

|  |
| --- |
| Experiment No. 12 |
| Course Project based on the content of the syllabus. |
| Name : Parth Sadanand Gawad |
| Roll No : **68** |
| Date of Performance: |
| Date of Submission: |



**Code :**

import javax.swing.\*; import java.awt.\*;

import java.awt.event.ActionEvent; import java.awt.event.ActionListener; import java.util.ArrayList;

public class StudentGradingSystem extends JFrame { private JTextField nameField;

private JTextField gradeField1, gradeField2, gradeField3; private JTextArea resultArea;

private ArrayList<String> studentRecords;

public StudentGradingSystem() { setTitle("Student Grading System"); setSize(400, 350);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); setLayout(new FlowLayout());

studentRecords = new ArrayList<>();

// Name label and field

add(new JLabel("Student Name:")); nameField = new JTextField(20); add(nameField);

// Grade labels and fields for 3 subjects add(new JLabel("Marks (Subject 1):")); gradeField1 = new JTextField(5); add(gradeField1);

add(new JLabel("Marks (Subject 2):")); gradeField2 = new JTextField(5); add(gradeField2);

add(new JLabel("Marks (Subject 3):")); gradeField3 = new JTextField(5); add(gradeField3);



// Calculate button

JButton calculateButton = new JButton("Calculate Grades"); add(calculateButton);

// Clear button

JButton clearButton = new JButton("Clear"); add(clearButton);

// Result area

resultArea = new JTextArea(5, 30); resultArea.setEditable(false); add(new JScrollPane(resultArea));

// Button action to calculate grades calculateButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) { calculateGrades();

}

});

// Button action to clear fields clearButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) { clearFields();

}

});

}

private void calculateGrades() { String name = nameField.getText();

String gradesInput = gradeField1.getText() + " " + gradeField2.getText() + " " + gradeField3.getText();

StringBuilder result = new StringBuilder(name + ", your grades are:\n");

try {

double grade1 = Double.parseDouble(gradeField1.getText()); double grade2 = Double.parseDouble(gradeField2.getText()); double grade3 = Double.parseDouble(gradeField3.getText());

// Store student record

studentRecords.add(name + " - Grades: " + gradesInput + " | Letter Grades: " +

getLetterGrade(grade1) + "\n " + getLetterGrade(grade2) + "\n " + getLetterGrade(grade3));

// Add letter grades to result result.append(getLetterGrade(grade1)).append("")

.append(getLetterGrade(grade2)).append("")

.append(getLetterGrade(grade3)).append("");

resultArea.setText(result.toString());

} catch (NumberFormatException e) { resultArea.setText("Please enter valid numeric grades.");

}

}

private void clearFields() { nameField.setText(""); gradeField1.setText(""); gradeField2.setText(""); gradeField3.setText(""); resultArea.setText("");

}

private String getLetterGrade(double grade) { if (grade >= 90) {

return "A\n";

} else if (grade >= 75) { return "B\n";

} else if (grade >= 60) { return "C\n";

} else if (grade >= 35) { return "D\n";

} else {

return "F\n";

}

}

public static void main(String[] args) {



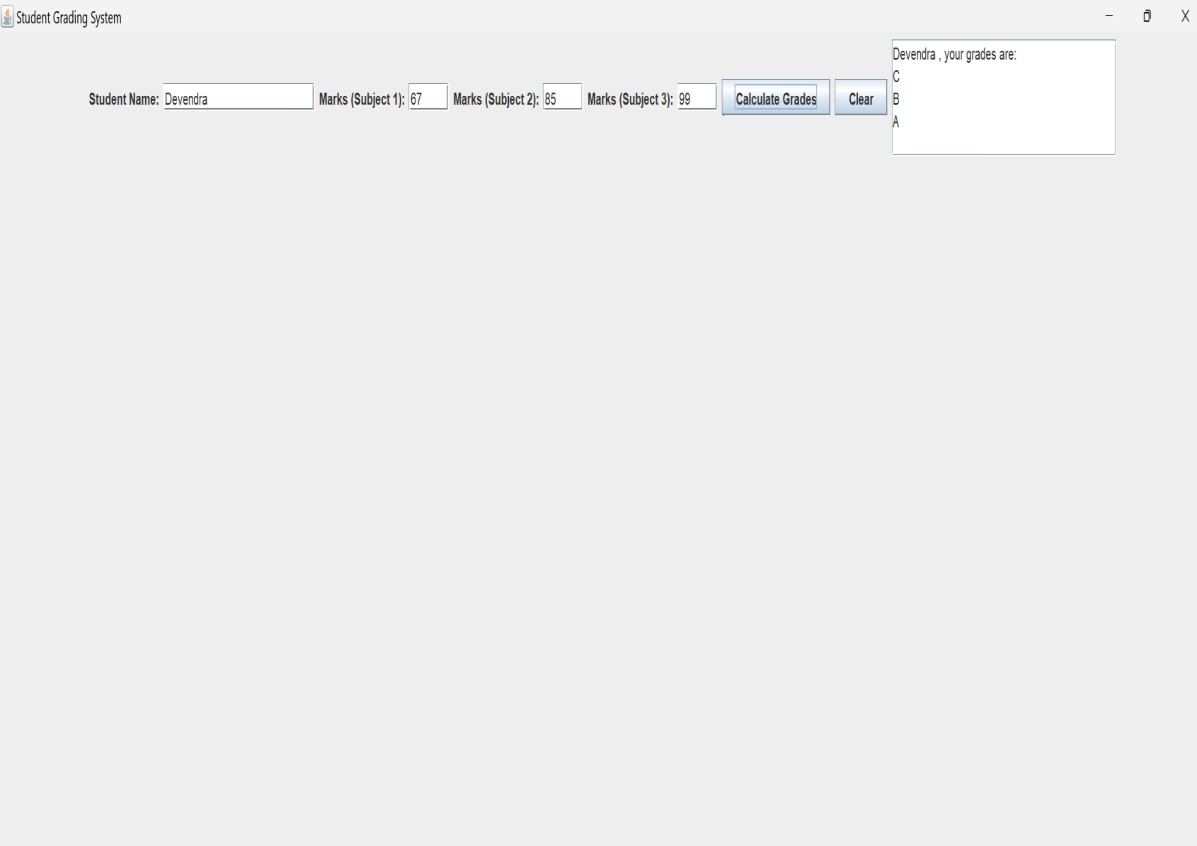
SwingUtilities.invokeLater(() -> { StudentGradingSystem gradingSystem = new

StudentGradingSystem(); gradingSystem.setVisible(true);

});

}

}

Output: