Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №2

по курсу «Программирование на языке JAVA»

на тему «Работа с коллекциями объектов»

Вариант №1

Выполнили:

студенты группы 20ВВП1

Тумасов В.В

Ермолаев А.Д

Портнов Н.С

Приняли:

к.т.н Юрова О.В

к.т.н Карамышева Н.С

Пенза 2023

**Цель работы:** изучить библиотеку стандартных коллекций Java Collections Framework, позволяющую хранить различные структуры данных.

Задание на лабораторную работу

Модифицировать приложение из предыдущей лабораторной работы, реализовав хранение данных таблицы с использованием библиотеки коллекций. Для этого реализовать класс RecIntegral, способный хранить одну запись таблицы. Для нечетных вариантов в качестве класса-коллекции выбрать ArrayList, для четных - LinkedList. Кроме того, добавить пару кнопок: очистить / заполнить, которые будут очищать таблицу и заполнять ее данными из коллекции соответственно. Оформление лабораторной работы должно быть выполнено в соответствии с требованиями, приведенными в Приложении 2.

Этапы выполнения:

1)Для выполнения лабораторного задания был создан класс RecIntegral,в котором,также,находится функция вычисления интеграла.

package lab1;

public class RecIntegral extends NewJFrame {

Float inf[];

float a4;

public RecIntegral(float a1,float a2,float a3)

{

float sch,b;

sch=0;

a4=0;

b=a1;

float t1=(float)((a2-a1)%a3);

if(t1 > 0)

{

a4=t1\*((float)0.5\*((1/a1)+(1/(a1+t1))));

}

b=a1+t1;

while((b + a3) <= a2)

{

a4+=a3\*(0.5\*((1/b)+(1/(b+a3))));

b+=a3;

}

inf=new Float[4];

inf[0]=a1;

inf[1]=a2;

inf[2]=a3;

inf[3]=a4;

}

public Float[] n1()

{

return inf;

}

public Float result()

{

return a4;

}

}

2)Методы n1() и result() выводят весь массив значений и только результат вычисления соответственно.

3)В классе NewJFrame был добавлен список обьектов класса RecIntegral.

ArrayList<RecIntegral> record = new ArrayList<RecIntegral>();

4)Был реализован метод для занесения в объект списка record значений,введенных с клавиатуры.

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

Float d;

num1=Float.parseFloat(jTextField6.getText());

num2=Float.parseFloat(jTextField5.getText());

num3=Float.parseFloat(jTextField8.getText());

record.add(new RecIntegral(num1,num2,num3));

String s=Float.toString(record.get(record.size()-1).result());

jTextField7.setText(s);

d=Float.parseFloat(jTextField7.getText());

Float data[] = {num1,num2,num3,d};

DefaultTableModel model = (DefaultTableModel)table1.getModel();

model.addRow(data);

}

5)Был реализован метод для очистки списка объектов.

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel model = (DefaultTableModel)table1.getModel();

model.setRowCount(0);

}

6)Был реализован метод для загрузки списка объектов в заранее созданную таблицу.

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel model = (DefaultTableModel)table1.getModel();

for(int i=0;i<record.size();i++)

{

model.addRow(record.get(i).n1());

}

}

Листинг

Класс NewJFrame

public class NewJFrame extends javax.swing.JFrame {

Scanner sc= new Scanner(System.in);

int func;

float num1,num2,num3;

ArrayList<RecIntegral> record = new ArrayList<RecIntegral>();

public NewJFrame() {

initComponents();

jTextField6.setEditable(false);

jTextField5.setEditable(false);

jTextField8.setEditable(false);

jTextField7.setEditable(false);

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel model = (DefaultTableModel)table1.getModel();

jTextField6.setText("");

jTextField5.setText("");

jTextField8.setText("");

jTextField7.setText("");

model.removeRow(table1.getSelectedRow());

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

jTextField6.setEditable(true);

jTextField5.setEditable(true);

jTextField8.setEditable(true);

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

Float d;

num1=Float.parseFloat(jTextField6.getText());

num2=Float.parseFloat(jTextField5.getText());

num3=Float.parseFloat(jTextField8.getText());

record.add(new RecIntegral(num1,num2,num3));

String s=Float.toString(record.get(record.size()-1).result());

jTextField7.setText(s);

d=Float.parseFloat(jTextField7.getText());

Float data[] = {num1,num2,num3,d};

DefaultTableModel model = (DefaultTableModel)table1.getModel();

model.addRow(data);

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel model = (DefaultTableModel)table1.getModel();

model.setRowCount(0);

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel model = (DefaultTableModel)table1.getModel();

for(int i=0;i<record.size();i++)

{

model.addRow(record.get(i).n1());

}

}

public static void main(String args[]) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new NewJFrame().setVisible(true);

}

});

}

Класс RecIntegral

package lab1;

public class RecIntegral extends NewJFrame {

Float inf[];

float a4;

public RecIntegral(float a1,float a2,float a3)

{

float sch,b;

sch=0;

a4=0;

b=a1;

float t1=(float)((a2-a1)%a3);

if(t1 > 0)

{

a4=t1\*((float)0.5\*((1/a1)+(1/(a1+t1))));

}

b=a1+t1;

while((b + a3) <= a2)

{

a4+=a3\*(0.5\*((1/b)+(1/(b+a3))));

b+=a3;

}

inf=new Float[4];

inf[0]=a1;

inf[1]=a2;

inf[2]=a3;

inf[3]=a4;

}

public Float[] n1()

{

return inf;

}

public Float result()

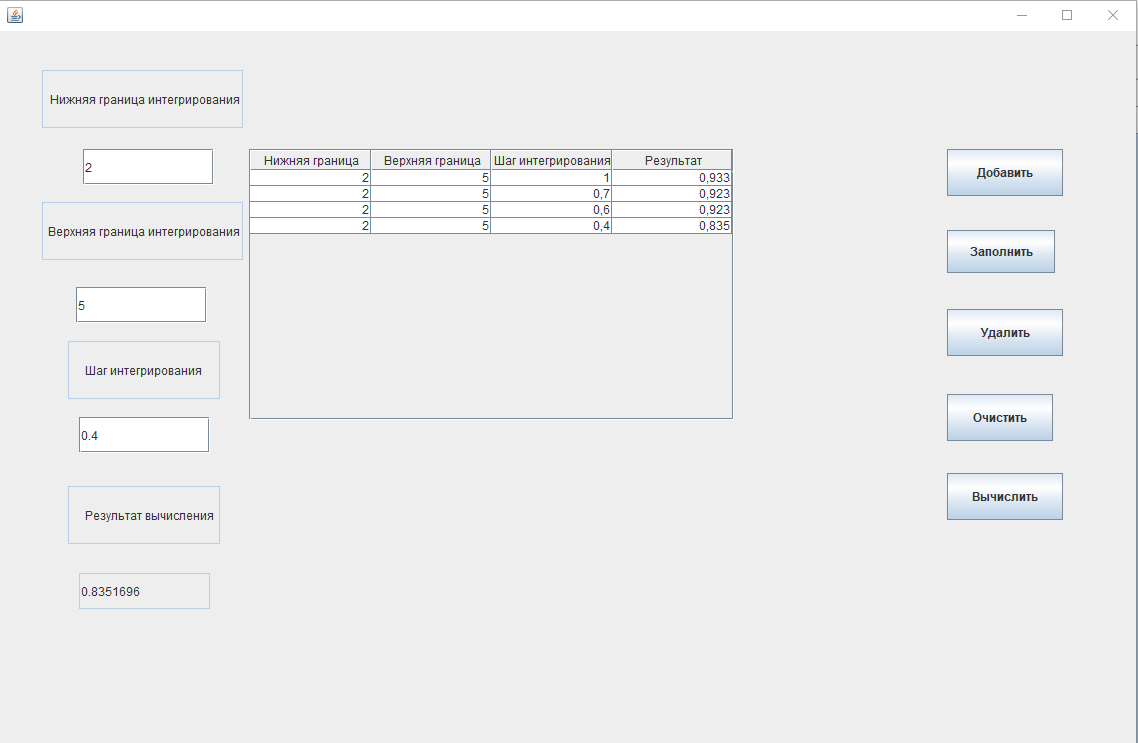
{

return a4;

}

}

Результат работы программы:



Вывод:в процессе выполнения данной лабораторной работы была изучена методика с коллекциями,в частности с ArrayList.