Week-4 Lab **Date: May 19, 2025**

Note:

1. Students will write the JAVA code for the following programming assignments and will execute through Command Prompt or Intellij IDEA IDE.
2. After successful execution, all the executed code and output needs to combine in a single PDF file to submit.
3. Submit the pdf file on Moodle link:

Problem Statements:

Case Study 1: Student Records Management System

Scenario:
A college wants to maintain digital records of students. Each student has a name, roll number, department, and CGPA. The system should allow adding new records, retrieving all records, and searching for a student by roll number. All data should be stored and retrieved from a file.

Question:
Design and implement a Java application that handles student records using file handling. Use character streams (like `FileWriter`, `BufferedWriter`, `FileReader`, `BufferedReader`) to:

- Write student details into a file.
- Read and display all student records.
- Search for a student by roll number.

Discuss how you handle file exceptions and what would happen if the file doesn't exist.

Case Study 2: Library Book Issue Tracker

Scenario:
A library maintains a file-based record of books issued to members. Each entry contains the book ID, book name, member ID, issue date, and return date.

Question:
Develop a Java program using byte streams (like `FileOutputStream`, `FileInputStream`) to:

- Add book issue records.
- Display all issue records.
- Update the return date for a specific record.

Explain your approach to reading/writing binary data and how you ensure data consistency in case of interrupted file operations.

Case Study 3: Daily Sales Logger for a Retail Store

Scenario:
A retail store logs daily sales transactions into a file. Each transaction includes item name, quantity sold, price per item, and date.

Question:
Create a Java application that:

- Appends new sales transactions to a file daily.
- Reads and summarizes total sales for a specific date.
- Handles exception like malformed entries in the file.

Demonstrate how you use `BufferedReader` and `BufferedWriter` with file append mode, and manage file access efficiently.