	School of Engineering & Technology	
	Department: SOET	**Session: 2025-2026**
	Program: B.Tech Computer Science Engineering	**Semester: III**
	Course Code:	**Number of students:**
	Course Name: - Java Programming	**Faculty: Dr Meenu**

Assignment – 3 (Unit 3)

Instructions:

- Assignment needs to be submitted by given date.
- Assignment must be submitted on (https://lms.krmangalam.edu.in/)
- Use of ChatGPT and similar tools is strictly prohibited.
- Assignment must be written on dedicated assignment copyfor Java
- Programming subject.
- All assignments must be prepared as per the format shared in classes
- Assignments need to be submitted by each individual.
- Total marks:10
- This assignment contributes to total 10% of internal evaluation.
- Submission Requirements: Submit individually via **GitHub** and provide the link of submission by **15th October,2025.**
- Assignments will be evaluated on the basis of thefollowing metrics.
- Originality
- Correctness
- Completeness

Sr.no	Assignment Details	COs
1	**Problem Statement:** Design and implement a student result management system that collects student details and their subject marks, calculates results, and displays them. The system must handle various types of errors and exceptions such as invalid marks, null values, input mismatches, and missing data. The application should demonstrate exception handling in Java, including use of built-in exceptions, custom exceptions, try- catch blocks, throw, throws, and finally clauses. **Project Objectives:** • Understand and implement Java exception handling. • Classify exceptions into **checked** and **unchecked** types. • Practice declaring and throwing exceptions using throw and throws. • Design and use custom exception classes for domain- specific validation. • Utilize try-catch-finally blocksfor robustprogram execution. **Learning Outcomes:** • Identify and handle runtime errors using exception handling. • Distinguish between checked and unchecked exceptions in Java. • Write clean, modular code thathandles exceptions gracefully. • Develop real-world Java applications using robust error- handling logic. **Project Instructions** **1. Student Class Design** **Attributes:** • rollNumber: Integer – Unique student ID. • studentName: String – Name of the student. • marks: Integer array – Stores marks for 3 subjects. **Methods:**	

- validateMarks(): Validates each subject's marks to be in range 0–100. Throws a **custom exception** if invalid.
- calculateAverage(): Calculates average marks.
- displayResult(): Displays roll number, name, marks, and result status (Pass/Fail).

2. Custom Exception Class

Class: InvalidMarksException

- Extends Exception.
- Constructor with a custom error message.

3. User Interface Class (ResultManager)

Attributes:

- Array to store multiple Student objects.
- Scanner object for input.

Methods:

- addStudent(): Accepts studentdata and validates marks. Throws InvalidMarksException.
- showStudentDetails(): Displays details for a specific student.
- mainMenu(): Provides options to perform various operations.
- Use **try-catch** to handle differentexceptions.
- Use finally to release Scanner ordisplay closing message.

4. Implementation Steps

1. Define the InvalidMarksException class.
2. Create the Student class with validation logic.
3. Create ResultManager class to manage students and handle exceptions.
4. Use arrays to store multiple Student objects.
5. Demonstrate:
    - Throwing and catching exceptions.
    - Declaring checked exceptions.
    - Usingthefinally clause.
    - Custom exception creation and usage.

Sample Interaction

===== Student Result Management System =====

1. Add Student
2. Show Student Details
3. Exit

Enter your choice: 1 Enter Roll Number: 101
Enter Student Name: Alice Enter marks forsubject 1: 85
Entermarksforsubject 2: 92
Enter marks for subject 3: 88 Student added successfully. Returning to main menu...

===== Student Result Management System =====

1. Add Student
2. Show Student Details
3. Exit

Enteryourchoice: 2
Enter Roll Number to search: 101 Roll Number: 101
Student Name: Alice Marks: 85 92 88
Average: 88.33333333333333
Result: Pass Search completed.

===== Student Result Management System =====

1. Add Student
2. Show Student Details
3. Exit

Enter your choice: 1 Enter Roll Number: 102 Enter Student Name: Bob
Entermarks forsubject 1: -10
Error: Invalid marksforsubject 1: -10 Returning to main menu...

===== Student Result Management System =====

1. Add Student
2. Show Student Details
3. Exit

Enteryourchoice: 3
Exiting program. Thank you!

Evaluation Highlight Rubrics (10 points):

Criterion	Points	Description
Correct Implementation of Exception Handling	2	- Demonstrates use of try-catch, throw, throws, and finally. - Handles both built-in and custom exceptions effectively.
Custom Exception Design and Usage	2	- InvalidMarksException class is correctly implemented and used. - Custom messages are meaningful and context-specific.
Validation Logic and Error Detection	1	- Marks are validated correctly (0–100). - Invalid inputs are detected and handled gracefully.
Functionality of Student Operations	2	- All operations (add student, show details) work as expected. - Handles multiple students using arrays.

Code Structure and Modularity	1	- Code is well-organized into classes and methods. - Follows object-oriented principles.
Input Handling and Scanner Management	1	- Uses Scanner effectively for input. - Properly closed or handled in finally block.
Code Documentation and Readability	1	- Includes meaningful comments and follows naming conventions. - Code is easy to read and maintain.