Міністерство освіти і науки України

Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій

Кафедра прикладних інформаційних систем

Звіт

з лабораторної роботи №7

тема: «Принципи побудови графічного інтерфейсу в Java»

Виконав:

Група ПП-11/2

%username%

Перевірив:

асистент

Гарко І.І..

Київ 202

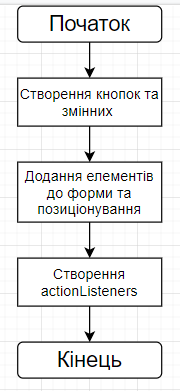
**Мета роботи**

Отримання практичних навичок побудови форм та обробки подій в них у мові Java.

**Завдання**

Створити калькулятор

Блоксхема:



Код:

package com.company;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Main extends JFrame {

private JButton numberOne;

private JButton numberTwo;

private JButton numberThree;

private JButton numberFour;

private JButton numberFive;

private JButton numberSix;

private JButton numberSeven;

private JButton numberEight;

private JButton numberNine;

private JButton numberZero;

private JButton equal;

private JButton add;

private JButton substract;

private JButton multiply;

private JButton divideAndConquer;

private JButton solve;

private JButton clr;

private JButton decimal;

private double TEMP;

private double SolveTEMP;

private JTextField result;

Boolean addBool = false;

Boolean subBool = false;

Boolean divBool = false;

Boolean mulBool = false;

String display = "";

public Main() {

JPanel buttons = new JPanel();

buttons.setLayout(new GridLayout(4, 3));

buttons.add(numberOne = new JButton("1"));

buttons.add(numberTwo = new JButton("2"));

buttons.add(numberThree = new JButton("3"));

buttons.add(numberFour = new JButton("4"));

buttons.add(numberFive = new JButton("5"));

buttons.add(numberSix = new JButton("6"));

buttons.add(numberSeven = new JButton("7"));

buttons.add(numberEight = new JButton("8"));

buttons.add(numberNine = new JButton("9"));

buttons.add(numberZero = new JButton("0"));

buttons.add(clr = new JButton("C"));

buttons.add(decimal = new JButton("."));

JPanel textBar = new JPanel();

textBar.setLayout(new FlowLayout());

textBar.add(result = new JTextField(20));

result.setHorizontalAlignment(JTextField.RIGHT);

result.setEditable(true);

JPanel operations = new JPanel();

operations.setLayout(new GridLayout(5, 1));

operations.add(add = new JButton("+"));

operations.add(substract = new JButton("-"));

operations.add(multiply = new JButton("\*"));

operations.add(divideAndConquer = new JButton("/"));

operations.add(solve = new JButton("="));

JPanel main = new JPanel();

main.setLayout(new GridLayout());

main.add(textBar, BorderLayout.NORTH);

main.add(buttons, BorderLayout.SOUTH);

main.add(operations, BorderLayout.EAST);

add(main);

numberOne.addActionListener(new ListenToOne());

numberTwo.addActionListener(new ListenToTwo());

numberThree.addActionListener(new ListenToThree());

numberFour.addActionListener(new ListenToFour());

numberFive.addActionListener(new ListenToFive());

numberSix.addActionListener(new ListenToSix());

numberSeven.addActionListener(new ListenToSeven());

numberEight.addActionListener(new ListenToEight());

numberNine.addActionListener(new ListenToNine());

numberZero.addActionListener(new ListenToZero());

add.addActionListener(new ListenToAdd());

substract.addActionListener(new ListenToSubtract());

multiply.addActionListener(new ListenToMultiply());

divideAndConquer.addActionListener(new ListenToDivide());

solve.addActionListener(new ListenToSolve());

clr.addActionListener(new ListenToClear());

decimal.addActionListener(new ListenToDecimal());

}

class ListenToClear implements ActionListener {

public void actionPerformed(ActionEvent e) {

result.setText("");

addBool = false;

subBool = false;

mulBool = false;

divBool = false;

TEMP = 0;

SolveTEMP = 0;

}

}

class ListenToOne implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "1");

}

}

class ListenToTwo implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "2");

}

}

class ListenToThree implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "3");

}

}

class ListenToFour implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "4");

}

}

class ListenToFive implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "5");

}

}

class ListenToSix implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "6");

}

}

class ListenToSeven implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "7");

}

}

class ListenToEight implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "8");

}

}

class ListenToNine implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "9");

}

}

class ListenToZero implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + "0");

}

}

class ListenToAdd implements ActionListener {

public void actionPerformed(ActionEvent e) {

TEMP = Double.parseDouble(result.getText());

result.setText("");

addBool = true;

}

}

class ListenToSubtract implements ActionListener {

public void actionPerformed(ActionEvent e) {

TEMP = Double.parseDouble(result.getText());

result.setText("");

subBool = true;

}

}

class ListenToMultiply implements ActionListener {

public void actionPerformed(ActionEvent e) {

TEMP = Double.parseDouble(result.getText());

result.setText("");

mulBool = true;

}

}

class ListenToDivide implements ActionListener {

public void actionPerformed(ActionEvent e) {

TEMP = Double.parseDouble(result.getText());

result.setText("");

divBool = true;

}

}

class ListenToDecimal implements ActionListener {

public void actionPerformed(ActionEvent e) {

display = result.getText();

result.setText(display + ".");

}

}

class ListenToSolve implements ActionListener {

public void actionPerformed(ActionEvent e) {

SolveTEMP = Double.parseDouble(result.getText());

if (addBool == true) {

SolveTEMP = SolveTEMP + TEMP;

result.setText(Double.toString(SolveTEMP));

} else if ( subBool == true) {

SolveTEMP = TEMP - SolveTEMP;

result.setText(Double.toString(SolveTEMP));

} else if ( mulBool == true) {

SolveTEMP = SolveTEMP \* TEMP;

result.setText(Double.toString(SolveTEMP));

} else if ( divBool == true) {

SolveTEMP = TEMP / SolveTEMP;

result.setText(Double.toString(SolveTEMP));

}

addBool = false;

subBool = false;

mulBool = false;

divBool = false;

}

}

public static void main(String[] args) {

Main calc = new Main();

calc.pack();

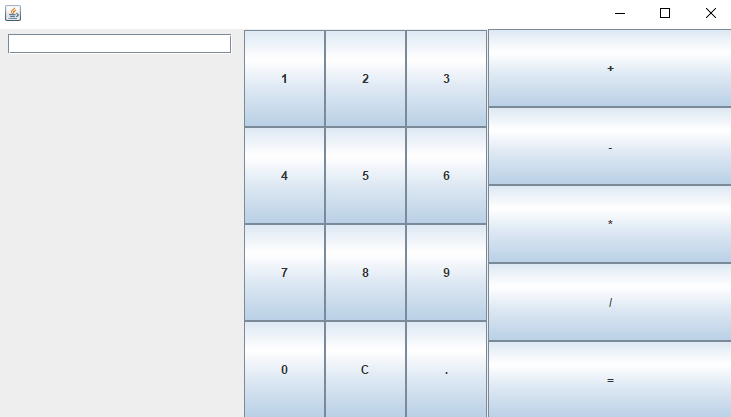
calc.setLocationRelativeTo(null);

calc.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

calc.setVisible(true);

}

}



**Контрольні запитання**

1. Що таке подія?

Об’єкт, що описує зміни стану джерела, з яким він пов’язаний

1. Які класи бібліотеки AWT відповідають за опис подій?

ActionEvent

1. Що таке Listener?

Метод,що викликається при виконанні користувачем дій, про які розробник хоче дізнатись

1. Як додати обробку події у елементи форми?

.addActionListener(new ActionListener())

1. Як видалити обробку події?

component.removeActionListener(theActionListenerYouWantToRemove)

1. Які події ви обробляли у лабораторній роботі?

Натискання кнопки

1. Як створити форму?

Створити нащадок класу JFrame

1. Як зв’язати двi форми?

Використати JDialog

1. Як видалити обробку події?

Backspace

1. Які події ви обробляли у лабораторній роботі?

Натискання кнопки