AJ Practical 1

Date: **23-02-2023**

Roll no.: **20BCE119**

Name: **Kartavya Patel**

Course Code and Name: 2CSDE60 **Advanced Java**

Task

- Create a basic swing application of calculator which incorporates frame and event handling.

Code

```
package lab.prac1;

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

class prac1 extends JFrame {
    JButton btnAdd, btnSubtract, btnDivide, btnMultiply, btnClear, btnDelete, btnEquals, btnDot;
    JButton numBtn[];
    JTextField output;
    String previous, current, operator;

    public void processOutputNumber() {
        if (current.length() > 0) {
            String integerPart = current.split("\\.")[0];
            String decimalPart = current.split("\\.")[1];
            if (decimalPart.equals("0")) {
                current = integerPart;
            }
        }
    }

    public void delete() {
        if (current.length() > 0) {
            current = current.substring(0, current.length() - 1);
```

```java
        }
    }

    public void clear() {
        current = "";
        previous = "";
        operator = null;
    }

    public void updateOutput() {
        output.setText(current);
    }

    public void appendToOutput(String num) {
        if (num.equals(".") && current.contains(".")) {
            return;
        }
        current += num;
    }

    public void selectOperator(String newOperator) {

        if (current.isEmpty()) {
            operator = newOperator;
            return;
        }

        if (!previous.isEmpty()) {
            calculate();
        }

        operator = newOperator;
        previous = current;
        current = "";
    }

    public void calculate() {
        if (previous.length() < 1 || current.length() < 1) {
            return;
        }
        double result = 0.0;
        double num1 = Double.parseDouble(previous);
        double num2 = Double.parseDouble(current);
    switch (operator) {
        case "*":
            result = num1 * num2;
            break;
        case "+":
            result = num1 + num2;
            break;
        case "-":
            result = num1 - num2;
            break;
        case "÷":
            result = num1 / num2;
            break;
        default:
            break;
    }
    current = String.valueOf(result);
    operator = null;
    previous = "";
    processOutputNumber();
}

private class NumberBtnHandler implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        JButton selectedBtn = (JButton) e.getSource();
        for (JButton btn : numBtn) {
            if (selectedBtn == btn) {
                appendToOutput(btn.getText());
                updateOutput();
            }
        }
    }
}

private class OperatorBtnHandler implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        JButton selectedBtn = (JButton) e.getSource();
        if (selectedBtn == btnMultiply) {
            selectOperator(btnMultiply.getText());
        } else if (selectedBtn == btnAdd) {
```

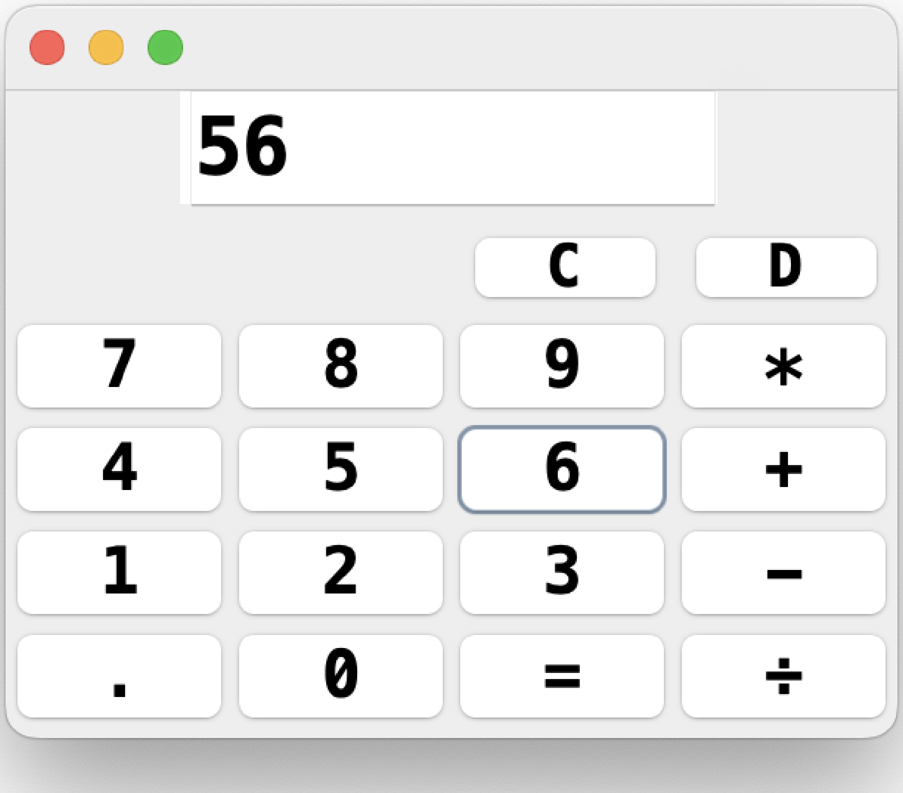
```
      selectOperator(btnAdd.getText());
    } else if (selectedBtn == btnSubtract) {
      selectOperator(btnSubtract.getText());
    } else if (selectedBtn == btnDivide) {
      selectOperator(btnDivide.getText());
    }
    updateOutput();
  }
}

private class OtherBtnHandler implements ActionListener {

  @Override
  public void actionPerformed(ActionEvent e) {
    JButton selectedBtn = (JButton) e.getSource();
    if (selectedBtn == btnDelete) {
      delete();
    } else if (selectedBtn == btnClear) {
      clear();
    } else if (selectedBtn == btnEquals) {
      calculate();
    }
    updateOutput();
  }
}

public prac1() {
  JPanel mainPanel = new JPanel();

  current = "";
  previous = "";

  JPanel row1 = new JPanel();
  JPanel row2 = new JPanel();
  JPanel row3 = new JPanel();
  JPanel row4 = new JPanel();
  JPanel row5 = new JPanel();

  output = new JTextField(16);
  btnSubtract = new JButton("-");
  btnAdd = new JButton("+");
  btnDivide = new JButton("÷");
  btnMultiply = new JButton("*");
```

```java
btnDot = new JButton(".");
btnEquals = new JButton("=");
btnClear = new JButton("C");
btnDelete = new JButton("D");

NumberBtnHandler numBtnHandler = new NumberBtnHandler();
OtherBtnHandler otherBtnHandler = new OtherBtnHandler();
OperatorBtnHandler opBtnHandler = new OperatorBtnHandler();

numBtn = new JButton[11];
numBtn[10] = btnDot;
for (int count = 0; count < numBtn.length - 1; count++) {
    numBtn[count] = new JButton(String.valueOf(count));
    numBtn[count].setFont(new Font("Monospaced", Font.BOLD, 22));
    numBtn[count].addActionListener(numBtnHandler);
}

btnDot.setFont(new Font("Monospaced", Font.BOLD, 22));
btnEquals.setFont(new Font("Monospaced", Font.BOLD, 22));
btnAdd.setFont(new Font("Monospaced", Font.BOLD, 22));
btnSubtract.setFont(new Font("Monospaced", Font.BOLD, 22));
btnDivide.setFont(new Font("Monospaced", Font.BOLD, 22));
btnMultiply.setFont(new Font("Monospaced", Font.BOLD, 22));
btnClear.setFont(new Font("Monospaced", Font.BOLD, 20));
btnDelete.setFont(new Font("Monospaced", Font.BOLD, 20));

output.setMaximumSize(new Dimension(185, 40));
output.setFont(new Font("Monospaced", Font.BOLD, 27));
output.setDisabledTextColor(new Color(0, 0, 0));
output.setMargin(new Insets(0, 5, 0, 0));
output.setText("0");

btnDot.addActionListener(numBtnHandler);
btnDelete.addActionListener(otherBtnHandler);
btnClear.addActionListener(otherBtnHandler);
btnEquals.addActionListener(otherBtnHandler);

btnMultiply.addActionListener(opBtnHandler);
btnAdd.addActionListener(opBtnHandler);
btnSubtract.addActionListener(opBtnHandler);
btnDivide.addActionListener(opBtnHandler);

row1.setLayout(new BoxLayout(row1, BoxLayout.LINE_AXIS));
```

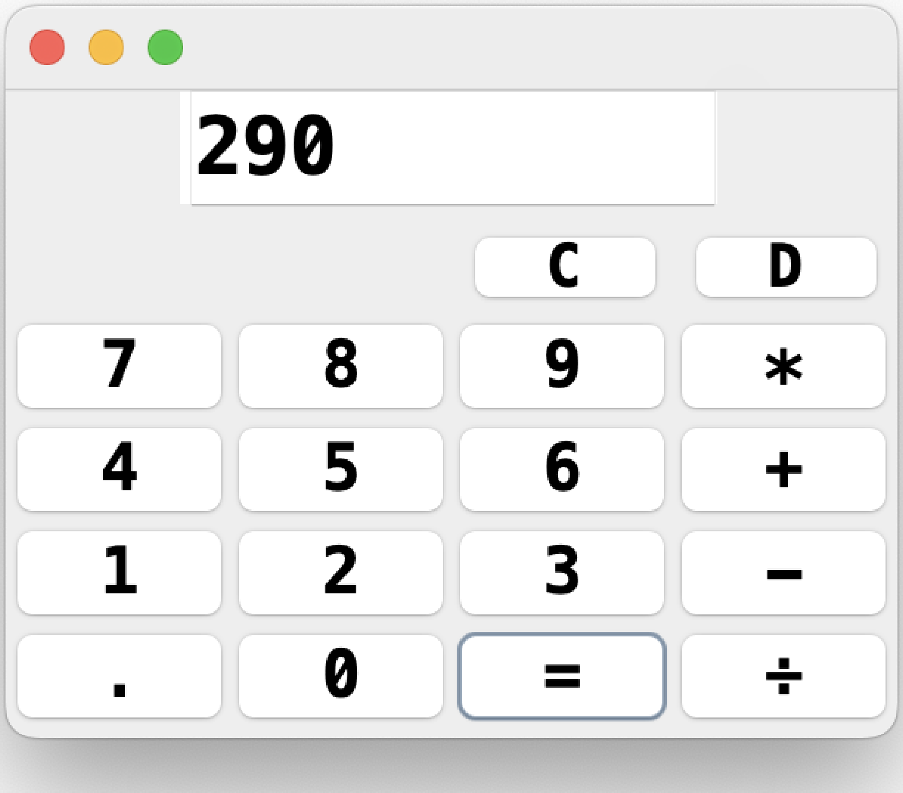
```java
        row2.setLayout(new BoxLayout(row2, BoxLayout.LINE_AXIS));
        row3.setLayout(new BoxLayout(row3, BoxLayout.LINE_AXIS));
        row4.setLayout(new BoxLayout(row4, BoxLayout.LINE_AXIS));
        row5.setLayout(new BoxLayout(row5, BoxLayout.LINE_AXIS));

        row1.add(Box.createHorizontalGlue());
        row1.add(btnClear);
        row1.add(btnDelete);

        row2.add(numBtn[7]);
        row2.add(numBtn[8]);
        row2.add(numBtn[9]);
        row2.add(btnMultiply);

        row3.add(numBtn[4]);
        row3.add(numBtn[5]);
        row3.add(numBtn[6]);
        row3.add(btnAdd);

        row4.add(numBtn[1]);
        row4.add(numBtn[2]);
        row4.add(numBtn[3]);
        row4.add(btnSubtract);

        row5.add(btnDot);
        row5.add(numBtn[0]);
        row5.add(btnEquals);
        row5.add(btnDivide);

        mainPanel.setLayout(new BoxLayout(mainPanel, BoxLayout.PAGE_AXIS));
        mainPanel.add(output);
        mainPanel.add(Box.createRigidArea(new Dimension(0, 5)));
        mainPanel.add(row1);
        mainPanel.add(row2);
        mainPanel.add(row3);
        mainPanel.add(row4);
        mainPanel.add(row5);

        add(mainPanel);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
        setSize(200, 280);
        setLocationRelativeTo(null);
```

```
    }

    public static void main(String[] args) {
        new prac1();
    }
}
```

Output



56



290