Practical 3(stud result GUI)

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class StudentResultGUI extends JFrame {

private static final String[] SUBJECT\_NAMES = {"Subject 1", "Subject 2", "Subject 3",

"Subject 4", "Subject 5"};

private final JTextField[] subjectMarksFields;

private final JButton calculateButton, clearButton;

private final JFrame resultFrame;

public StudentResultGUI() {

// Main window for entering marks

setTitle("Student Result");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new BorderLayout());

// Menu bar

JMenuBar menuBar = new JMenuBar();

JMenu fileMenu = new JMenu("File");

JMenuItem exitMenuItem = new JMenuItem("Exit");

exitMenuItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

System.exit(0);

}

});

fileMenu.add(exitMenuItem);

menuBar.add(fileMenu);

setJMenuBar(menuBar);

JPanel centerPanel = new JPanel();

centerPanel.setLayout(new GridLayout(SUBJECT\_NAMES.length, 2, 5, 5));

JLabel[] subjectLabels = new JLabel[SUBJECT\_NAMES.length];

subjectMarksFields = new JTextField[SUBJECT\_NAMES.length];

for (int i = 0; i < SUBJECT\_NAMES.length; i++) {

subjectLabels[i] = new JLabel(SUBJECT\_NAMES[i]);

subjectMarksFields[i] = new JTextField(10);

centerPanel.add(subjectLabels[i]);

centerPanel.add(subjectMarksFields[i]);

}

add(centerPanel, BorderLayout.CENTER);

// Button Panel

JPanel buttonPanel = new JPanel(new FlowLayout(FlowLayout.CENTER));

calculateButton = new JButton("Calculate Result");

calculateButton.addActionListener(new CalculateButtonListener());

buttonPanel.add(calculateButton);

clearButton = new JButton("Clear");

clearButton.addActionListener(new ClearButtonListener());

buttonPanel.add(clearButton);

add(buttonPanel, BorderLayout.SOUTH);

// Result window (initially hidden)

resultFrame = new JFrame("Student Result");

resultFrame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

resultFrame.setLayout(new FlowLayout());

resultFrame.setSize(300, 150);

resultFrame.setLocationRelativeTo(null); // Center the window

resultFrame.setVisible(false);

pack();

setLocationRelativeTo(null); // Center the window

setVisible(true);

}

private class CalculateButtonListener implements ActionListener {

@Override

public void actionPerformed(ActionEvent e) {

try {

int[] marks = new int[SUBJECT\_NAMES.length];

for (int i = 0; i < SUBJECT\_NAMES.length; i++) {

marks[i] = Integer.parseInt(subjectMarksFields[i].getText());

}

int totalMarks = 0;

double averageMarks = 0.0;

String grade = "";

for (int mark : marks) {

totalMarks += mark;

}

averageMarks = (double) totalMarks / marks.length;

if (averageMarks >= 60) {

grade = "Pass";

} else {

grade = "Fail";

}

// Display result in separate window

JLabel totalLabel = new JLabel("Total Marks: " + totalMarks);

JLabel averageLabel = new JLabel("Average Marks: " + String.format("%.2f",

averageMarks));

JLabel gradeLabel = new JLabel("Grade: " + grade);

resultFrame.getContentPane().removeAll(); // Clear previous results before adding new ones

resultFrame.add(totalLabel);

resultFrame.add(averageLabel);

resultFrame.add(gradeLabel);

resultFrame.pack();

resultFrame.setVisible(true);

} catch (NumberFormatException ex) {

JOptionPane.showMessageDialog(null, "Please enter valid integer marks for all subjects.", "Invalid Input", JOptionPane.ERROR\_MESSAGE);

}

}

}

private class ClearButtonListener implements ActionListener {

@Override

public void actionPerformed(ActionEvent e) {

for (JTextField field : subjectMarksFields) {

field.setText("");

}

resultFrame.setVisible(false); // Hide any previously displayed results

}

}

public static void main(String[] args) {

SwingUtilities.invokeLater(StudentResultGUI::new);

}

}