AI ASSISTED CODING

ROLL NO: 2503A51L16

NAME: K. JASHUVA

LAB NO: 10

BRANCH: CSE

TASK 1

Task Description: AI-Assisted Code Review (Basic Errors)

- Write python program as shown below.
- Use an AI assistant to review and suggest corrections.

Given code:

```
def calcFact(n):
  result=1
  x=0
  for i in range(1,n):
      result=result*i
  return result

def main():
     num = 5
     FACT = calcFact(num)
     print("the factorial of",num,"is",FACT)
     t=10
     if FACT>10:
            print("BIG Number")
     else:
      print("small number")

main()
```

Prompt:

I am submitting a Python program for an AI-assisted code review. Please identify and correct basic syntax errors, logical mistakes, naming convention violations, and any inefficient code. Make sure the corrected version follows Python best practices (PEP 8), uses meaningful variable names, and removes unused or redundant code. Also, include comments explaining what was fixed.

Code Generated:

```python
def calc_fact(n):
    # Initialize result to 1 (since factorial of 0 is 1)
    result = 1
    # Loop from 1 to n (inclusive)
    for i in range(1, n + 1):
        result = result * i
    return result

def main():
    num = 5
    # Calculate factorial
    fact = calc_fact(num)
    print("The factorial of", num, "is", fact)
    t = 10
    # Check if factorial is greater than 10
    if fact > t:
        print("BIG Number")
    else:
        print("small number")

main()
```

Output:

```python
def calc_fact(n):
    # Initialize result to 1 (since factorial of 0 is 1)
    result = 1
    # Loop from 1 to n (inclusive)
    for i in range(1, n + 1):
        result = result * i
    return result

def main():
    num = 5
    # Calculate factorial
    fact = calc_fact(num)
    print("The factorial of", num, "is", fact)
    t = 10
    # Check if factorial is greater than 10
    if fact > t:
        print("BIG Number")
    else:
        print("small number")
```

```
PS C:\B.TECH\AI LAB\10> & C:/Users/kamer/anaconda3/python.exe "c:/B.TECH/AI LAB/10/TASK 1.py"
The factorial of 5 is 120
BIG Number
PS C:\B.TECH\AI LAB\10>
```

Observations & Corrections:

1. The factorial calculation should start from 1, not 0, and should include n in the range.
2. The variable x=0 is unused and can be removed.
3. The range in for i in range(1, n): should be range(1, n+1) to include n.
4. Variable names should be lowercase by convention (FACT → fact).
5. The indentation and structure can be improved for clarity.
6. Add comments for better understanding.

TASK 2

Task Description: Automatic Inline Comments

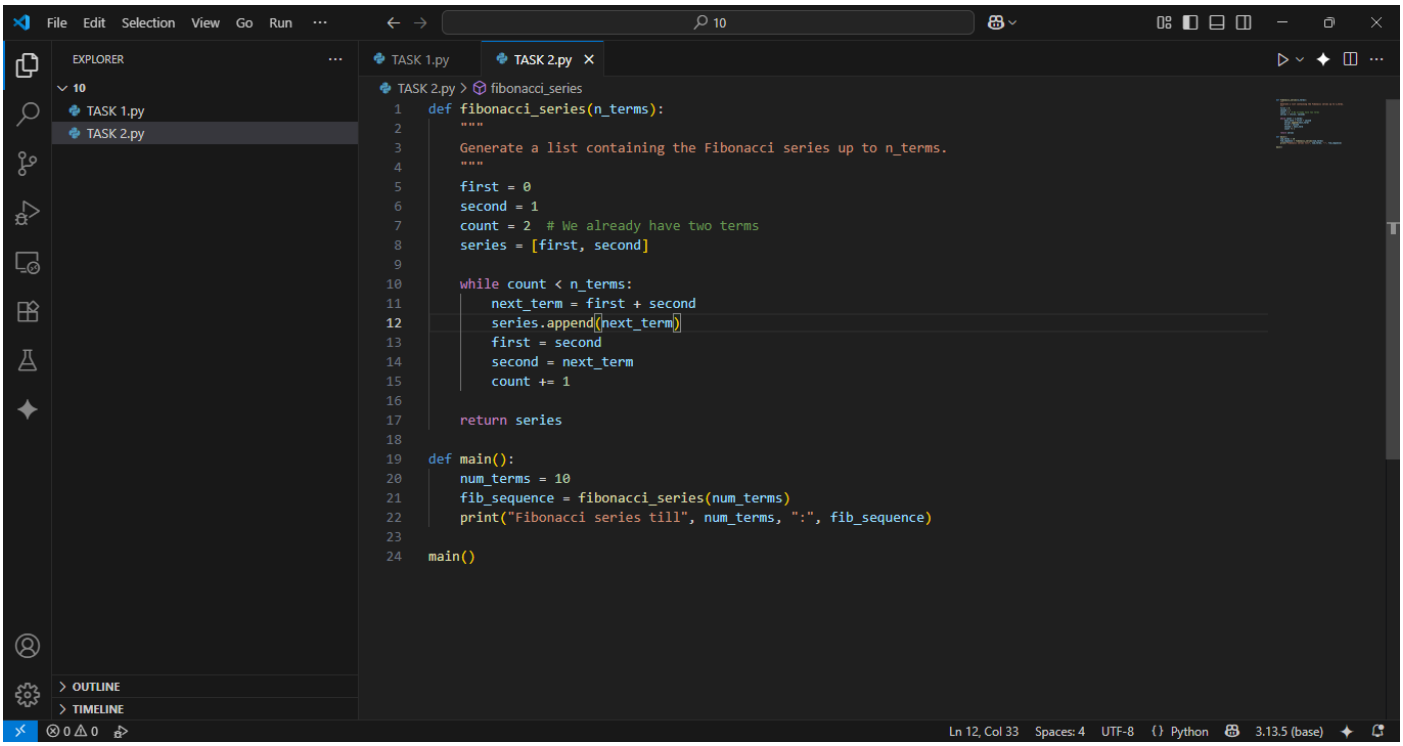
- Write the Python code for Fibonacci as shown below and execute.
- Ask AI to improve variable names, add comments, and apply PEP8 formatting (cleaned up).
- Students evaluate which suggestions improve readability most.one.

Given Code:

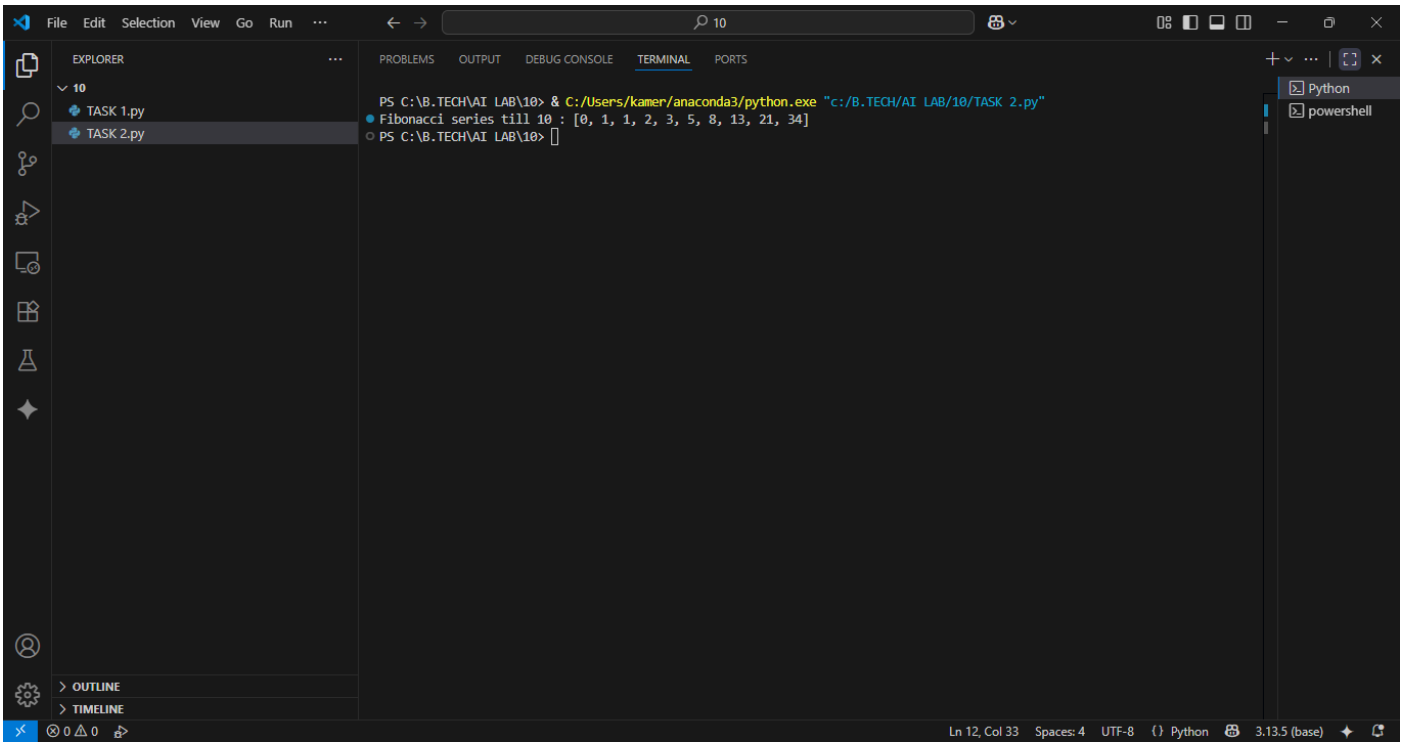
```python
def f1(xX):
    a=0
    b=1
    c=2
    Zz=[a,b]
    while c<=xX:
        d=a+b
        Zz.append(d)
        a=b
        b=d
        c=c+1
    return Zz

def m():
      NN=10
      ans=f1(NN)
      print("fib series till",NN,":",ans)

m()
```

Prompt: Please review the following Python code for generating a Fibonacci series. Improve the variable names for clarity, add inline comments to explain the logic, and apply PEP8 formatting for better readability. Return a corrected and well-documented version of the code.

Code Generated:



Output:



Observations:

- Variable names are descriptive (n_terms, first, second, series, etc.).
- Added a docstring and inline comments for clarity.
- PEP8 formatting: consistent indentation, spaces around operators, and lowercase function names.
- The code is easier to read and understand.

Task 3

Task Description:

- Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide).
- Incorporate manual docstring in code with NumPy Style
- Use AI assistance to generate a module-level docstring + individual function docstrings.
- Compare the AI-generated docstring with your manually written one.