GUI Programming 2024 to 2025 – Year 2

Labwork 2: (10% overall or 100 points of 600 points for labwork this semester)

Topic: Beginning SWING: Fields, JComboBox, JLists, RadioButton, Layout Managers

IMPORTANT NOTES:

- **NO COPYING PERMITTED AND ZERO MARKS WILL APPLY TO COPIED WORK. FURTHER ACTION MAY BE TAKEN AGAINST STUDENTS THAT HAVE BEEN FOUND TO COPY WORK.**

- **ASSESSMENT WILL INVOLVE ONE-TO-ONE QUESTIONS ABOUT YOUR SUBMITTED WORK. A COMPLETED SELF-ASSESSMENT SHEET WILL BE USED TO GUIDE THE ASSESSMENT. USE COMMENTS IN YOUR CODE TO ENSURE YOU DON'T FORGET WHY YOU WROTE CODE YOU MAY LATER BE ASKED ABOUT.**

- **ALL WORK MUST BE SUBMITTED TO MOODLE BY DATES SPECIFIED (SUBMISSION DEADLINES WILL BE POSTED ON MOODLE).**

- **MANY OF THE TASKS ASSIGNED BELOW CAN BE COMPLEX AND\OR THE DESCRIPTIONS MAY REQUIRE FURTHER CLARIFICATIONS. PLEASE USE THE AVAILABLE LAB TIMES TO ASK FOR CLARIFICATIONS AND ADVICE\HINTS ON THE TASKS BELOW.**

- **YOU CAN USE A SIMPLE JAVA ENABLED TEXT EDITOR IF YOU WISH, e.g., TEXTPAD or NOTEPAD. HOWEVER, I SUPPORT THE MOVING ON TO A MORE ADVANCED IDE AT THIS POINT ALSO (e.g., Eclipse or IntelliJ or NetBeans).**

- **CHATGPT and other similar AI tools that can code simple solutions are not permitted. THEY DO NOT TEACH YOU HOW TO BECOME A GOOD PROGRAMMER EITHER!**

Part 1 – Create a JFrame with fields and password fields (20 points)

Create a Java class called **Lab2Part1**. Make the class a JFrame and make it visible and set the size to a medium size frame (you choose the exact dimensions). Add a panel to the JFrame content pane so that it contains a club member details form made up of **JLabels, JTextFields and JPasswordFields** (a form like when you are joining a club). Include information like, first name, surname, city, age etc. to gather information about the joining club member. Use JLabels to the left of the input field to indicate where the information needs to be typed (in the field just next to it). Include an 'Enter Password' field and 'Verify Password' field in the form (note: this should use JPasswordField). All of the components should be laid out neatly in a JPanel on the JFrame (hint: use an appropriate layout manager set for the panel).

- Create the JFrame and settings   (2 points)
- Create the JPanel   (2 points)
- Set a good layout manager on JPanel   (2 points)
- Create at least 5 labels for the fields   (5 points)
- Create at least 5 input fields for the member info (5 points)
- Create the password field labels   (2 points)
- Create the password fields (enter and verify)   (2 points)

Part 2 – JList (20 points)

Create a class called **Lab2Part2**. Create a JFrame that holds TWO **JList's.** The first JList allows the user to select from a list of all the provinces of Ireland (find out the provinces somewhere if you don't know them!) and the second JList allows the user to select from a list of Ireland's four cities or towns of your choice (find this info somewhere also if you don't know four towns\citites). Place the two JLists in the same JPanel on the <u>bottom</u> of the JFrame so that they are <u>side-by-side</u>.

- Create the frame   (2 points)
- Create the first JList with complete String info (provinces)   (6 points)
- Create the second JList with complete String info (cities)   (6 points)
- Create and add both JList to panel   (3 points)
- Layout the JLists as described (at bottom and side-by-side)   (3 points)

Part 3 – JComboBox and JRadioButtons in same interface (30 points)

Create a class called **Lab2Part3**. Create a JFrame that has TWO JComboBox's at the top of the frame in a JPanel (or you can use more than one panel to hold the ComboBoxes if you think you need more). Display the two JComboBox's **one on top of the other** at the **top** of the JFrame. The JComboBox's must be filled with at least five Strings each - but **you choose** what each ComboBox displays in the dropdown menu (they could be two lists of favourite places or food or songs or musicians or sports or anything else appropriate).
Create TWO JRadioButtons on the bottom of the JFrame (a second panel could be needed for this). The buttons with state "Top Combo" or "Bottom Combo". Make it so that **only ONE** of the radio buttons can be selected at a time (hint: use a *ButtonGroup*). Add a label above the radio buttons which states "Please select which ComboBox you like the most from the two shown above". Make the Font in the label big and bold (you may choose how large the Font will be).

- Create the frame   (2 points)
- Create the first JComboBox complete with String info   (5 points)
- Create the second JComboBox complete with String info   (5 points)
- Add the JComboBox's to the top of frame one-on-top-of-the-other (5 points)
- TWO JRadioButtons complete with text added to bottom of frame   (5 points)
- Add the label with the big and bold font above the buttons   (3 points)
- Make the radio buttons so only ONE can be selected   (5 points)

Part 4 – Advanced Layout - Maybe draw out a plan\sketch first! (30 points)

Create a class called **Lab2Part4**. Create a JFrame that models the ATM machine menu options using buttons (JButton) and labels (JLabel) only (panel(s) will be needed also). The GUI should contain all of the usual menu options you see at your ATM (minimum 6 menu options with a button to click; you should use labels to help the user use the system properly, e.g., "Please select option" etc.). In this lab section use a **Layout manager** to achieve the layout of the GUI. Use images and change background colours in the GUI buttons\frame to the GUI to make it look interesting\appealing (use the Oracle website to find out options for changes in colour to panel backgrounds). Note: The objective this week is to try to make the GUI <u>look good</u> but will not actually do anything yet (no balance returned etc.)!!!

- Create the JFrame, set appropriate size and set title (with Fonts)   (4 points)
- Create the buttons for the menu options (6 minimum)   (6 points)
- Use well positioned labels to instruct\helps the user (titles etc.)   (5 points)
- Use panel(s) to group components in layout   (5 points)
- Make GUI look unique (use your imagination. e.g., colour\images) (5 points)
- Use of well selected Layout manager(s) (Border/Flow/Grid)   (5 points)