Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester:(Spring, Year:2024), B.Sc. in CSE (Day)

Lab Report NO: 08
Course Title: Object-Oriented Programming Lab
Course Code: CSE 202 Section: 223 D9

Lab Experiment Name: Implementing Simple

GUI using AWT and SWING Lab. Manual.

Student Details

Name		ID
1.	Promod Chandra Das	231002005

Lab Date :
Submission Date :
Course Teacher's Name : Noyan Ali

Lab Report Status

Marks: …… **Signature:.....**
Comments:..... **Date:.....**

TITLE OF THE LAB REPORT EXPERIMENT:-

Implementing Simple

GUI using AWT and SWING Lab. Manual

OBJECTIVES/AIM :

- To gather knowledge of graphical user interface.
- • To implement simple GUI using AWT and SWING on lab..

Problem analysis:

For solving real life problem, all should know about the graphical user interface. In this Lab experiment the
main focus will be on GUI. Using GUI and java language, all have to implement different types of desktop based
application. So that, they can feel a little touch on industry.

- Write a program in java to generate calculator using GUI.

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

public class Calculator extends JFrame implements ActionListener {
   JButton b10, b11, b12, b13, b14, b15;
   JButton[] b = new JButton[10];
   int i, r, n1, n2;
   JTextField res;
   char op;

   public Calculator() {
      super("Calculator");
      setLayout(new BorderLayout());
      JPanel p = new JPanel();
      p.setLayout(new GridLayout(4, 4));
      for (int i = 0; i <= 9; i++) {
         b[i] = new JButton(i + "");
         p.add(b[i]);
         b[i].addActionListener(this);
      }
      b10 = new JButton("+");
      p.add(b10);
      b10.addActionListener(this);
```

```
    b11 = new JButton("-");
    p.add(b11);
    b11.addActionListener(this);

    b12 = new JButton("*");
    p.add(b12);
    b12.addActionListener(this);

    b13 = new JButton("/");
    p.add(b13);
    b13.addActionListener(this);

    b14 = new JButton("=");
    p.add(b14);
    b14.addActionListener(this);

    b15 = new JButton("C");
    p.add(b15);
    b15.addActionListener(this);

    res = new JTextField(10);
    add(p, BorderLayout.CENTER);
    add(res, BorderLayout.NORTH);
    setVisible(true);
    setSize(200, 200);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public void actionPerformed(ActionEvent ae) {
    JButton pb = (JButton) ae.getSource();
    if (pb == b15) {
        r = n1 = n2 = 0;
        res.setText("");
    } else if (pb == b14) {
        n2 = Integer.parseInt(res.getText());
        eval();
        res.setText("" + r);
    } else {
        boolean opf = false;
        if (pb == b10) {
            op = '+';
            opf = true;
        }
        if (pb == b11) {
            op = '-';
            opf = true;
        }
        if (pb == b12) {
            op = '*';
            opf = true;
        }
        if (pb == b13) {
            op = '/';
            opf = true;
        }
```

```
        if (opf == false) {
            for (i = 0; i < 10; i++) {
                if (pb == b[i]) {
                    String t = res.getText();
                    t += i;
                    res.setText(t);
                }
            }
        } else {
            n1 = Integer.parseInt(res.getText());
            res.setText("");
        }
    }
}

void eval() {
    switch (op) {
        case '+':
            r = n1 + n2;
            break;
        case '-':
            r = n1 - n2;
            break;
        case '*':
            r = n1 * n2;
            break;
        case '/':
            r = n1 / n2;
            break;
    }
}

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

- Lab Exercise (Submit as a report)

• Write a program in java using GUI to design a CGPA calculator However, Input must be taken from users.

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

public class CGPACalculator extends JFrame implements ActionListener {
    private JTextField[] creditFields;
    private JTextField[] gradeFields;
    private JTextField resultField;
    private JButton calculateButton;

    public CGPACalculator() {
```

```
    super("CGPA Calculator");

    // Create panels
    JPanel inputPanel = new JPanel(new GridLayout(0, 3));
    JPanel buttonPanel = new JPanel();
    JPanel resultPanel = new JPanel();

    // Create labels for input
    inputPanel.add(new JLabel("Course"));
    inputPanel.add(new JLabel("Credits"));
    inputPanel.add(new JLabel("Grade"));

    // Initialize input fields for 6 courses
    creditFields = new JTextField[6];
    gradeFields = new JTextField[6];

    for (int i = 0; i < 6; i++) {
        inputPanel.add(new JLabel("Course " + (i + 1) + ":"));
        creditFields[i] = new JTextField();
        gradeFields[i] = new JTextField();
        inputPanel.add(creditFields[i]);
        inputPanel.add(gradeFields[i]);
    }

    // Initialize the calculate button and result field
    calculateButton = new JButton("Calculate CGPA");
    calculateButton.addActionListener(this);

    resultField = new JTextField(10);
    resultField.setEditable(false);

    // Add components to panels
    buttonPanel.add(calculateButton);
    resultPanel.add(new JLabel("CGPA: "));
    resultPanel.add(resultField);

    // Add panels to frame
    add(inputPanel, BorderLayout.CENTER);
    add(buttonPanel, BorderLayout.SOUTH);
    add(resultPanel, BorderLayout.NORTH);

    // Setup frame
    setSize(400, 300);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setVisible(true);
}

public void actionPerformed(ActionEvent e) {
    if (e.getSource() == calculateButton) {
        try {
            double totalCredits = 0;
            double totalGradePoints = 0;

            for (int i = 0; i < 6; i++) {
                double credits = Double.parseDouble(creditFields[i].getText());
```

```
                double grade = Double.parseDouble(gradeFields[i].getText());

                totalCredits += credits;
                totalGradePoints += credits * grade;
            }

            double cgpa = totalGradePoints / totalCredits;
            resultField.setText(String.format("%.2f", cgpa));
        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(this, "Please enter valid numbers for credits and grades.", "Input
Error", JOptionPane.ERROR_MESSAGE);
        }
      }
   }

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

➢ Discussion & Conclusion

A calculator has been constructed in this project by using Java object oriented, GUI concept. While designing the calculator's layout, some difficulties were encountered. There was a flaw with the logic of the root function when it was being built. Connecting the equal button to the mathematical operations button took some extra time. Building the logic for the backspace button was the most critical part of running this program. The moment when the application was run properly and all the operations worked correctly, that moment was the most delightful moment during the whole time of making this project. I learned about plenty of java GUI methods, logic, and operations via this application. The aim of this program was to develop a calculator utilizing a Java GUI, which was completed successfully