import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class SimpleCalculator extends JFrame implements ActionListener {

// Declare components

private JTextField displayField;

private JButton[] numberButtons = new JButton[10];

private JButton addButton, subButton, mulButton, divButton, equalsButton, clearButton;

private JPanel panel;

// Variables for calculations

private double num1 = 0, num2 = 0;

private char operator;

// Constructor

public SimpleCalculator() {

setTitle("Simple Calculator");

setSize(400, 500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new BorderLayout());

// Create display field

displayField = new JTextField();

displayField.setFont(new Font("Arial", Font.PLAIN, 24));

displayField.setEditable(false);

add(displayField, BorderLayout.NORTH);

// Create panel for buttons

panel = new JPanel();

panel.setLayout(new GridLayout(4, 4, 10, 10));

// Add number buttons to the panel

for (int i = 0; i < 10; i++) {

numberButtons[i] = new JButton(String.valueOf(i));

numberButtons[i].setFont(new Font("Arial", Font.PLAIN, 20));

numberButtons[i].addActionListener(this);

panel.add(numberButtons[i]);

}

// Create operation buttons

addButton = new JButton("+");

subButton = new JButton("-");

mulButton = new JButton("\*");

divButton = new JButton("/");

equalsButton = new JButton("=");

clearButton = new JButton("C");

// Set button fonts and action listeners

JButton[] operationButtons = {addButton, subButton, mulButton, divButton, equalsButton, clearButton};

for (JButton button : operationButtons) {

button.setFont(new Font("Arial", Font.PLAIN, 20));

button.addActionListener(this);

panel.add(button);

}

// Add panel to the frame

add(panel, BorderLayout.CENTER);

}

@Override

public void actionPerformed(ActionEvent e) {

for (int i = 0; i < 10; i++) {

if (e.getSource() == numberButtons[i]) {

displayField.setText(displayField.getText() + i);

}

}

if (e.getSource() == addButton) {

num1 = Double.parseDouble(displayField.getText());

operator = '+';

displayField.setText("");

} else if (e.getSource() == subButton) {

num1 = Double.parseDouble(displayField.getText());

operator = '-';

displayField.setText("");

} else if (e.getSource() == mulButton) {

num1 = Double.parseDouble(displayField.getText());

operator = '\*';

displayField.setText("");

} else if (e.getSource() == divButton) {

num1 = Double.parseDouble(displayField.getText());

operator = '/';

displayField.setText("");

} else if (e.getSource() == equalsButton) {

num2 = Double.parseDouble(displayField.getText());

switch (operator) {

case '+':

displayField.setText(String.valueOf(num1 + num2));

break;

case '-':

displayField.setText(String.valueOf(num1 - num2));

break;

case '\*':

displayField.setText(String.valueOf(num1 \* num2));

break;

case '/':

if (num2 != 0) {

displayField.setText(String.valueOf(num1 / num2));

} else {

displayField.setText("Error");

}

break;

}

} else if (e.getSource() == clearButton) {

displayField.setText("");

}

}

// Main method

public static void main(String[] args) {

SimpleCalculator calculator = new SimpleCalculator();

calculator.setVisible(true);

}

}