A close-up of logos

Description automatically generated

**BITP 3113**

**OBJECT ORIENTED PROGRAMMING**

**LAB:**

**WEEK 11 – JAVA GUI (SECTION 1)**

**LECTURER NAME:**

**DR EMALIANA BINTI KASMURI**

**STUDENT NAME:**

**MUHAMMAD AFIQ MUHAIMIN BIN MOHD ZAINI**

**MATRIC NUMBER:**

**B032410648**

**SEMESTER: 2-2024/25**

# Section 2: Understanding Common Java GUI Components

1. Download the **Week 11 - Section 02 - Supplementary Codes.zip** file from ulearn.
2. Unzip the file.
3. Execute the MainGUIApp.java. The initial output shall be similar as shown in [Figure](#_bookmark5) [1](#_bookmark5).

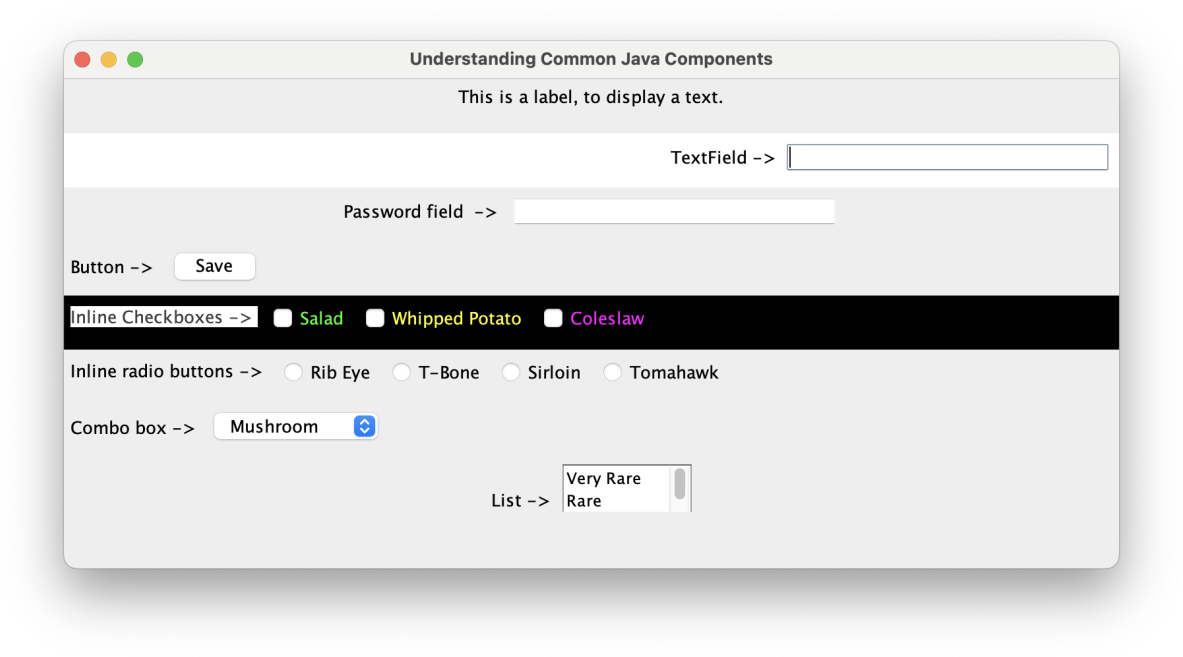


Figure 4: Common GUI components for Java desktop application

## Exercise 5: GUI Component Identification

1. Open the **ComponentScreen.java** file.
2. Observe the code and the output from the execution.
3. Identify the object and its constructor for each of the GUI components loaded on the screen from MainGUIApp.java execution. Record the answer in [Table 2](#_bookmark14).

Table 2: GUI component observation results

|  |  |  |
| --- | --- | --- |
| **GUI Component** | **Object** | **Constructor** |
| JLabel | Label | new JLabel(String text) |
| JCheckBox | cbWhippedPotato | new JCheckBox(“Whipped Potato”) |
| JRadioButton | rbTomahawk | new JRadioButton(“Tomahawk”) |
| JList<String> | lstDoneness | new JList<String>(doneness) |

## Exercise 6: Overriding GUI Component Color

1. **Describe how should a programmer change the color for the GUI components.**

Programmer can change the color of the GUI using the method ***setBackground(Color color)* to change the background colorand *setForeground(Color color)* to change the text or foreground color**

1. Based on your description, change the default color for one of the GUI components to the color of your choice.

A screen shot of a computer program

AI-generated content may be incorrect.

Output

A green rectangle with white text

AI-generated content may be incorrect.

## Exercise 7: Relocate Initial Screen Display

1. Amend the class to display the screen in the center of the screen.
2. Highlight the code to show your solution.

A screen shot of a computer

AI-generated content may be incorrect.

## Exercise 8: Layout Manager Identification

1. Identify the classes that contains the term Layout from the code.
2. These are known as layout manager classes. A layout manager class arrange the GUI components layout on a panel or frame. List the layout object and classes in the in [Table 3](#_bookmark18).

Table 3: Layout manager class observation results

|  |  |  |
| --- | --- | --- |
| **Object** | **Package** | **Constructor** |
| flowLayout | java.awt | new FlowLayout() |
| gridLayout | java.awt | new GridLayour(9, 1) |
| borderLayout | java.awt | new BorderLayout() |

**Exercise G: Observing Layout Manager Behavior**

The purpose of layout manager is to hold a group of GUI components and arrange it according to layout policy design for a particular manager class.

1. Observe the arrangement of GUI components in each panel.
2. Record your observation in [Table 4](#_bookmark20).

Table 4: Panel arrangement observation results

|  |  |
| --- | --- |
| **Panel** | **Observation** |
| **Label panel** | Contains a single label centered horizontally with the sentence “This is a label, to display a text” using *FlowLayout* |
| **Text field panel** | The label for the text field and the box itself is aligned to the right using *FlowLayout.RIGHT* |
| **Password field panel** | Label and password field arranged in centered format which is default |
| **Button panel** | The label “Button ->” and the Save button is aligned to the left using FlowLayout.LEFT |

|  |  |
| --- | --- |
| **Check boxes panel** | Label “Inline Checkboxes ->” and the button (Salad, Whipped Potato, Coleslaw) are aligned to the left with custom color for the text |
| **Radio buttons panel** | Label “Inline radio buttons ->” and the button (Rib eye, T-Bone, Sirloin, Tomahawk) are arranged horizontally and aligned to the left |
| **Combo box panel** | Label “Combo box ->” and the list (Mushroom, Black Pepper , Barbeque) inside the dropdown are placed side by side using default layout |
| **List panel** | Label “List ->” and the list (Very Rare, Rare, Medium Rare, Medium, Medium Well, Well Done) inside the dropdown are placed side by side using default layout |

## 

## Exercise 10: Concluding Layout Behavior

Draw some conclusions based on the observation results in Exercise 9. Record your observation in ulearn.

* **Use of FlowLayout**
  + Most of the panels use FlowLayout as their guideline for arrangement of components. This allows the layout manager to align components in a single row or left/center/right to suit the design of each panel.
* Custom Alignment will enhance readability
  + Panels that are aligned with the ***FlowLayout.RIGHT***for the **Text Field Panel** and ***FlowLayout.LEFT*** for the **Button Panel** shows that the different alignments of the button can help improve the logical grouping and visual clarity.
* Grouping Similar Components:
  + The use of **ButtonGroup** for radio buttons ensures that only one option can be selected at a time and placing them inline (horizontally) improves space utilization and user experience.
* Scrollable Lists Improve Usability:
  + The List Panel uses a **JScrollPane**, which allows users to scroll through options, making it suitable for data with multiple entries.
* Panel-specific Styling:
  + Some panels, like the **Check Boxes Panel**, apply custom foreground and background colors, enhancing visual appeal and distinguishing between component groups.

### End of Document