Assignment-3

Page No.
Date : / /

```
import java.util.*;

class InvalidMarksException extends Exception {
    public InvalidMarksException(String message) {
        super(message);
    }
}

class Student {
    private int rollNumber;
    private string studentName;
    private int[] marks = new int[3];

    public Student(int rollNumber, String
    studentName, int[] marks) {
        this.rollNumber = rollNumber;
        this.studentName = studentName;
        this.marks = marks;
    }

    public void validateMarks() throws InvalidMar-
    ksException {
        for(int i=0; i<marks.length; i++) {
            if(marks[i]<0 || marks[i]>100) {
                throw new InvalidMarksException(
                    "Invalid marks for subject "+
                    (i+1)+": "+marks[i]
                );
            }
        }
    }
}
```

```
public double calculateAverage() {
    int sum = 0;
    for (int m : marks) {
        sum += m;
    }
    return sum / 3.0;
}

public void displayResult() {
    System.out.println("Roll Number: " +
    rollNumber);
    System.out.println("Student Number: "
    + studentNumber);
    System.out.println("Marks: ");
    for (int m : marks) {
        System.out.print(m + " ");
    }
    System.out.println();
    double avg = calculateAverage();
    System.out.println("Average: "
    + avg);
    if (avg >= 40)
        System.out.println("Result: Pass");
    else
        System.out.println("Result: Fail");
}
public int getRollNumber() {
    return rollNumber;
}
}
```

```
public class ResultManager {
    private Student[] students = new Student
    [50];
    private int count = 0;
    private Scanner sc = new Scanner (System.in);

    public void addStudent() {
        try {
            System.out.print("Enter Roll Number
            : ");
            int roll = sc.nextInt();
            sc.nextLine();

            System.out.print("Enter Student Name: ")
            String name = sc.nextLine();

            int[] marks = new int[3];
            for(int i=0; i<3; i++) {
                System.out.print("Enter marks for
                subject" + (i+1) + ": ");
                marks[i] = sc.nextInt();
            }
            Student s = new Student(roll, name,
            marks);
            s.validateMarks();
            students[count++] = s;
            System.out.println("Student added
            Successfully.");
```

```
} catch (InvalidMarksException e) {
    System.out.println("Error: " +
    e.getMessage());
} catch (InputMismatchException e) {
    System.out.println("Error: Invalid
    input type! ");
    sc.nextLine();
} catch (Exception e) {
    System.out.println("Unexpected
    error : " + e.getMessage());
} finally {
    System.out.println("Returning
    to main menu.... ");
}
}
public void showStudentDetails() {
    try {
        System.out.print("Enter Roll
        Number to search: ");
        int roll = sc.nextInt();
        boolean found = false;
        for (int i = 0; i < count; i++) {
            if (students[i].getRollNumber()
            == roll) {
                students[i].displayResult();
                found = true;
                break;
            }
        }
```

```
            if(!found)
                System.out.println("Student not
                found!");
        }catch (InputMismatchException e){
            System.out.println("Error: Invalid
            input!");
            sc.nextLine();
        }finally{
            System.out.println("Search completed.");
        }
    }
    public void mainMenu(){
        int choice = 0;
        do{
            System.out.println("\n===== Student
            Result Management System =====");
            System.out.println("1. Add Student");
            System.out.println("2. Show Student
            Details");
            System.out.println("3. Exit");
            System.out.print("Enter your choice
            : ");

            try{
                choice = sc.nextInt();
                switch(choice){
                    case 1:
                        addStudent();
                        break;
```

```
            case 2:
                showStudentDetails();
                break;
            case 3:
                System.out.println("
                Exiting program. Thank
                you!");
                break;
            default:
                System.out.println
                ("Invalid choice!");
        }
    }catch (InputMismatchException e)
    {
        System.out.println("Error:
        Enter a valid number!");
        sc.nextLine();
    }
}while (choice != 3);
sc.close();
}
public static void main (String[] args)
{
    ResultManager rm = new ResultManager
    ();
    rm.mainmenu();
}
}
```