Министерство науки и образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №3

по курсу «Программирование на Java»

на тему «Обработка исключений»

Вариант 3

Выполнили студенты группы 19ВВ3:

Ланцов Андрей

Трубаненко Александр

Приняли:

Юрова О.В.

**2022**

**Цель работы:** изучить механизм обработки исключительных ситуаций.

**Задание на лабораторную работу:**

Модифицировать приложение из предыдущей лабораторной работы, реализовав 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. Оформление лабораторной работы должно быть выполнено в соответствии с требованиями, приведенными в Приложении 2.

**Ход работы:**

1. Добавили диалоговое окно в конструкторе
2. Cоздали класс RecIntegral для обработки ошибки

**package** Lab1;  
  
**public class** RecIntegral{  
 **public static final double *MAX*** = 1000000;  
 **public static final double *MIN*** = 0.000001;  
  
 **public double upperEdge**;  
 **public double lowerEdge**;  
 **public double step**;  
   
 RecIntegral(**double** upperEdge,**double** lowerEdge,**double** step) **throws** RecIntegralException {  
 **if** (  
 Double.*compare*(lowerEdge,***MIN***) < 0 ||  
 Double.*compare*(upperEdge,***MIN***) < 0 ||  
 Double.*compare*(step,***MIN***) < 0)  
 **throw new** RecIntegralExceptionLowerBound();  
  
 **if** (  
 Double.*compare*(lowerEdge,***MAX***) > 0 ||  
 Double.*compare*(upperEdge,***MAX***) > 0 ||  
 Double.*compare*(step,***MAX***) > 0)  
 **throw new** RecIntegralExceptionUpperBound();  
  
 **if** (Double.*compare*(lowerEdge,upperEdge) > 0)  
 **throw new** RecIntegralExceptionRange();  
  
 **this**.**upperEdge** = upperEdge;  
 **this**.**lowerEdge** = lowerEdge;  
 **this**.**step** = step;  
 }  
}

**Результат работы**

Ввод значения меньше 0,000001

Изображение выглядит как текст

Автоматически созданное описание

Ввод значения больше

Изображение выглядит как текст

Автоматически созданное описание

**Листинг**

GUI.java

**package** Lab1;  
  
**import** javax.swing.table.DefaultTableModel;  
**import** javax.swing.JOptionPane;  
  
*/\*  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template  
 \*/  
  
/\*\*  
 \*  
 \** ***@author*** *tront  
 \*/***public class** GUI **extends** javax.swing.JFrame {  
  
 */\*\*  
 \* Creates new form GUI  
 \*/* **public** GUI() {  
 initComponents();  
 }  
  
 */\*\*  
 \* This method is called from within the constructor to initialize the form.  
 \* WARNING: Do NOT modify this code. The content of this method is always  
 \* regenerated by the Form Editor.  
 \*/* @SuppressWarnings(**"unchecked"**)  
 *// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents* **private void** initComponents() {  
  
 **jDialog1** = **new** javax.swing.JDialog();  
 **jLabel4** = **new** javax.swing.JLabel();  
 **jButton6** = **new** javax.swing.JButton();  
 **jDialog2** = **new** javax.swing.JDialog();  
 **jScrollPane1** = **new** javax.swing.JScrollPane();  
 **jTable1** = **new** javax.swing.JTable();  
 **Add** = **new** javax.swing.JButton();  
 **Del** = **new** javax.swing.JButton();  
 **Calculate** = **new** javax.swing.JButton();  
 **jTextField1** = **new** javax.swing.JTextField();  
 **jTextField2** = **new** javax.swing.JTextField();  
 **jTextField3** = **new** javax.swing.JTextField();  
 **Fill** = **new** javax.swing.JButton();  
 **Clear** = **new** javax.swing.JButton();  
 **jLabel1** = **new** javax.swing.JLabel();  
 **jLabel2** = **new** javax.swing.JLabel();  
 **jLabel3** = **new** javax.swing.JLabel();  
  
 **jLabel4**.setFont(**new** java.awt.Font(**"Segoe UI"**, 0, 24)); *// NOI18N* **jLabel4**.setText(**"Ошибочка)"**);  
 **jLabel4**.setToolTipText(**""**);  
  
 **jButton6**.setText(**"Закрыть"**);  
 **jButton6**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 jButton6ActionPerformed(evt);  
 }  
 });  
  
 javax.swing.GroupLayout jDialog1Layout = **new** javax.swing.GroupLayout(**jDialog1**.getContentPane());  
 **jDialog1**.getContentPane().setLayout(jDialog1Layout);  
 jDialog1Layout.setHorizontalGroup(  
 jDialog1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addGroup(javax.swing.GroupLayout.Alignment.***TRAILING***, jDialog1Layout.createSequentialGroup()  
 .addContainerGap(70, Short.***MAX\_VALUE***)  
 .addGroup(jDialog1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***, **false**)  
 .addComponent(**jLabel4**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addComponent(**jButton6**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***))  
 .addGap(70, 70, 70))  
 );  
 jDialog1Layout.setVerticalGroup(  
 jDialog1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addGroup(jDialog1Layout.createSequentialGroup()  
 .addGap(35, 35, 35)  
 .addComponent(**jLabel4**)  
 .addGap(28, 28, 28)  
 .addComponent(**jButton6**)  
 .addContainerGap(35, Short.***MAX\_VALUE***))  
 );  
  
 javax.swing.GroupLayout jDialog2Layout = **new** javax.swing.GroupLayout(**jDialog2**.getContentPane());  
 **jDialog2**.getContentPane().setLayout(jDialog2Layout);  
 jDialog2Layout.setHorizontalGroup(  
 jDialog2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addGap(0, 400, Short.***MAX\_VALUE***)  
 );  
 jDialog2Layout.setVerticalGroup(  
 jDialog2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addGap(0, 300, Short.***MAX\_VALUE***)  
 );  
  
 setDefaultCloseOperation(javax.swing.WindowConstants.***EXIT\_ON\_CLOSE***);  
  
 **jTable1**.setModel(**new** javax.swing.table.DefaultTableModel(  
 **new** Object [][] {  
  
 },  
 **new** String [] {  
 **"Нижняя граница"**, **"Верхняя граница"**, **"Шаг интегрирования"**, **"Результат"** }  
 ));  
 **jScrollPane1**.setViewportView(**jTable1**);  
  
 **Add**.setText(**"Добавить"**);  
 **Add**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 AddActionPerformed(evt);  
 }  
 });  
  
 **Del**.setText(**"Удалить"**);  
 **Del**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 DelActionPerformed(evt);  
 }  
 });  
  
 **Calculate**.setText(**"Вычислить"**);  
 **Calculate**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 CalculateActionPerformed(evt);  
 }  
 });  
  
 **jTextField1**.setText(**"0"**);  
 **jTextField1**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 jTextField1ActionPerformed(evt);  
 }  
 });  
  
 **jTextField2**.setText(**"0"**);  
  
 **jTextField3**.setText(**"0"**);  
  
 **Fill**.setText(**"Заполнить"**);  
 **Fill**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 FillActionPerformed(evt);  
 }  
 });  
  
 **Clear**.setText(**"Очистить"**);  
 **Clear**.addActionListener(**new** java.awt.event.ActionListener() {  
 **public void** actionPerformed(java.awt.event.ActionEvent evt) {  
 ClearActionPerformed(evt);  
 }  
 });  
  
 **jLabel1**.setText(**"Нижняя"**);  
  
 **jLabel2**.setText(**"Верхняя"**);  
  
 **jLabel3**.setText(**"Шаг"**);  
  
 javax.swing.GroupLayout layout = **new** javax.swing.GroupLayout(getContentPane());  
 getContentPane().setLayout(layout);  
 layout.setHorizontalGroup(  
 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addComponent(**jScrollPane1**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, 1030, Short.***MAX\_VALUE***)  
 .addGroup(layout.createSequentialGroup()  
 .addGap(90, 90, 90)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***, **false**)  
 .addComponent(**jLabel1**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addComponent(**Fill**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addComponent(**Add**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, 100, Short.***MAX\_VALUE***)  
 .addComponent(**jTextField1**))  
 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.***RELATED***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***, **false**)  
 .addComponent(**Del**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, 100, Short.***MAX\_VALUE***)  
 .addComponent(**jTextField2**)  
 .addComponent(**Clear**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addComponent(**jLabel2**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***))  
 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.***RELATED***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***, **false**)  
 .addComponent(**Calculate**, javax.swing.GroupLayout.***DEFAULT\_SIZE***, 100, Short.***MAX\_VALUE***)  
 .addComponent(**jTextField3**)  
 .addComponent(**jLabel3**, javax.swing.GroupLayout.Alignment.***TRAILING***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***))  
 .addContainerGap(javax.swing.GroupLayout.***DEFAULT\_SIZE***, Short.***MAX\_VALUE***))  
 );  
 layout.setVerticalGroup(  
 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***LEADING***)  
 .addGroup(layout.createSequentialGroup()  
 .addComponent(**jScrollPane1**, javax.swing.GroupLayout.***PREFERRED\_SIZE***, 283, javax.swing.GroupLayout.***PREFERRED\_SIZE***)  
 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.***RELATED***)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***BASELINE***)  
 .addComponent(**jTextField2**, javax.swing.GroupLayout.***PREFERRED\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***PREFERRED\_SIZE***)  
 .addComponent(**jTextField3**, javax.swing.GroupLayout.***PREFERRED\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***PREFERRED\_SIZE***)  
 .addComponent(**jTextField1**, javax.swing.GroupLayout.***PREFERRED\_SIZE***, javax.swing.GroupLayout.***DEFAULT\_SIZE***, javax.swing.GroupLayout.***PREFERRED\_SIZE***))  
 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.***UNRELATED***)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***BASELINE***)  
 .addComponent(**jLabel1**)  
 .addComponent(**jLabel2**, javax.swing.GroupLayout.***PREFERRED\_SIZE***, 16, javax.swing.GroupLayout.***PREFERRED\_SIZE***)  
 .addComponent(**jLabel3**))  
 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.***RELATED***)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***BASELINE***)  
 .addComponent(**Add**)  
 .addComponent(**Del**)  
 .addComponent(**Calculate**))  
 .addGap(66, 66, 66)  
 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.***BASELINE***)  
 .addComponent(**Fill**)  
 .addComponent(**Clear**))  
 .addContainerGap(111, Short.***MAX\_VALUE***))  
 );  
  
 pack();  
 }*// </editor-fold>//GEN-END:initComponents* **private void** AddActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_AddActionPerformed  
 //* ***TODO add your handling code here:*** DefaultTableModel dt = (DefaultTableModel) **jTable1**.getModel();  
  
 String upper = **jTextField1**.getText();  
 String lower = **jTextField2**.getText();  
 String step = **jTextField3**.getText();  
  
 RecIntegral rec;  
 **try**{  
 rec = **new** RecIntegral(  
 Double.*parseDouble*(upper),  
 Double.*parseDouble*(lower),  
 Double.*parseDouble*(step)  
 );  
 }**catch** (RecIntegralException e){  
 JOptionPane.*showMessageDialog*(**null**,e.getInfo());  
 **return**;  
 }  
  
 dt.addRow(**new** Object[]{upper,lower,step});  
 Globals.*tableRecs*.add(rec);  
 }*//GEN-LAST:event\_AddActionPerformed* **private void** jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_jTextField1ActionPerformed  
 //* ***TODO add your handling code here:*** }*//GEN-LAST:event\_jTextField1ActionPerformed* **private void** DelActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_DelActionPerformed* DefaultTableModel dt = (DefaultTableModel) **jTable1**.getModel();  
 **int** rowIndex = **jTable1**.getSelectedRow();  
 **if** (rowIndex != -1){  
 dt.removeRow(rowIndex);   
 }  
 **if** (Globals.*tableRecs*.size() != 0){  
 Globals.*tableRecs*.remove(Globals.*tableRecs*.size()-1);  
 }  
  
 }*//GEN-LAST:event\_DelActionPerformed* **private void** CalculateActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_CalculateActionPerformed* **int** rowIndex = **jTable1**.getSelectedRow();  
 DefaultTableModel dt = (DefaultTableModel) **jTable1**.getModel();  
   
 **double** from = Double.*parseDouble*(**jTable1**.getValueAt(rowIndex,0).toString());  
 **double** to = Double.*parseDouble*(**jTable1**.getValueAt(rowIndex,1).toString());  
 **double** step = Double.*parseDouble*(**jTable1**.getValueAt(rowIndex,2).toString());  
 String result = String.*valueOf*(TrapIntergration.*integrate\_cos*(from, to, step));  
 dt.setValueAt(result, rowIndex,3);  
 }*//GEN-LAST:event\_CalculateActionPerformed* **private void** ClearActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_ClearActionPerformed* DefaultTableModel dt = (DefaultTableModel) **jTable1**.getModel();  
 *//int tableSize = dt.getRowCount();* **while** (dt.getRowCount() != 0){  
 dt.removeRow(0);  
 }  
  
 }*//GEN-LAST:event\_ClearActionPerformed* **private void** FillActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_FillActionPerformed* DefaultTableModel dt = (DefaultTableModel) **jTable1**.getModel();  
  
 **for**(RecIntegral it : Globals.*tableRecs*){  
 dt.addRow(**new** Object[]{  
 Double.*toString*(it.**upperEdge**),  
 Double.*toString*(it.**lowerEdge**),  
 Double.*toString*(it.**step**)  
 });  
 }  
 }*//GEN-LAST:event\_FillActionPerformed* **private void** jButton6ActionPerformed(java.awt.event.ActionEvent evt) {*//GEN-FIRST:event\_jButton6ActionPerformed  
 //* ***TODO add your handling code here:*** }*//GEN-LAST:event\_jButton6ActionPerformed  
  
 /\*\*  
 \** ***@param args*** *the command line arguments  
 \*/* **public static void** main(String args[]) {  
 */\* Set the Nimbus look and feel \*/  
 //<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*(GUI.**class**.getName()).log(java.util.logging.Level.***SEVERE***, **null**, ex);  
 } **catch** (InstantiationException ex) {  
 java.util.logging.Logger.*getLogger*(GUI.**class**.getName()).log(java.util.logging.Level.***SEVERE***, **null**, ex);  
 } **catch** (IllegalAccessException ex) {  
 java.util.logging.Logger.*getLogger*(GUI.**class**.getName()).log(java.util.logging.Level.***SEVERE***, **null**, ex);  
 } **catch** (javax.swing.UnsupportedLookAndFeelException ex) {  
 java.util.logging.Logger.*getLogger*(GUI.**class**.getName()).log(java.util.logging.Level.***SEVERE***, **null**, ex);  
 }  
 *//</editor-fold>  
   
 /\* Create and display the form \*/* java.awt.EventQueue.*invokeLater*(**new** Runnable() {  
 **public void** run() {  
 **new** GUI().setVisible(**true**);  
 }  
 });  
 }  
  
 *// Variables declaration - do not modify//GEN-BEGIN:variables* **private** javax.swing.JButton **Add**;  
 **private** javax.swing.JButton **Calculate**;  
 **private** javax.swing.JButton **Clear**;  
 **private** javax.swing.JButton **Del**;  
 **private** javax.swing.JButton **Fill**;  
 **private** javax.swing.JButton **jButton6**;  
 **private** javax.swing.JDialog **jDialog1**;  
 **private** javax.swing.JDialog **jDialog2**;  
 **private** javax.swing.JLabel **jLabel1**;  
 **private** javax.swing.JLabel **jLabel2**;  
 **private** javax.swing.JLabel **jLabel3**;  
 **private** javax.swing.JLabel **jLabel4**;  
 **private** javax.swing.JScrollPane **jScrollPane1**;  
 **private** javax.swing.JTable **jTable1**;  
 **private** javax.swing.JTextField **jTextField1**;  
 **private** javax.swing.JTextField **jTextField2**;  
 **private** javax.swing.JTextField **jTextField3**;  
 *// End of variables declaration//GEN-END:variables*}

Recintegral.java

*/\*  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template  
 \*/***package** Lab1;  
  
**public class** RecIntegral{  
 **public static final double *MAX*** = 1000000;  
 **public static final double *MIN*** = 0.000001;  
  
 **public double upperEdge**;  
 **public double lowerEdge**;  
 **public double step**;  
   
 RecIntegral(**double** upperEdge,**double** lowerEdge,**double** step) **throws** RecIntegralException {  
 **if** (  
 Double.*compare*(lowerEdge,***MIN***) < 0 ||  
 Double.*compare*(upperEdge,***MIN***) < 0 ||  
 Double.*compare*(step,***MIN***) < 0)  
 **throw new** RecIntegralExceptionLowerBound();  
  
 **if** (  
 Double.*compare*(lowerEdge,***MAX***) > 0 ||  
 Double.*compare*(upperEdge,***MAX***) > 0 ||  
 Double.*compare*(step,***MAX***) > 0)  
 **throw new** RecIntegralExceptionUpperBound();  
  
 **if** (Double.*compare*(lowerEdge,upperEdge) > 0)  
 **throw new** RecIntegralExceptionRange();  
  
 **this**.**upperEdge** = upperEdge;  
 **this**.**lowerEdge** = lowerEdge;  
 **this**.**step** = step;  
 }  
}

RecIntegralException.java

**package** Lab1;  
  
**public abstract class** RecIntegralException **extends** Exception {  
 RecIntegralException() {  
 **super**();  
 }  
 **abstract** String getInfo();  
}

RecIntegralExceptionLowerBound.java

**package** Lab1;  
  
**public class** RecIntegralExceptionLowerBound **extends** RecIntegralException {  
 **private static final** String ***info*** = **new** String(**"Значения не могут быть меньше "**+RecIntegral.***MIN***);  
 RecIntegralExceptionLowerBound(){  
 **super**();  
 }  
 String getInfo(){  
 **return *info***;  
 }  
}

RecIntegralExceptionUpperBound.java

**package** Lab1;  
  
**public class** RecIntegralExceptionUpperBound **extends** RecIntegralException {  
 **private static final** String ***info*** = **new** String(**"Значения не могут быть больше "**+RecIntegral.***MAX***);  
 RecIntegralExceptionUpperBound() {  
 **super**();  
 }  
 String getInfo(){  
 **return *info***;  
 }  
}

RecIntegralExceptionRange.java

**package** Lab1;  
  
**public class** RecIntegralExceptionRange **extends** RecIntegralException {  
 **private static final** String ***info*** = **new** String(**"Нижняя граница не может быть больше верхней"**);  
 RecIntegralExceptionRange() {  
 **super**();  
 }  
 String getInfo(){  
 **return *info***;  
 }  
}

Trapintegretion.java

*/\*  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  
 \* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template  
 \*/***package** Lab1;  
**import static** java.lang.Math.\*;  
*/\*\*  
 \*  
 \** ***@author*** *Mike Lane  
 \*/***public class** TrapIntergration {  
 **public static double** integrate\_cos(**double** from,**double** to,**double** step) {  
 **int** iterations\_count = (**int**)((*abs*(from)+*abs*(to))/step);  
 **double** result = 0;  
 result += *abs*(*cos*(from));  
 **for** (**int** i = 1; i < iterations\_count-1; i++) {  
 result += 2\**abs*(*cos*(from+step\*i));  
 }  
 result += *cos*(to);  
 **return** result\*(step/2);  
 }   
}

**Вывод:**

Изучили механизмы обработки исключительных ситуаций.