

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
ProgramName: B. Tech		SET-C	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		ExpectedTime to complete
1	<p><b>Task Description#1</b></p> <ul style="list-style-type: none"> <li>Write python program for a <b>SRU_Student</b> class with attributes like Name, Roll No, and Department. And, add a method <b>Student_Data</b> that helps to store details of student as a text document. Use VS Code with GitHub Copilot</li> </ul> <p><b>Prompt:</b> Generate python program for a <b>SRU_Student</b> class with attributes like Name, Roll No, and Department, add a method <b>Student_Data</b> that helps to store details of student as a text document</p> <p><b>Code:</b></p> <pre> task1.py x Welcome task1.py &gt; ... 1 class SRU_Student: 2     def __init__(self, name, roll_no, department): 3         print("• SRU_Student class initialization") 4         self.name = name 5         self.roll_no = roll_no 6         self.department = department 7 8     def Student_Data(self, filename): 9         print("• Student_Data") 10        with open(filename, 'w') as file: 11            file.write(f"Name: {self.name}\n") 12            file.write(f"Roll No: {self.roll_no}\n") 13            file.write(f"Department: {self.department}\n") 14        print(f"• Text Document having student data + '{filename}' created successfully") 15 16 17 student = SRU_Student("Pranith reddy", "52269", "Computer Science") 18 student.Student_Data("student_data.txt") 19           </pre>		22.08.2025 03.00PM

## Output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\pc\OneDrive\Documents\AIAC\lab test> & C:\Users\pc\AppData\Local\Programs\Python\Python313\python.exe "c:/Users/pc/OneDrive/Documents/AIAC/lab test/task1.py"
• SRU Student class initialization
• Student_Data
• Text Document having student data -> 'student_data.txt' created successfully
PS C:\Users\pc\OneDrive\Documents\AIAC\lab test> |
```

- Defines a class [SRU\\_Student](#) with attributes for name, roll number, and department.
- Initializes these attributes when a new student object is created and prints a message.
- Provides a method [Student\\_Data](#) that writes the student's details to a text file and prints status messages.
- Creates an instance of the class with sample data and calls the method to save the details in a file.

### Expected Output#1

- SRU\_Student class initialization
- Student\_Data
- Text Document having student data

## Task Description#2

- Write a python function to return sum of even and odd numbers from the given list

### Prompt:

A python function to return sum of even and odd numbers from the given list of sample numbers 1,2,3,4,5,6

### Code:

```
task2.py • task1.py
task2.py > sum_even_odd
1 def sum_even_odd(numbers):
2     |
3     even_sum = 0
4     odd_sum = 0
5
6     for value in numbers:
7         if value % 2 == 0:
8             even_sum += value
9         else:
10            odd_sum += value
11
12    return even_sum, odd_sum
13
14
15 if __name__ == "__main__":
16    sample_numbers = [1, 2, 3, 4, 5, 6]
17    even_total, odd_total = sum_even_odd(sample_numbers)
18    print(f"Input: {sample_numbers}")
19    print(f"Sum of even numbers: {even_total}")
20    print(f"Sum of odd numbers: {odd_total}")
21
22
```

## Output:

```

Problems Output Debug Console Terminal Ports
PS C:\Users\pc\OneDrive\Documents\AIAC\lab test> & C:/Users/pc/AppData/Local/Programs/Python/Python313/python.exe "c:/Use
rs/pc/OneDrive/Documents/AIAC/lab test/task2.py"
Input: [1, 2, 3, 4, 5, 6]
Sum of even numbers: 12
Sum of odd numbers: 9
PS C:\Users\pc\OneDrive\Documents\AIAC\lab test>

```

### Explanation:

- Function: sum\_even\_odd(numbers)
- Input: a list of integers, numbers.
- Output: a tuple (even\_sum, odd\_sum) where:
- even\_sum = sum of all even numbers in the list.
- odd\_sum = sum of all odd numbers in the list.
- How it works:
- Initializes even\_sum and odd\_sum to 0.
- Loops through each value in numbers.
- If value % 2 == 0, adds it to even\_sum, else adds it to odd\_sum.
- Returns the two sums.
- Runnable example (already in the file):
- When you run python task2.py, it uses sample\_numbers = [1, 2, 3, 4, 5, 6],
- Calls sum\_even\_odd(sample\_numbers),
- Prints:
- Sum of even numbers: 12
- Sum of odd numbers: 9
- Notes:
- Empty list → returns (0, 0).
- Works with negative integers too.
- Ensure the list contains integers to avoid errors.

### Expected Output#2

- Function
- Output execution

**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
<b>Total</b>	<b>15 Marks</b>