

Министерство науки и высшего образования РФ

Пензенский государственный университет

Кафедра «Вычислительная техника»

Отчет

по лабораторной работе №1

по дисциплине «Программирование на языке Java»

на тему: «Графические интерфейсы»

Вариант 6

Выполнили студенты группы 23ВВВ3:

Балаев Г. С.

Саранцев. Е. А.

Приняли:

Юрова О. В.

Карамышева Н. С.

Пенза, 2025

Цель работы

Научиться разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing.

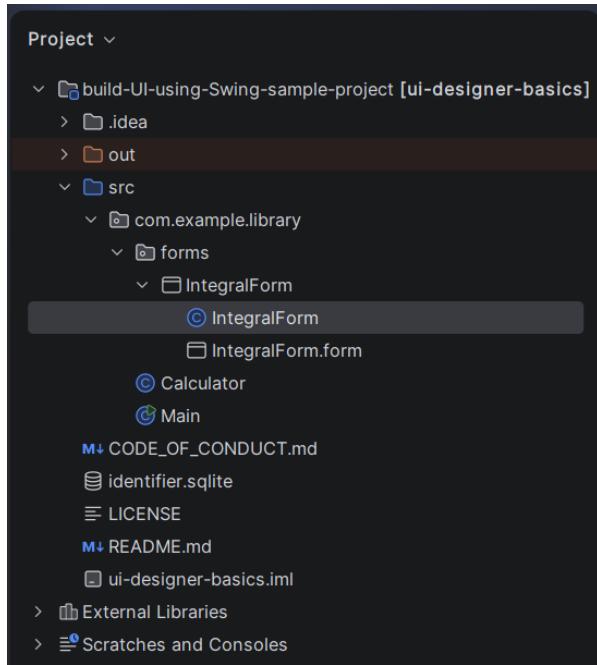
Лабораторное задание

Вычислить определенный интеграл функции в соответствии с вариантом задания (Приложение 1). Разработать приложение, обладающее графическим интерфейсом с использованием языка Java и библиотеки Swing. Приложение должно содержать 3 поля ввода (JTextField), доступных для редактирования, и соответственно таблицу (JTable) с четырьмя колонками: нижняя граница интегрирования, верхняя граница интегрирования, шаг интегрирования и результат вычисления. Кроме того, должны присутствовать 3 кнопки (JButton): добавить, удалить, вычислить. Для добавления/удаления строки и вычисления значения определенного интеграла для функции в соответствии с вариантом задания (Приложение 1) и параметров выделенной строки таблицы. Результат должен выводиться в четвертой колонке, которая не доступна для редактирования. Оформление лабораторной работы должно быть выполнено в соответствии с требованиями, приведенными в Приложении 2.

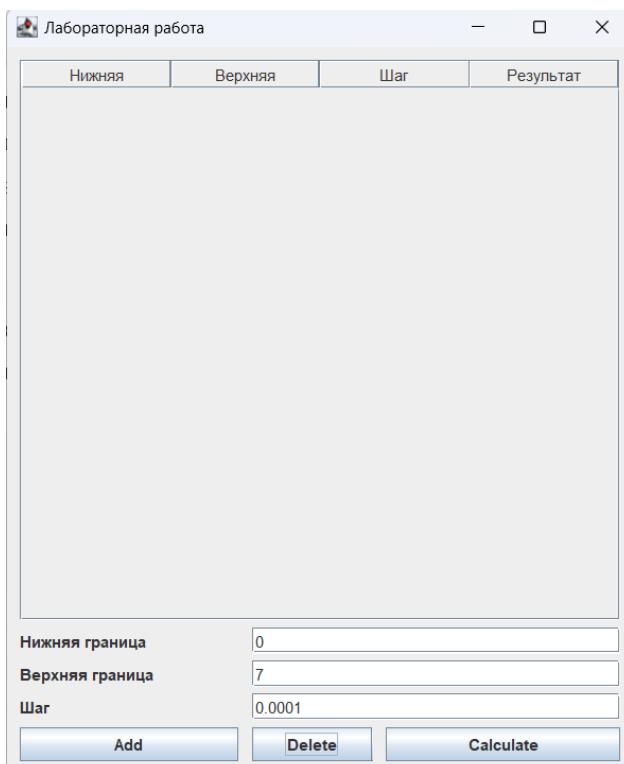
Вариант 2: $\sin(x)$

Ход работы:

- Создали проект в IDE JetBrains для разработки графического интерфейса.



- Создали графический интерфейс приложения с использованием языка Java и библиотеки Swing в режиме Design. Добавили 3 поля ввода данных, таблицу с данными и 3 кнопки для добавления и удаления строки данных, а также для вычисления результата.



3. Сравнили результаты работы программы (с разными шагами) с вычислением интеграла в профессиональных математических программах.

Лабораторная работа

Нижняя	Верхняя	Шаг	Результат
0.0	7.0	1.0	0,0365
0.0	7.0	0.5	0,0229
0.0	7.0	0.25	0,1064
0.0	7.0	0.125	0,1698
0.0	7.0	0.0625	0,2065
0.0	7.0	0.03125	0,2259
0.0	7.0	1.0E-4	0,2460

Нижняя граница
Верхняя граница
Шаг

Add Delete Calculate

1 X

$$\int_0^7 \sin(x) dx$$

= 0.246097745657

Результат работы программы:

The screenshot shows a Java Swing application window titled "Лабораторная работа". The window contains a table with four columns: "Нижняя" (Lower), "Верхняя" (Upper), "Шаг" (Step), and "Результат" (Result). Below the table are three input fields: "Нижняя граница" (Lower boundary) with value 0, "Верхняя граница" (Upper boundary) with value 7, and "Шаг" (Step) with value 0.0001. At the bottom are three buttons: "Add", "Delete", and "Calculate".

Нижняя	Верхняя	Шаг	Результат
0.0	7.0	1.0	0,0365
0.0	7.0	0.5	0,0229
0.0	7.0	0.25	0,1064
0.0	7.0	0.125	0,1698
0.0	7.0	0.0625	0,2065
0.0	7.0	0.03125	0,2259
0.0	7.0	1.0E-4	0,2460

Нижняя граница: 0
Верхняя граница: 7
Шаг: 0.0001

Add Delete Calculate

Вывод:

В ходе данной лабораторной работы мы научились создавать графический интерфейс с использованием языка Java и библиотеки Swing в IDE JetBrains.

Листинг:

Main.java:

```
package com.example.library;

import javax.swing.*;
import com.example.library.forms.IntegralForm;

public class Main {
    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            @Override
            public void run() {
                JFrame frame = new JFrame("Лабораторная работа");

                IntegralForm form = new IntegralForm();
                frame.setContentPane(form.getRootPanel());

                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.pack();
                frame.setLocationRelativeTo(null);
                frame.setVisible(true);
            }
        });
    }
}
```

Calculator.java:

```
package com.example.library;

import static java.lang.Math.sin;

public class Calculator {

    private static double f(double x) {
        return sin(x);
    }

    public static double calculate(double left, double right, double step) {
        int n = (int)((right - left) / step);
        double result = 0;

        for (int i = 0; i < n - 1; i++) {
            double x0 = left + i * step;
            double x1 = left + (i + 1) * step;
            result += (f(x0) + f(x1)) * step / 2;
        }

        double last = left + n * step;
        if (last < right) {
            double lastTrap = (f(last) + f(right)) * (right - last) / 2;
            result += lastTrap;
        }
        return result;
    }
}
```

IntegralForm.java:

```
package com.example.library.forms;

import com.example.library.Calculator;

import javax.swing.*;
import javax.swing.table.DefaultTableModel;

public class IntegralForm {
    private JPanel rootPanel;
    private JTextField fieldLowerBound;
    private JTextField fieldUpperBound;
    private JTextField fieldStep;
    private JButton btnAdd;
    private JButton btnDelete;
    private JButton btnCalculate;
    private JTable table1;

    private DefaultTableModel model;

    public IntegralForm() {
        String[] columns = {"Нижняя", "Верхняя", "Шаг", "Результат"};
        model = new DefaultTableModel(columns, 0) {
            public boolean isCellEditable(int row, int column) {
                if (column == 3) {
                    return false;
                }
                return true;
            }
        };
        table1.setModel(model);

        btnAdd.addActionListener(e -> {
            double low =
Double.parseDouble(fieldLowerBound.getText());
            double high =
Double.parseDouble(fieldUpperBound.getText());
            double step = Double.parseDouble(fieldStep.getText());

            model.addRow(new Object[]{low, high, step, ""});
        });

        btnCalculate.addActionListener(e -> {
            int row = table1.getSelectedRow();
            if (row != -1) {
                double left = Double.parseDouble(model.getValueAt(row,
0).toString());
                double right =
Double.parseDouble(model.getValueAt(row, 1).toString());
                double step = Double.parseDouble(model.getValueAt(row,
2).toString());
                double res = Calculator.calculate(left, right, step);
            }
        });
    }
}
```

```
        model.setValueAt(String.format("% .4f", res), row, 3);
    }
}) ;

btnDelete.addActionListener(e -> {
    int row = table1.getSelectedRow();
    if (row != -1) model.removeRow(row);
}) ;
}

public JPanel getRootPanel() {
    return rootPanel;
}
}
```