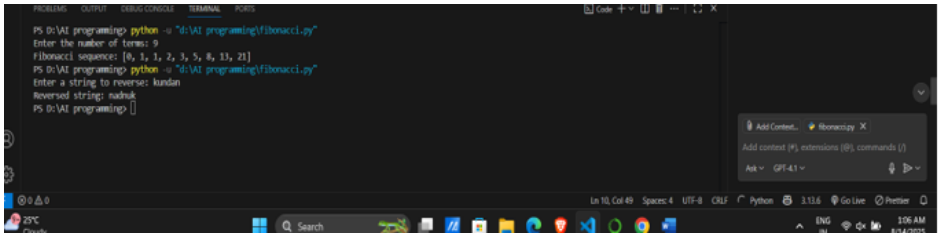
- Write a comment like # Function to reverse a string and use Copilot to generate the function.

Prompt:write a python code to reverse the given string.make code simple

Code:



Output:



Observation:

Observation (Theory)

The Python program is designed to reverse a user-input string and display the result.

- The function reverse_string(s) takes a string s as input and returns its reversed form using Python's slicing technique:

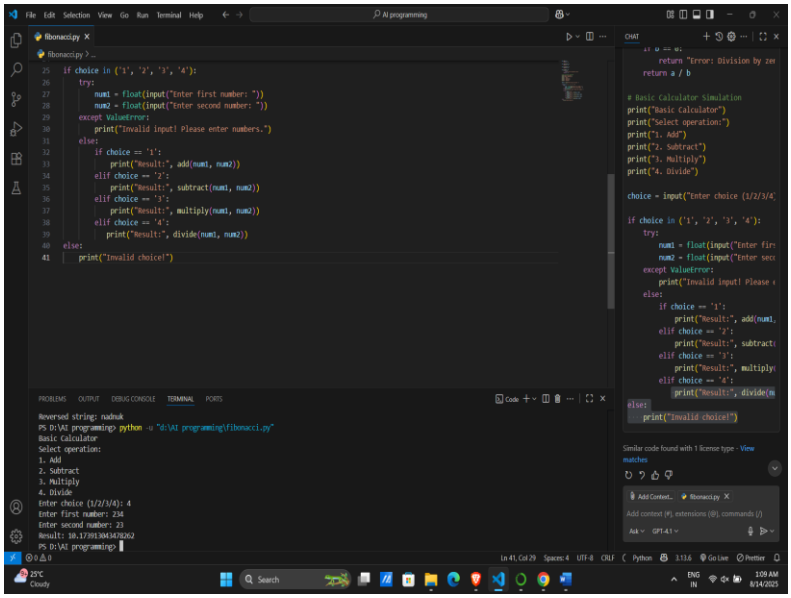
- return s[::-1]
    - The slice [::-1] effectively reverses the string by starting from the end and stepping backwards.
- The program prompts the user to enter a string using the input() function.
- The entered string is passed to the reverse_string() function.
- The reversed string is then displayed using the print() function.

Task: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

Prompt:make a python code to make calculator that perform a basic calculator (add, subtract, multiply, divide).

Code:



Output:

Observation:

The Python program functions as a basic calculator, allowing the user to perform arithmetic operations such as addition, subtraction, multiplication, and division.

☑ Program Workflow:

1. User Interface:

- The program displays a menu listing four operations:
- 1. Add
- 2. Subtract
- 3. Multiply
- 4. Divide

2. User Input:

- The user is prompted to select an operation (choice) by entering a number between 1 and 4.
- If the input is valid (choice in ('1', '2', '3', '4')), the program proceeds to request two numbers as input.
- A try-except block is used to ensure both numbers are valid floating-point values. If not, it catches the error and prompts the user accordingly.

3. Operation Handling:

- Based on the user's choice:
  - 1: Calls add(num1, num2)

- 2: Calls subtract(num1, num2)
- 3: Calls multiply(num1, num2)
- 4: Calls divide(num1, num2)

- The result is then printed using print("Result:", ...).

4. Invalid Input Handling:

- If the user enters a choice outside of the range 1–4, the program prints "Invalid choice!".

- Task:5

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

Code:

```
def count_lines(filename):
    """Read a file and return the number of lines."""
    try:
        with open(filename, 'r') as file:
            lines = file.readlines()
            return len(lines)
    except FileNotFoundError:
        print(f"File '{filename}' not found.")
        return -1
    except Exception as e:
        print(f"Error reading file: {e}")
        return -1

def main():
    print("=== File Line Counter ===")

    filename = input("Enter filename: ")

    line_count = count_lines(filename)

    if line_count >= 0:
        print(f"Number of lines: {line_count}")
    else:
        print("Could not count lines.")

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

Output:

```
Hi I am Balaji.
I am from Warangal.
```

```
PS D:\AI programming> py
                     > python -u "d:\AI programming\line.py"
=== File Line Counter ===
Enter filename: kund
Number of lines: 2
PS D:\AI programming>
```

Code
Code