Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №1

по курсу «Программирование на Java»

Выполнили:

студенты группы 20ВВП1

Кирюткин И. А.

Кодиров И.Н.

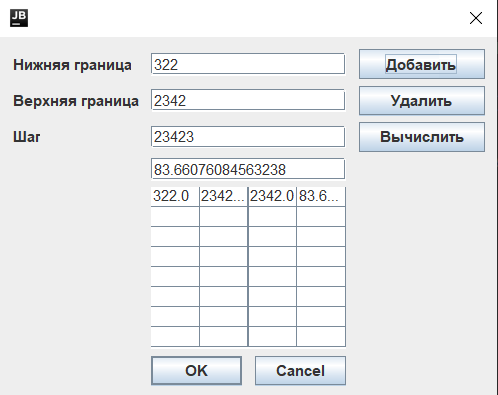
Приняли:

Юрова О.В.

**Цель работы:** научиться разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing.

**Задание:** Вычислить определенный интеграл функции. 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.

**Результат работы программы:**



Листинг:

import javax.swing.\*;  
import javax.swing.table.DefaultTableModel;  
import java.awt.event.\*;  
  
public class Form extends JDialog {  
 private JPanel contentPane;  
 private JButton buttonOK;  
 private JButton buttonCancel;  
 private JTable table1;  
 private JTextField textField1;  
 private JTextField textField2;  
 private JTextField textField3;  
 private JButton ButtonAdd;  
 private JButton ButtonDelete;  
 private JButton ButtonCalc;  
 private JTextField textField4;  
 private double[] dataT = new double[4];  
 private double[] dataCount = new double[4];  
 int num =1;  
 boolean flagSave=false;  
 String[] columnNames = {  
 "Верхняя граница интегрирования",  
 "Нижняя граница интегрирования",  
 "Шаг интегрирования",  
 "Результат"  
 };  
 Object[][] data = new Object[8][4];  
 public Form() {  
 flagSave=false;  
 //contentPane.add(table1);  
 setContentPane(contentPane);  
 setModal(true);  
 getRootPane().setDefaultButton(buttonOK);  
 table1.setModel(new DefaultTableModel(data, columnNames));  
 buttonOK.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 onOK();  
 }  
 });  
  
 buttonCancel.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
  
 }  
 });  
 ButtonAdd.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 AddTable();  
 }  
 });  
 ButtonDelete.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 delete();  
 }  
 });  
 ButtonCalc.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 Calc();  
 }  
 });  
 // call onCancel() when cross is clicked  
 setDefaultCloseOperation(DO\_NOTHING\_ON\_CLOSE);  
 addWindowListener(new WindowAdapter() {  
 public void windowClosing(WindowEvent e) {  
 onCancel();  
 }  
 });  
  
 // call onCancel() on ESCAPE  
 contentPane.registerKeyboardAction(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 onCancel();  
 }  
 }, KeyStroke.getKeyStroke(KeyEvent.VK\_ESCAPE, 0), JComponent.WHEN\_ANCESTOR\_OF\_FOCUSED\_COMPONENT);  
  
 }  
 public static double InFunction(double x) //Подынтегральная функция  
 {  
 return Math.sin(x\*x);  
 }  
 private void delete()  
 {  
 textField1.setText("");  
 textField2.setText("");  
 textField3.setText("");  
 textField4.setText("");  
 flagSave=false;  
 }  
 private void Calc() {  
 dataT[0] =Double.valueOf(textField1.getText());  
 dataT[1] =Double.valueOf(textField3.getText());  
 dataT[2] =Double.valueOf(textField2.getText());  
 dataT[3] =Double.valueOf("0");  
 dataT[3]= Trap(dataT[0],dataT[1],dataT[2]);  
 for(int i=0;i<4;i++)  
 {  
 dataCount[i]=dataT[i];  
 }  
 flagSave=true;  
 //createTable();  
 textField4.setText(""+dataT[3]);  
  
 }  
  
 private void AddTable() {  
 // dataT[0] =Double.valueOf(textField1.getText());  
 // dataT[1] =Double.valueOf(textField3.getText());  
 // dataT[2] =Double.valueOf(textField2.getText());  
 // dataT[3] =Double.valueOf("0");  
 //dataT[3]= Trap(dataT[0],dataT[1],dataT[2]);  
  
 if(flagSave) {  
 createTable();  
 num++;  
 }//textField4.setText(""+dataT[3]);  
 }  
  
 public void createTable(){  
 int i=100;  
  
 data[num-1][0]=dataCount[0];//1  
 data[num-1][1]=dataCount[1];//2  
 data[num-1][2]=dataCount[2];//3  
 data[num-1][3]=dataCount[3];//4  
 table1.setModel(new DefaultTableModel(data, columnNames));  
 }  
 public double Trap(double a,double b, double h){  
 double result=0;  
 int n = (int)((a-b)/h);  
 result += (InFunction(a)+InFunction(b))/2;  
 for(int i = 1; i < n; i++) {  
 result += InFunction(b + h \* i);  
 }  
 return h\*result;  
 }  
  
 private void onOK() {  
 // add your code here  
 dispose();  
 }  
  
 private void onCancel() {  
 // add your code here if necessary  
 dispose();  
 }  
  
 public static void main(String[] args) {  
 Form dialog = new Form();  
 dialog.pack();  
 dialog.setVisible(true);  
 System.exit(0);  
 }  
}

**Вывод:** изучили основные формы библиотеки свинг, реализовали простое приложение для вычисления интеграла.