CCCS214: Object-Oriented Programming II

Lab 5: Advanced GUI Application

OOP2 – Fall 2022

Lab Objectives

- Be able to add List, Combo box and a menu to a menu bar
- Be able to add menu items to a menu
- Be able to write event handlers for list, Combo box and menu items
- Be able to deal with font, colour and image.

1- List

List displays a series of items from which the user may select one or more items. **JList** class is used to create a list with multiple values, allowing a user to select any of these values. When a value is selected from JList, a **ListSelectionEvent** is generated, which is handled by implementing **ListSelectionListener** interface.

Month Selection Example:

This example was discussed during the lectures. Try to implement it on netbeans so you can practice what you have learned practically.

```
import javax.swing.*;
import javax.swing.event.*;
import java.awt.*;

/**
This class demonstrates the List Component.
*/

public class ListWindowWithScroll extends JFrame
{
private JPanel monthPanel; // To hold components
private JPanel selectedMonthPanel; // To hold components
private JList monthList; // The months
private JScrollPane scrollPane; // A scroll pane
private JTextField selectedMonth; // The selected month
private JLabel label; // A message

// The following array holds the values that will
// be displayed in the monthList list component.
private String[] months = { "January", "February",
"March", "April", "May", "June", "July",
"August", "September", "October", "November",
"December" };

/**
Constructor
```

```
*/

 public ListWindowWithScroll()
 {
 // Set the title.
 setTitle("List Demo");

 // Specify an action for the close button.
 setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

 // Add a BorderLayout manager.
 setLayout(new BorderLayout());

 // Build the month and selectedMonth panels.
 buildMonthPanel();
 buildSelectedMonthPanel();

 // Add the panels to the content pane.
 add(monthPanel, BorderLayout.CENTER);

add(selectedMonthPanel, BorderLayout.SOUTH);

 // Pack and display the window.
 pack();
 setVisible(true);
 }

 /**
The buildMonthPanel method adds a list containing
the names of the months to a panel.
*/

 private void buildMonthPanel()
 {
 // Create a panel to hold the list.
 monthPanel = new JPanel();

 // Create the list.
 monthList = new JList(months);

 // Set the selection mode to single selection.
 monthList.setSelectionMode(
 ListSelectionModel.SINGLE_SELECTION);

 // Register the list selection listener.
 monthList.addListSelectionListener(
 new ListListener());

 // Set the number of visible rows to 6.
 monthList.setVisibleRowCount(6);

 // Add the list to a scroll pane.
 scrollPane = new JScrollPane(monthList);

 // Add the scroll pane to the panel.
 monthPanel.add(scrollPane);
 }

 /**
The buildSelectedMonthPanel method adds an
uneditable text field to a panel.
*/

 private void buildSelectedMonthPanel()
 {
 // Create a panel to hold the text field.
 selectedMonthPanel = new JPanel();

// Create the label.
 label = new JLabel("You selected: ");

// Create the text field.
 selectedMonth = new JTextField(10);

 // Make the text field uneditable.
 selectedMonth.setEditable(false);

 // Add the label and text field to the panel.
 selectedMonthPanel.add(label);
selectedMonthPanel.add(selectedMonth);
 }

 /**
Private inner class that handles the event when
the user selects an item from the list.
*/

 private class ListListener
 implements ListSelectionListener
 {
```

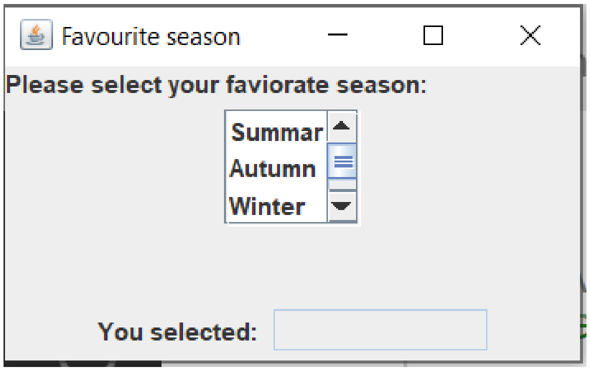
```
 public void valueChanged(ListSelectionEvent e)
 {
// Get the selected month.
 String selection =
 (String) monthList.getSelectedValue();

 // Put the selected month in the text field.
 selectedMonth.setText(selection);
 }
 }

 /**
 The main method creates an instance of the
 ListWindowWithScroll class which causes it
 to display its window.
 */

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

2- Combo Box

A combo box presents a list of items that the user may select from. Unlike a list component, a combo box presents its items in a drop-down list. **JComboBox** class which is in the **javax.swing package** is used to create a combo box. When a value is selected from **combo box**, an **Action Event** is generated, which is handled by implementing **actionListener** interface.

Task 1: Create the following GUI:

The application display the list of the seasons and ask the user to select the favourite season from the list and display it in the jLabel.



Please select your faviorate season:

You selected:

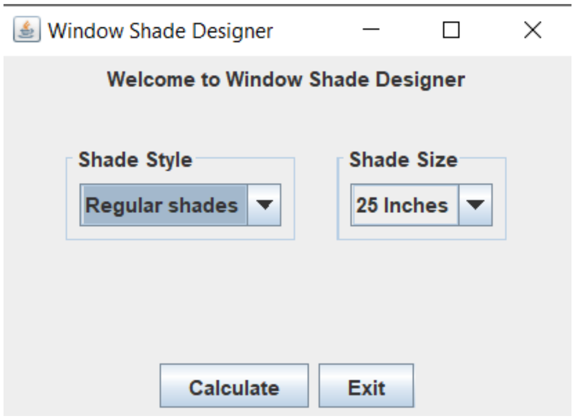
Task 2:

Create a window shade designer application that allows the user to select the style and size of window shade from combo boxes and the total charges should be displayed.

A custom window shade designer charges a base fee of $50 per shade. In addition, charges are added for certain styles and sizes as follows:

Styles:
Regular shades: Add $0
Folding shades: Add $10
Roman shades: Add $15

Sizes:
25 inches wide: Add $0
27 inches wide: Add $2
32 inches wide: Add $4



Welcome to Window Shade Designer

Shade Style: Regular shades

Shade Size: 25 Inches