Министерство науки и высшего образования РФ

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №4

по дисциплине «разработка кроссплатформенных приложений»

на тему «Работа с файлами»

Вариант №2

Выполнили: студенты группы 22ВВВ1

Дунюшкин В.А.

НиколаевА.А.

Приняла:

Доцент Юрова О.В.

Пенза 2025

**Тема работы:**

Изучить механизм работы с файлами

**Ход работы:**

Модифицировать приложение из предыдущей лабораторной работы, реализовав сохранение в файл и загрузку данных из файла. Предусмотреть сохранение данных, как в текстовом виде, так и в двоичном (с использованием механизма сериализации).

**Описаниe:**

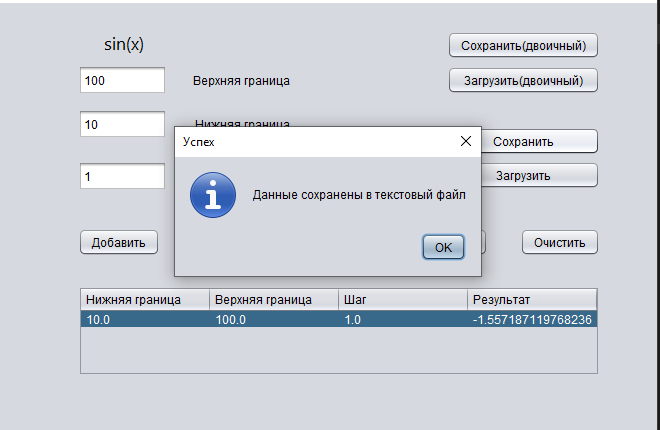
1. Создаём вложенный классы saveToBinaryFile, loadFromBinaryFile, saveToTextFile, loadFromTextFile которые будут сохранять и загружать данные из файлов

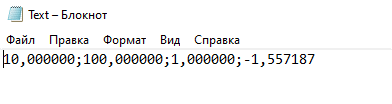
private void saveToTextFile(File file) {  
 try (FileWriter writer = new FileWriter(file)) {  
 DefaultTableModel model = (DefaultTableModel) Main\_Table.getModel();  
 for (int i = 0; i < model.getRowCount(); i++) {  
 Object resultObj = model.getValueAt(i, 3);  
 double result = (resultObj instanceof Number) ? ((Number)resultObj).doubleValue() :  
 (resultObj != null && !resultObj.toString().isEmpty()) ?  
 Double.*parseDouble*(resultObj.toString()) : 0.0;  
  
 writer.write(String.*format*("%f;%f;%f;%f%n",  
 Double.*parseDouble*(model.getValueAt(i, 0).toString()),  
 Double.*parseDouble*(model.getValueAt(i, 1).toString()),  
 Double.*parseDouble*(model.getValueAt(i, 2).toString()),  
 result));  
 }  
 JOptionPane.*showMessageDialog*(this, "Данные сохранены в текстовый файл",  
 "Успех", JOptionPane.*INFORMATION\_MESSAGE*);  
 } catch (IOException | NumberFormatException e) {  
 JOptionPane.*showMessageDialog*(this, "Ошибка сохранения: " + e.getMessage(),  
 "Ошибка", JOptionPane.*ERROR\_MESSAGE*);  
 }  
}  
  
private void loadFromTextFile(File file) {  
 try (BufferedReader reader = new BufferedReader(new FileReader(file))) {  
 DefaultTableModel model = (DefaultTableModel) Main\_Table.getModel();  
 model.setRowCount(0);  
 listR.clear();  
  
 String line;  
 while ((line = reader.readLine()) != null) {  
 String[] parts = line.split(";");  
 if (parts.length == 4) {  
 double low = Double.*parseDouble*(parts[0].replace(',', '.'));  
 double high = Double.*parseDouble*(parts[1].replace(',', '.'));  
 double step = Double.*parseDouble*(parts[2].replace(',', '.'));  
 double result = Double.*parseDouble*(parts[3].replace(',', '.'));  
  
 model.addRow(new Object[]{low, high, step, result});  
 listR.add(new Rect\_integral(high, low, step, result));  
 }  
 }  
 } catch (IOException | NumberFormatException e) {  
 JOptionPane.*showMessageDialog*(this, "Ошибка загрузки: " + e.getMessage(),  
 "Ошибка", JOptionPane.*ERROR\_MESSAGE*);  
 }  
}

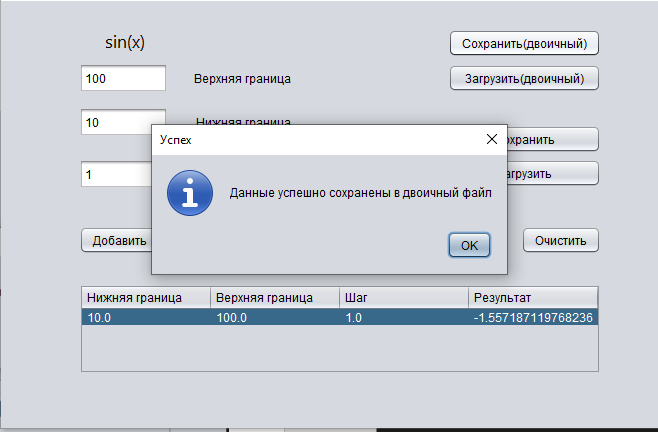
private void saveToBinaryFile(File file) {  
 try (ObjectOutputStream out = new ObjectOutputStream(  
 new BufferedOutputStream(new FileOutputStream(file)))) {  
 out.writeObject(listR);  
 JOptionPane.*showMessageDialog*(this, "Данные успешно сохранены в двоичный файл",   
 "Успех", JOptionPane.*INFORMATION\_MESSAGE*);  
 } catch (IOException e) {  
 JOptionPane.*showMessageDialog*(this, "Ошибка при сохранении файла: " + e.getMessage(),   
 "Ошибка", JOptionPane.*ERROR\_MESSAGE*);  
 }  
}  
  
// Метод для загрузки из двоичного файла  
private void loadFromBinaryFile(File file) {  
 try (ObjectInputStream in = new ObjectInputStream(  
 new BufferedInputStream(new FileInputStream(file)))) {  
 @SuppressWarnings("unchecked")  
 LinkedList<Rect\_integral> loadedList = (LinkedList<Rect\_integral>) in.readObject();  
  
 DefaultTableModel model = (DefaultTableModel) Main\_Table.getModel();  
 model.setRowCount(0);  
 listR.clear();  
  
 for (Rect\_integral rec : loadedList) {  
 model.addRow(new Object[]{  
 rec.getLow\_value(),   
 rec.getHigh\_value(),   
 rec.getStep(),   
 rec.getResult()  
 });  
 listR.add(rec);  
 }  
 JOptionPane.*showMessageDialog*(this, "Данные успешно загружены из двоичного файла",   
 "Успех", JOptionPane.*INFORMATION\_MESSAGE*);  
 } catch (IOException | ClassNotFoundException e) {  
 JOptionPane.*showMessageDialog*(this, "Ошибка при загрузке файла: " + e.getMessage(),   
 "Ошибка", JOptionPane.*ERROR\_MESSAGE*);  
 }  
}

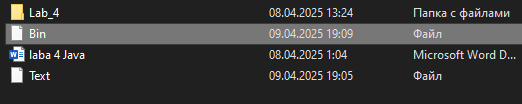
1. В программе вводим данные
2. После можем сохранить данные и загрузить их в следующий запуск программы

**Результат работы программы:**









**Вывод:**

В ходе лабораторной работы научились работать с файлами