Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе № 3

по дисциплине: «Программирование на языке Java»

на тему: «Обработка исключительных ситуаций»

Выполнили:

студенты группы 20ВВП1

Макарова А. Ю.

Тельнова А. Д.

Приняли:

к.т.н., доцент

Юрова О. В.

к.т.н., доцент  
Карамышева Н.С.

Пенза, 2023

# Цель работы

Изучить механизм обработки исключительных ситуаций.

# Лабораторное задание

Требуется реализовать проверку вводимых данных с использованием механизма исключений. Необходимо создать свой класс, унаследованный от класса Exception, и генерировать исключение, если возникает попытка создать экземпляр класса RecIntegral со значениями, не являющимися числами в диапазоне от 0,000001 до 1000000. В качестве обработки исключения необходимо выводить диалог, содержащий предупреждение о некорректности введенных данных.

# Описание работы программы

Создан класс SomeProblems, унаследованный от класса Exception. В случае, когда возникает исключение, будет выведено сообщение об ошибке. Текст сообщения об ошибке передается в переменную message.

В обработчике событий для кнопки «Добавить» реализована конструкция try-catch. При попытке создать экземпляр класса RecIntegral со значениями, не являющимися числами в диапазоне от 0,000001 до 1000000, будет сгенерировано исключение с помощью throws new с сообщением о некорректности введенных данных. В блоке catch реализована обработка исключения – поля для ввода данных очищаются.

В обработчике событий для кнопки «Вычислить» реализована конструкция try-catch. При попытке изменить введенные значения на числа, которые находятся вне допустимого диапазона, будет сгенерировано исключение с помощью throws new с сообщением о некорректности данных. В блоке catch реализована обработка исключения – поля с некорректными данными очищаются.

# Листинг программы

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import java.util.LinkedList;

public class JFrame extends javax.swing.JFrame {

public JFrame() {

initComponents();

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

private void initComponents() {

jScrollPane1 = new javax.swing.JScrollPane();

Table = new javax.swing.JTable();

AddButton = new javax.swing.JButton();

DeleteButton = new javax.swing.JButton();

CalcButton = new javax.swing.JButton();

LowTextField = new javax.swing.JTextField();

UpTextField = new javax.swing.JTextField();

StepTextField = new javax.swing.JTextField();

LowLabel = new javax.swing.JLabel();

UpLabel = new javax.swing.JLabel();

StepLabel = new javax.swing.JLabel();

ClearButton = new javax.swing.JButton();

FillButton = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jScrollPane1.setBackground(new java.awt.Color(255, 255, 255));

Table.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Нижняя граница", "Верхняя граница", "Шаг", "Результат"

}

) {

boolean[] canEdit = new boolean [] {

true, true, true, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

Table.setSelectionBackground(new java.awt.Color(255, 20, 147));

Table.setShowGrid(false);

Table.getTableHeader().setReorderingAllowed(false);

jScrollPane1.setViewportView(Table);

AddButton.setBackground(new java.awt.Color(255, 20, 147));

AddButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

AddButton.setForeground(new java.awt.Color(255, 255, 255));

AddButton.setText("Добавить");

AddButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

AddButtonActionPerformed(evt);

}

});

DeleteButton.setBackground(new java.awt.Color(255, 20, 147));

DeleteButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

DeleteButton.setForeground(new java.awt.Color(255, 255, 255));

DeleteButton.setText("Удалить");

DeleteButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

DeleteButtonActionPerformed(evt);

}

});

CalcButton.setBackground(new java.awt.Color(255, 20, 147));

CalcButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

CalcButton.setForeground(new java.awt.Color(255, 255, 255));

CalcButton.setText("Вычислить");

CalcButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

CalcButtonActionPerformed(evt);

}

});

LowTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

UpTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

StepTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

LowLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

LowLabel.setForeground(new java.awt.Color(204, 0, 102));

LowLabel.setText("Нижняя граница");

UpLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

UpLabel.setForeground(new java.awt.Color(204, 0, 102));

UpLabel.setText("Верхняя граница");

StepLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

StepLabel.setForeground(new java.awt.Color(204, 0, 102));

StepLabel.setText("Шаг");

ClearButton.setBackground(new java.awt.Color(255, 20, 147));

ClearButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

ClearButton.setForeground(new java.awt.Color(255, 255, 255));

ClearButton.setText("Очистить таблицу");

ClearButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

ClearButtonActionPerformed(evt);

}

});

FillButton.setBackground(new java.awt.Color(255, 20, 147));

FillButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

FillButton.setForeground(new java.awt.Color(255, 255, 255));

FillButton.setText("Заполнить таблицу");

FillButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

FillButtonActionPerformed(evt);

}

});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(0, 43, Short.MAX\_VALUE)

.addComponent(LowLabel)

.addGap(18, 18, 18)

.addComponent(LowTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 350, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(26, 26, 26)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jScrollPane1)

.addGroup(layout.createSequentialGroup()

.addComponent(AddButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(58, 58, 58)

.addComponent(DeleteButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(CalcButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

.addGap(0, 9, Short.MAX\_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(UpLabel, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(StepLabel, javax.swing.GroupLayout.Alignment.TRAILING))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

.addComponent(UpTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 350, Short.MAX\_VALUE)

.addComponent(StepTextField)))

.addGroup(layout.createSequentialGroup()

.addComponent(ClearButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 205, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(FillButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 205, javax.swing.GroupLayout.PREFERRED\_SIZE)))))

.addGap(24, 24, 24))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(14, 14, 14)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(LowTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 40, Short.MAX\_VALUE)

.addComponent(LowLabel))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(UpTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(UpLabel))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(StepTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(StepLabel))

.addGap(48, 48, 48)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(AddButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(DeleteButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(CalcButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(37, 37, 37)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 230, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(ClearButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(FillButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(19, 19, 19))

);

pack();

}// </editor-fold>//GEN-END:initComponents

LinkedList <RecIntegral> data = new LinkedList <>();

private void AddButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_AddButtonActionPerformed

if (LowTextField.getText().equals("") || UpTextField.getText().equals("") || StepTextField.getText().equals(""))

JOptionPane.showMessageDialog(this, "Заполните все поля данными!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

String l = LowTextField.getText();

String u = UpTextField.getText();

String s = StepTextField.getText();

try {

if(Double.valueOf(l) > 1000000 || Double.valueOf(l) < 0.000001){

throw new SomeProblems("Неверное значение нижнего предела!");

}

else if (Double.valueOf(u) > 1000000 || Double.valueOf(u) < 0.000001){

throw new SomeProblems("Неверное значение верхнего предела!");

}

else if (Double.valueOf(s) > 1000000 || Double.valueOf(s) < 0.000001){

throw new SomeProblems("Неверное значение шага интегрирования!");

}

else if (Double.valueOf(l)> Double.valueOf(u)|| Double.valueOf(s) == 0 || Double.valueOf(s) > Double.valueOf(u) - Double.valueOf(l)){

throw new SomeProblems("Что-то не то с данными!"); }

} catch (SomeProblems e) {

LowTextField.setText("");

UpTextField.setText("");

StepTextField.setText("");

return;

}

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

model.addRow(new Object[]{LowTextField.getText(),UpTextField.getText(),StepTextField.getText()});

RecIntegral object = new RecIntegral();

object.NewNode(l, u, s);

data.add(object);

LowTextField.setText("");

UpTextField.setText("");

StepTextField.setText("");

//}

}//GEN-LAST:event\_AddButtonActionPerformed

private void DeleteButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_DeleteButtonActionPerformed

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

int SelectedRow = Table.getSelectedRow();

if (Table.getSelectedRowCount() == 1 ){

model.removeRow(Table.getSelectedRow());

data.remove(SelectedRow);

}

else

JOptionPane.showMessageDialog(this, "Выберите строку, которую нужно удалить!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}//GEN-LAST:event\_DeleteButtonActionPerformed

private void CalcButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_CalcButtonActionPerformed

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

if (Table.getSelectedRowCount() == 1 )

{

double IntegralFunc = 0;

double low = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),0).toString());

double up = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),1).toString());

double step = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),2).toString());

try {

if(low > up || step == 0 || low > 1000000 || low < 0.000001 || up > 1000000 || up < 0.000001 || step > 1000000 || step < 0.000001 || step > up - low)

throw new SomeProblems("Вы некорректно изменили данные в таблице!");

} catch (SomeProblems e) {

model.setValueAt(null,Table.getSelectedRow(),0);

model.setValueAt(null,Table.getSelectedRow(),2);

return;

}

for (double i = low; i < up - step; i+= step)

{

if (up - low > step)

IntegralFunc += ((Math.sin(i) + Math.sin(i+step))/2)\*step;

else

IntegralFunc += ((Math.sin(up) + Math.sin(i-step))/2)\*step;

}

model.setValueAt(IntegralFunc, Table.getSelectedRow(), 3);

}

else

JOptionPane.showMessageDialog(this, "Выберите строку для вычисления!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}//GEN-LAST:event\_CalcButtonActionPerformed

private void FillButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_FillButtonActionPerformed

DefaultTableModel module = (DefaultTableModel)Table.getModel();

if (!data.isEmpty())

{

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

{

RecIntegral object = data.get(i);

module.addRow(new Object[]{object.low, object.up, object.step, null});

}

}

else

JOptionPane.showMessageDialog(this, "Linked List пуст!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}//GEN-LAST:event\_FillButtonActionPerformed

private void ClearButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_ClearButtonActionPerformed

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

if (Table.getRowCount() != 0 ){

model.setRowCount(0);

}

else

JOptionPane.showMessageDialog(this, "Таблица и так пустая!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}//GEN-LAST:event\_ClearButtonActionPerformed

public class RecIntegral {

public String low;

public String up;

public String step;

public void NewNode (String low, String up, String step)

{

this.low = low;

this.up = up;

this.step = step;

}

}

static class SomeProblems extends Exception{

public SomeProblems(String message) {

JOptionPane.showMessageDialog(null, message, "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}

}

public static void main(String args[]) {

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

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

public void run() {

JFrame frame = new JFrame();

frame.setVisible(true);

frame.setLocationRelativeTo(null);

frame.setTitle("Вычисляем интегральчик");

frame.getContentPane().setBackground(new java.awt.Color(255, 228, 225));

}

});

}

// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton AddButton;

private javax.swing.JButton CalcButton;

private javax.swing.JButton ClearButton;

private javax.swing.JButton DeleteButton;

private javax.swing.JButton FillButton;

private javax.swing.JLabel LowLabel;

private javax.swing.JTextField LowTextField;

private javax.swing.JLabel StepLabel;

private javax.swing.JTextField StepTextField;

private javax.swing.JTable Table;

private javax.swing.JLabel UpLabel;

private javax.swing.JTextField UpTextField;

private javax.swing.JScrollPane jScrollPane1;

// End of variables declaration//GEN-END:variables

}

# Результат работы программы

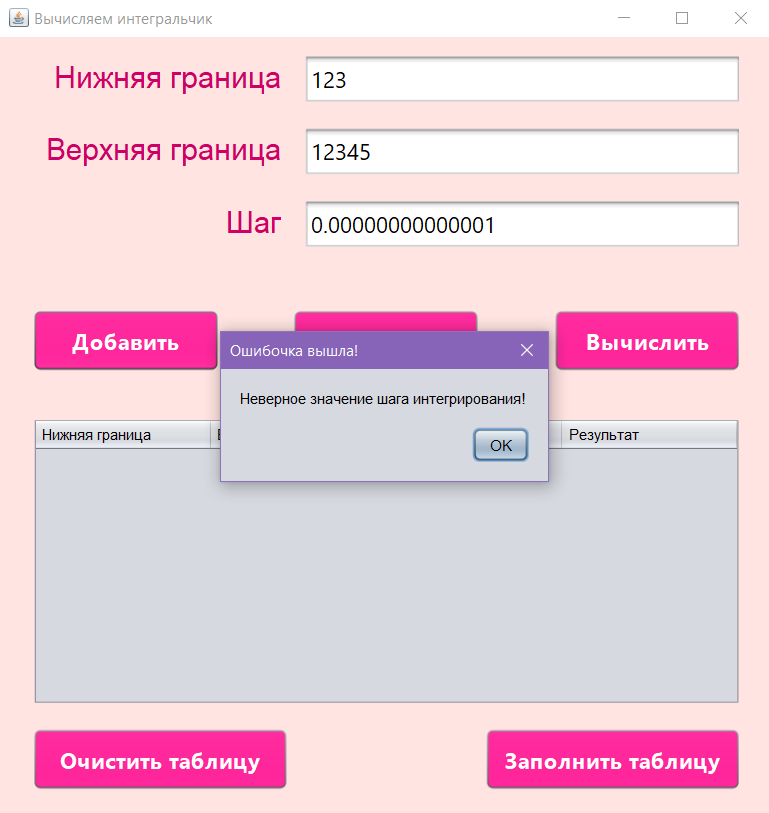


Рисунок – Сообщение об ошибке при попытке ввести значение шага интегрирования

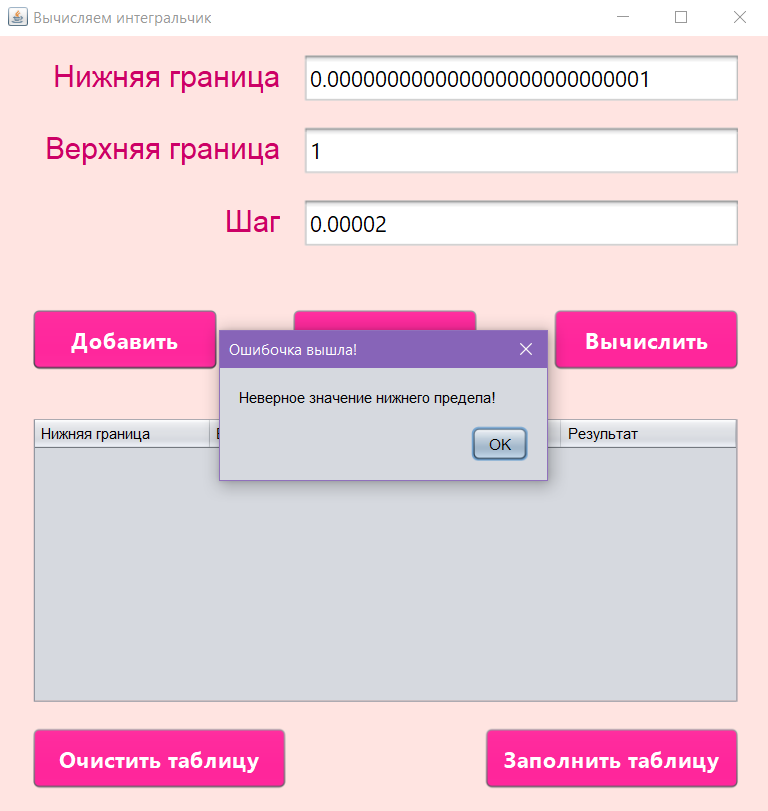


Рисунок – Сообщение об ошибке при попытке ввести значение нижнего предела интегрирования

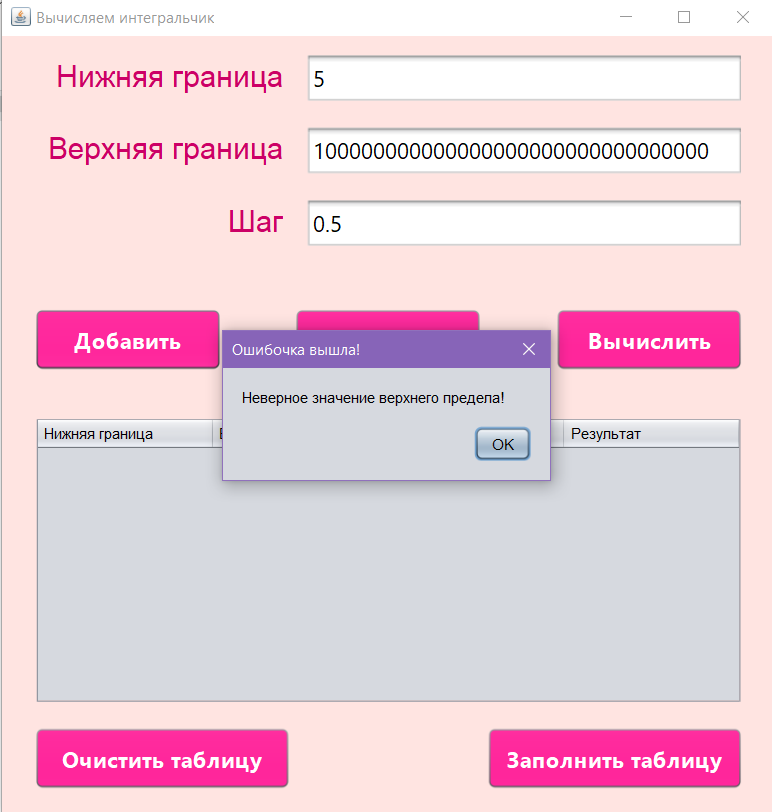
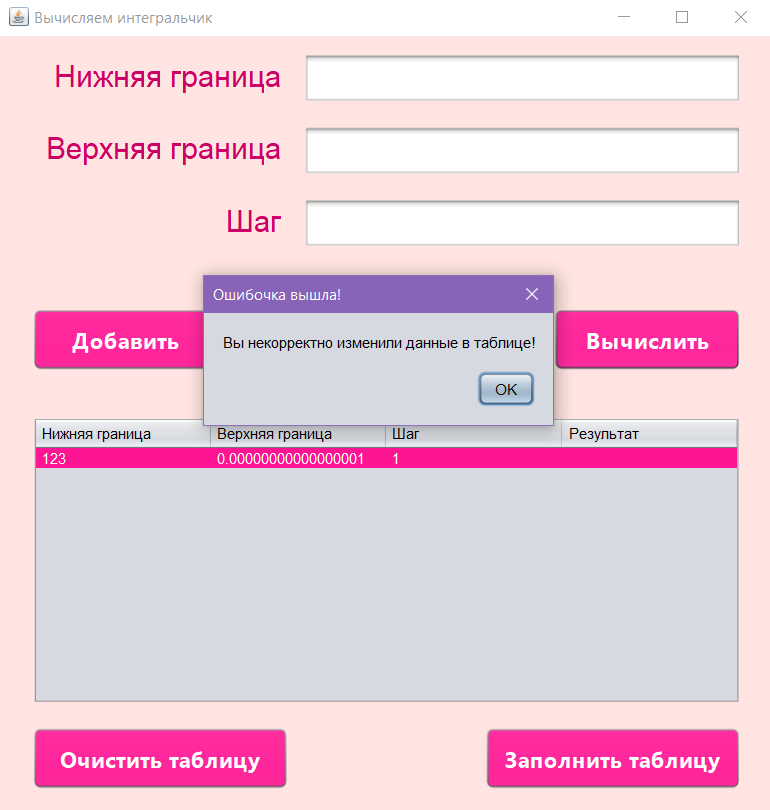


Рисунок – Сообщение об ошибке при попытке ввести значение верхнего предела интегрирования



**Рисунок 4 – Сообщение об ошибке при попытке изменить данные в таблице**

# Вывод

Изучен механизм обработки исключительных ситуаций.

Модифицировано приложение из лабораторной работы №2. Создан класс, унаследованный от класса Exception. При попытке создать экземпляр класса RecIntegral со значениями, не являющимися числами в диапазоне от 0,000001 до 1000000 генерируется исключение. В качестве обработки исключения выводится диалог, содержащий предупреждение о некорректности введенных данных.