

Lab Assignment: GUI Development with Swing

Objective:

Learn to create interactive graphical user interfaces using Swing and understand event handling.

Requirements:

1. Create a Swing application with:
 - A JFrame with the title "Student Information System".
 - A JTextField to accept a student name.
 - A JComboBox to select a course from a predefined list.
 - A JButton to save the student details.
 - A JTextArea to display saved student details.
2. Add an event listener for the button to save and display student details in the text area.
3. Implement basic validation to ensure the text field is not empty.

Code Starter:

```
// StudentInfoApp.java
import javax.swing.*;
import java.awt.event.*;
import java.util.ArrayList;

public class StudentInfoApp {
    private JFrame frame;
    private JTextField nameField;
    private JComboBox<String> courseBox;
    private JTextArea detailsArea;
    private ArrayList<String> studentDetails;

    public StudentInfoApp() {
        studentDetails = new ArrayList<>();
        frame = new JFrame("Student Information System");

        nameField = new JTextField();
        courseBox = new JComboBox<>(new String[]
            { "Math", "Science", "History" });
        detailsArea = new JTextArea();
        JButton saveButton = new JButton("Save");

        // TODO: Add components to frame and handle events
    }

    public static void main(String[] args) {
        new StudentInfoApp();
    }
}
```

```
    }  
}
```

Exercises:

1. Add a `JMenuBar` with options like `File -> Exit` and `Help -> About`.
2. Use a `JTable` to display student details instead of a `JTextArea`.
3. Add a feature to clear all fields when a "Clear" button is clicked.

Bonus Tasks:

1. Implement data persistence by saving and loading student details to/from a file.
2. Create a "Search" feature to find student details by name.
3. Use a custom layout manager (e.g., `GridBagLayout`) for better UI organization.

Submission:

Submit your code and screenshots of your application.