BITP 3113
OBJECT ORIENTED PROGRAMMING

LAB:
WEEK 11 – JAVA GUI (SECTION 2)

LECTURER NAME:
DR EMALIANA BINTI KASMURI

STUDENT NAME:
MUHAMMAD AFIQ MUHAIMIN BIN MOHD ZAINI

MATRIC NUMBER:
B032410648

SEMESTER: 2-2024/25

Section 3: Distance Converter Application Development

This section will develop an application called Distance Converter Application. The application is a single-screen application that receives in a distance input in kilometer and imperial conversion unit from the user. The initial look of the application is shown in Figure 5.

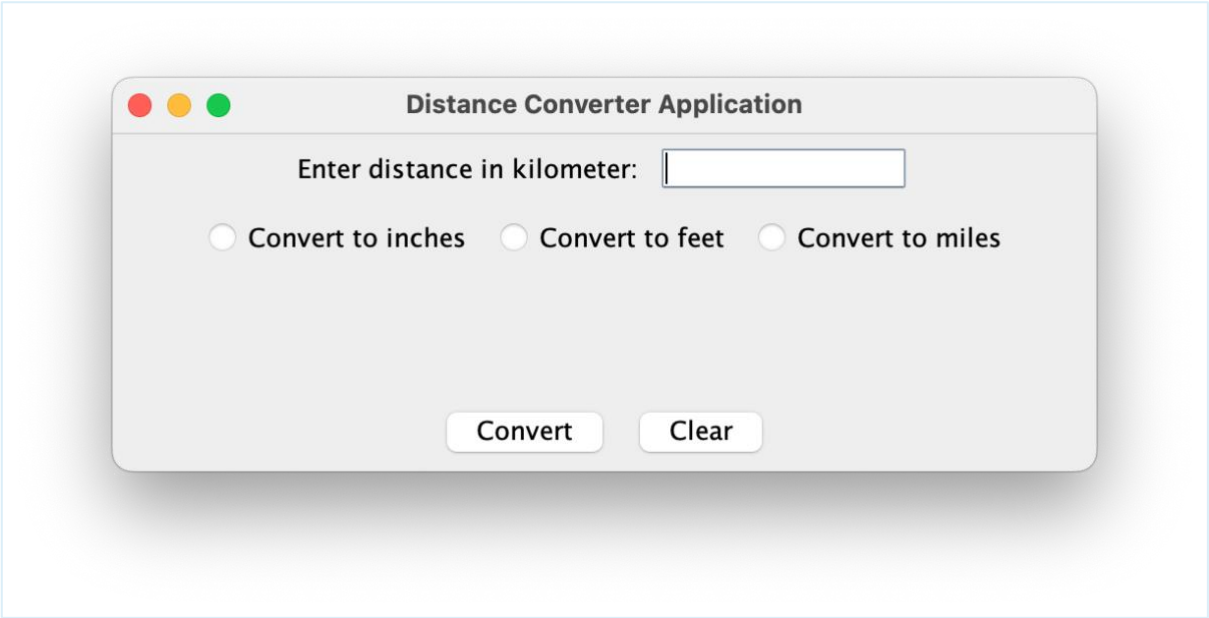


Figure 5: Distance Convertor Application window

Exercise 11: Java Distance Converter Window Definition

1. Create a new window named **Distance Converter Application** using the knowledge and skill comprehended from the previous exercises.
2. The window should be launched from a class named `ConverterApp`.

Exercise 12: User Input Panel Definition

The application consists of three panels. The first panel contains a label and a text field.

1. Define and construct a private object of `JTextField` named `txtDistance` as member of the class defined in Exercise 4. This object will be displayed in the window defined in that class.
2. Create a private method named `getUserInputPanel()`. This method shall return an object of `JPanel`.
3. Provide the implementation of the method using the Java code shown in Figure 6.

```
// A panel to hold GUI component
JPanel userInputPanel = new JPanel();
```

```
// Define label and add to panel
JLabel lblDistance = new JLabel("Enter distance in kilometer:");
userInputPanel.add(lblDistance);

// Set text field initial column size and add  to panel
txtDistance.setColumns(10);
userInputPanel.add(txtDistance);

return userInputPanel;
```

Figure 6: Java code for the method getUserInputPanel ( ) implementation

4. Provide an appropriate comment for method.
5. Add the code snippets shown in Figure 7 before the window is visible.

```
// Add panel to the NORTH region
JPanel userInputPanel = getUserInputPanel();
super.add(userInputPanel);
```

Figure 7: Code snippet to add user input panel to the window

6. Import the necessary classes.
7. Fix any errors.
8. Save the class.
9. Execute the window from class `ConverterApp` defined in the previous exercise. The output shall be similar as shown in Figure 8.

Figure 8: Expected output from Exercise 5

Exercise 13: User Selection Panel Definition

Using the knowledge and skill comprehended in from the previous exercises, create a panel to lay three (3) radio buttons as shown in Figure 5, using `JRadioButtons`. Group the radio buttons using `ButtonGroup` class.

The solution must be neat, readable, easy to read and comply to object-oriented principles.

Exercise 14: Button Panel

Using the knowledge and skill comprehended in from the previous exercises, create a panel to lay two (2) buttons as shown in Figure 5 using `JButtons`.

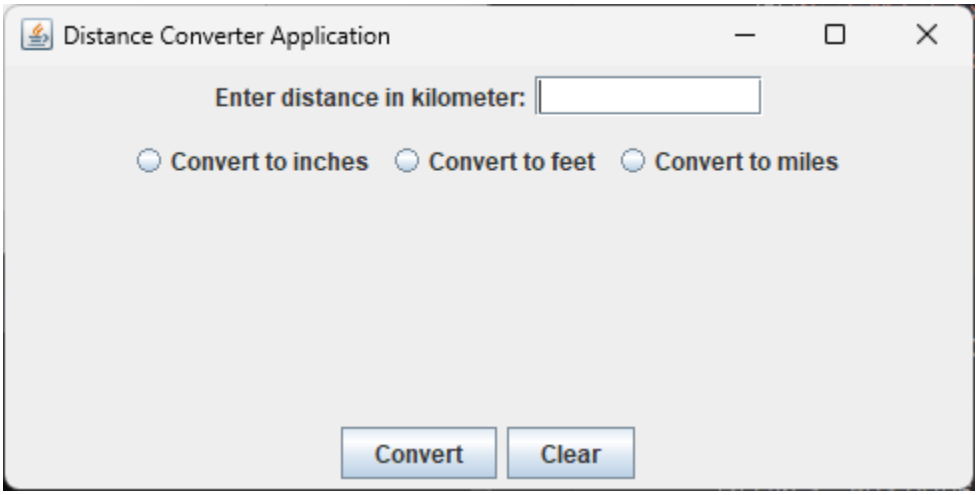
The solution must be neat, readable, easy to read and comply to object -oriented principles.

Exercise 15: Screen Launcher

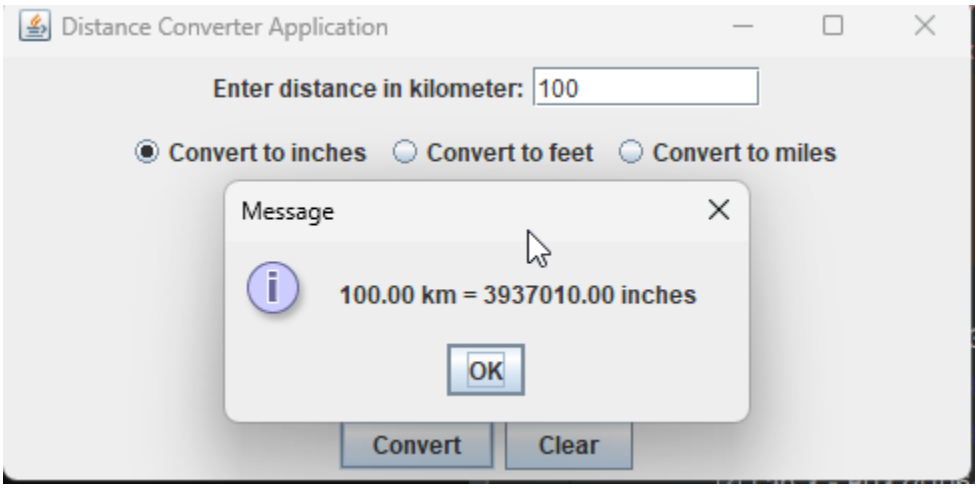
Create a class to display the screen created from the previous exercises in this section using the knowledge and skill comprehended in from the previous exercises. The solution must be neat, readable, easy to read and comply with object -oriented principles.

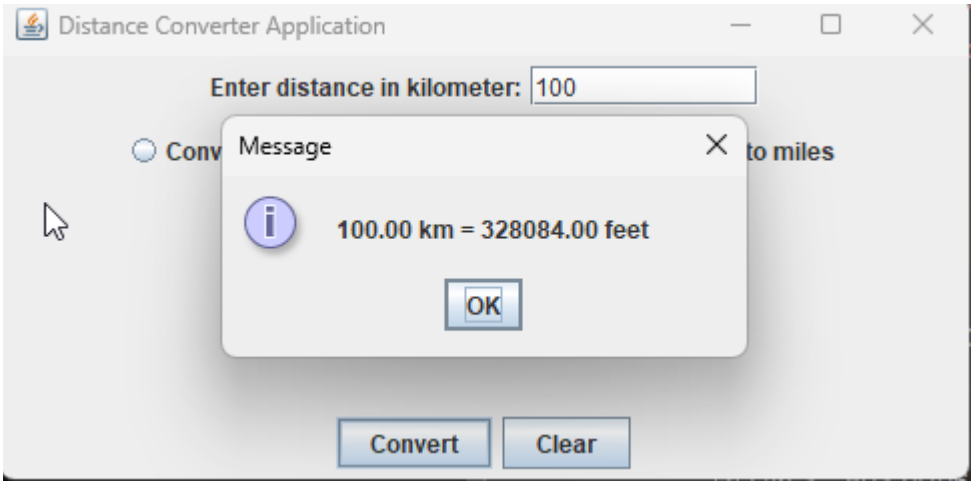
Submit your solutions and screen shot output in ulearn.

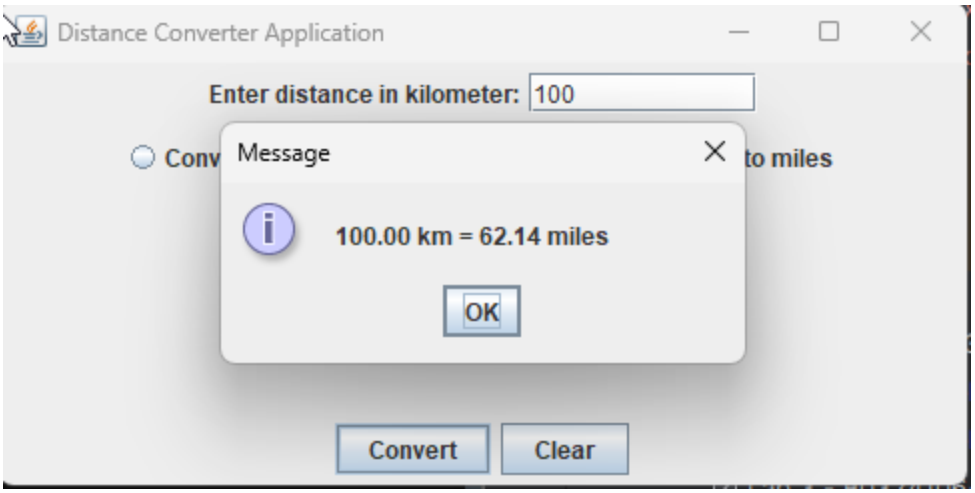
Interface



Convert 100km to each of the units







Code Snippet

ConverterApp.java

```
/**
 * This class displays the GUI for java distance converter app
 *
 * @author Muhammad Afiq Muhaimin Bin Mohd Zaini, FTMK
 */
public class ConverterApp {
    public static void main(String[] args) {
        //Launch the GUI
        new DistanceConverter();
    }
}
```

DistanceConverter.java

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

/**
 * DistanceConverter class creates the main GUI window
 * for converting kilometers into inches, feet, or miles.
 *
 * Author: Muhammad Afiq Muhaimin Bin Mohd Zaini, FTMK
 */
public class DistanceConverter extends JFrame {

    // GUI components
    private JTextField txtDistance;
    private JRadioButton rbInches, rbFeet, rbMiles;
    private JButton btnConvert, btnClear;

    // Constructor
    public DistanceConverter() {
        setTitle("Distance Converter Application");
        setSize(500, 250);
```

```
        setLayout(new BorderLayout());

        // Add panels
        add(getUserInputPanel(), BorderLayout.NORTH);
        add(getConversionOptionPanel(), BorderLayout.CENTER);
        add(getButtonPanel(), BorderLayout.SOUTH);

        setLocationRelativeTo(null); // Center window
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true); // Show window
    }

    /**
     * Creates a panel with label and input text field.
     */
    private JPanel getUserInputPanel() {
        txtDistance = new JTextField(10);
        JLabel lblDistance = new JLabel("Enter distance in kilometer:");

        JPanel panel = new JPanel();
        panel.add(lblDistance);
        panel.add(txtDistance);

        return panel;
    }

    /**
     * Creates a panel with three radio buttons for unit selection.
     */
    private JPanel getConversionOptionPanel() {
        rbInches = new JRadioButton("Convert to inches");
        rbFeet = new JRadioButton("Convert to feet");
        rbMiles = new JRadioButton("Convert to miles");

        ButtonGroup group = new ButtonGroup();
        group.add(rbInches);
        group.add(rbFeet);
        group.add(rbMiles);

        JPanel panel = new JPanel();
        panel.add(rbInches);
        panel.add(rbFeet);
        panel.add(rbMiles);

        return panel;
    }

    /**
     * Creates a panel with Convert and Clear buttons.
     */
    private JPanel getButtonPanel() {
        btnConvert = new JButton("Convert");
        btnClear = new JButton("Clear");

        // Convert button action
        btnConvert.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
```

```
                convertDistance();
            }
        });

        // Clear button action
        btnClear.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                txtDistance.setText("");
                rbInches.setSelected(false);
                rbFeet.setSelected(false);
                rbMiles.setSelected(false);
            }
        });

        JPanel panel = new JPanel();
        panel.add(btnConvert);
        panel.add(btnClear);

        return panel;
    }

    /**
     * Converts distance from kilometers to selected imperial unit.
     */
    private void convertDistance() {
        try {
            double km = Double.parseDouble(txtDistance.getText());
            double result = 0;
            String unit = "";

            if (rbInches.isSelected()) {
                result = km * 39370.1;
                unit = "inches";
            } else if (rbFeet.isSelected()) {
                result = km * 3280.84;
                unit = "feet";
            } else if (rbMiles.isSelected()) {
                result = km * 0.621371;
                unit = "miles";
            } else {
                JOptionPane.showMessageDialog(this, "Please select a unit to
convert.");
                return;
            }

            String message = String.format("%.2f km = %.2f %s", km, result,
unit);
            JOptionPane.showMessageDialog(this, message);

        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(this, "Invalid input. Please enter
a valid number.");
        }
    }
}
```

End of Document