

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
ProgramName: B. Tech		SET-B	AcademicYear:2025-2026
Roll No.			
Instructor(s)Name		1. Dr. Venkataramana 2. Dr. Ch. Sridhar 3. Mr. Kundan Kumar	
CourseCode	24CS002PC215	CourseTitle	AI Assisted Coding
Year/Sem	II/I	Regulation	R24
Date and Day of Assignment	22.08.2025	Time(s)	01.00PM To 03.00PM
Duration	2 Hours	Applicable to Batches	24BTCAIAIB09, 24BTCAIAIB10
Lab Test :1 (Present test number)/4 (Total number of tests)			
Q.No.	Question	Expected Time to complete	
1	<p>Task Description#1</p> <ul style="list-style-type: none"> Write python program for a function factorial_febo(n) that return both factorial of given number and Fibonacci series as per input "n" <p>Expected Output#1</p> <ul style="list-style-type: none"> factorial_febo(n) and output <p>prompt : write a python program for a function factorial_febo(n) that return both factorial of given number and Fibonacci series as per input "n"</p> <p>Code:</p>	22.08.2025 03.00PM	

The screenshot shows a code editor with a file explorer on the left. The file explorer shows a project named 'AIAC' with a subdirectory 'lab test -1' containing a file 'task1.py'. The code editor has two tabs: 'Welcome' and 'task1.py'. The 'task1.py' tab is active, showing a Python script. The script defines a function 'factorial_fibo(n)' that calculates the factorial and Fibonacci series for a given input 'n'. The script includes comments and example usage code. The status bar at the bottom indicates 'Ln 20, Col 51 Spaces: 4 UT'.

```
1 def factorial_fibo(n):  
    # write a python code that gives the output of both factorial and febonacci series as per the given input  
    # write a fython python program for a function factorial_fibo(n) that return both factorial of  
    # given number and Fibonacci series as per input "n"  
    Ask or edit in context GPT-4.1 @  
    Accept Close  
2 # Calculate factorial  
3 factorial = 1  
4 for i in range(1, n + 1):  
5     factorial *= i  
6  
7 # Generate Fibonacci series up to n terms  
8 fibo_series = []  
9 a, b = 0, 1  
10 for _ in range(n):  
11     fibo_series.append(a)  
12     a, b = b, a + b  
13  
14 return factorial, fibo_series  
15  
16 # Example usage:  
17 n = int(input("Enter a number: "))  
18 fact, fibo = factorial_fibo(n)  
19 print(f"Factorial of {n}: {fact}")  
20 print(f"Fibonacci series up to {n} terms: {fibo}")
```

Output:

The screenshot shows a terminal window with the following output:

```
> & C:/Users/sahas/AppData/Local/Programs/Python/Python311/Python.exe -u C:/Users/sahas/OneDrive/Desktop/AIAC/lab test -1/task1.py  
Enter a number: 4  
Factorial of 4: 24  
Fibonacci series up to 4 terms: [0, 1, 1, 2]
```

Explanation: this code does give the output both factorial and febonacci series of a number

Task Description#2

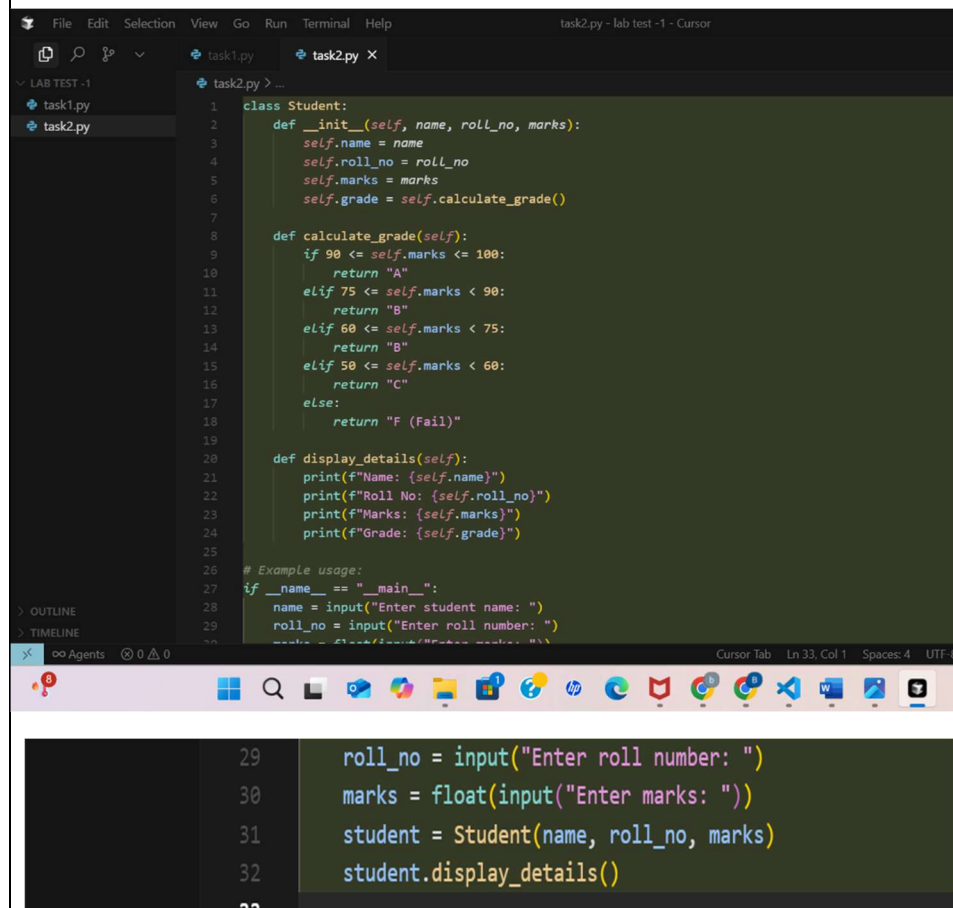
- Write Python program in cursor AI for **student class** with attributes like Name, Roll no, Marks and also construct a method i.e., **display_details** to display name, rollno, marks and grades

Table: Grades Classification Based on Marks	
Marks Range (%)	Grade
90 – 100	A+
75 – 89	A
60 – 74	B
50 – 59	C
Below 50	F (Fail)

Expected Output#2

- Student Class and Display Details
- Prompt: write a python code for student class with attributes like Name, Rollno, Marks and also construct a method i.e., display_details to display name, rollno, marks and grades as follows 90 – 100 → A+, 75 – 89 → A, 60 – 74 → B, 50 – 59 → C, Below 50 → F (Fail)

Code:



```
1 class Student:
2     def __init__(self, name, roll_no, marks):
3         self.name = name
4         self.roll_no = roll_no
5         self.marks = marks
6         self.grade = self.calculate_grade()
7
8     def calculate_grade(self):
9         if 90 <= self.marks <= 100:
10             return "A"
11         elif 75 <= self.marks < 90:
12             return "B"
13         elif 60 <= self.marks < 75:
14             return "B"
15         elif 50 <= self.marks < 60:
16             return "C"
17         else:
18             return "F (Fail)"
19
20     def display_details(self):
21         print(f"Name: {self.name}")
22         print(f"Roll No: {self.roll_no}")
23         print(f"Marks: {self.marks}")
24         print(f"Grade: {self.grade}")
25
26 # Example usage:
27 if __name__ == "__main__":
28     name = input("Enter student name: ")
29     roll_no = input("Enter roll number: ")
30     marks = float(input("Enter marks: "))
31     student = Student(name, roll_no, marks)
32     student.display_details()
```

Output:

```
Problems Output Debug Console Terminal Ports
PS C:\Users\sahas\OneDrive\Desktop\AIAC\lab test -1> & C:/Users/sahas/AppData/Local/Programs/Python/Python38-32/Python.exe C:/Users/sahas/OneDrive/Desktop/AIAC/lab test -1/task2.py"
Enter student name: sahasra
Enter roll number: 23
Enter marks: 67
Name: sahasra
Roll No: 23
Marks: 67.0
Grade: B
```

Explanation: this code is about student marks and grade of the student and details of the student such as name,roll no.

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

Evaluation Criteria:

Criteria	Max Marks
Factorial Function (Task#1)	5
Sorting Function (Task#2)	5
Viva	5
Total	15 Marks