Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Санкт-Петербургский государственный электротехнический университет

“ЛЭТИ” им.В.И.Ульянова (Ленина) »

Кафедра МОЭВМ

**ОТЧЕТ**

**по лабораторно-практической работе № 2,3,4,5**

**«Разработка интерфейса пользователя»**

**по дисциплине «Объектно - ориентированное программирование на языке Java»**

Выполнил Бондаренко Р.В.

Факультет КТИ

Группа № 1307

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_

Санкт-Петербург

2013 г

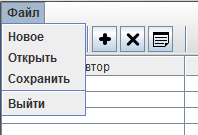
**Цель работы**

1. Знакомство с правилами построения экранной формы.
2. Обработка событий.
3. Обработка исключений.
4. Сохранение и загрузка данных из файла.

**Описание экранной формы**

Расположение и состав в экранная форме.

1. Вверху экранной формы находится строка меню JMenuBar, которая содержит меню JMenu “Файл”. В пункте “Файл” содержится элементы строк меню JMenuItem ( “Новое”, “Открыть”, “Сохранить” и ”Выйти”).



1. Инструментальная панель JToolBar размещена менеджером граничного размещения BorderLayout .NORTH сразу под меню в экранной форме. Она состоит из трёх box:
2. Первый box1 расположен с левой стороны и состоит из кнопок JButon ( "Сохранить список книг", "Открыть список книг", "Добавить информацию", "Редактировать", "Удалить информацию о книге" и "Профиль администратора").
3. Второй box2 расположен по центру и состоит из метки JLabel “Поиск по:”,

выпадающего списка JComboBox, текстового поля JTextField и кнопки “Поиск”.

1. Третий бок расположен с правой стороны и состоит из метки JLabel “Сортировка по” и выпадающего списка JComboBox.

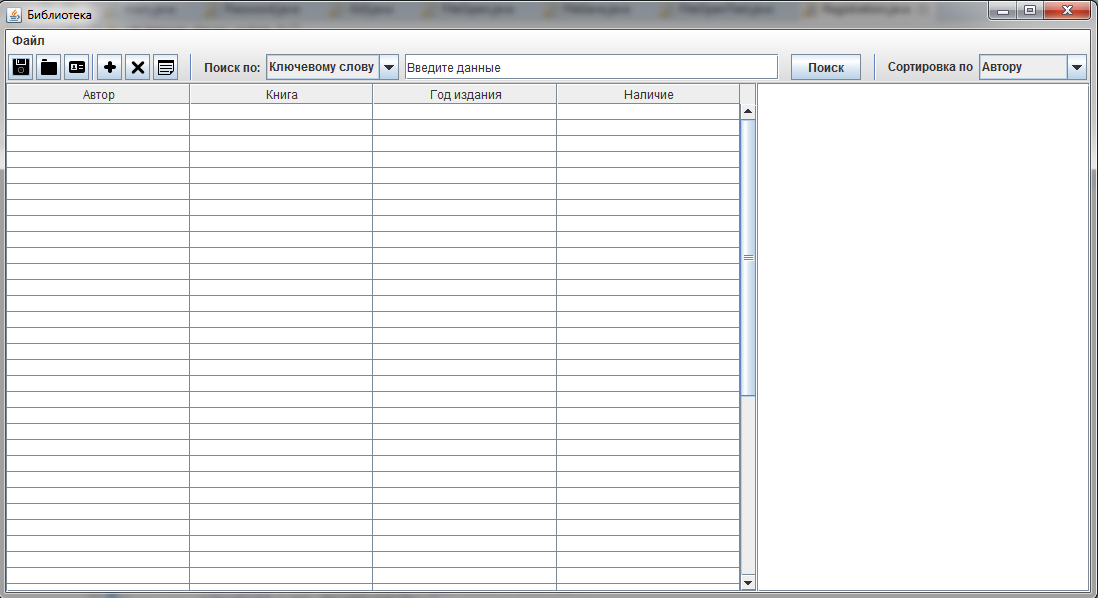
Все три box также размещены в JToolBar менеджером граничного размещения BorderLayout .



1. Область для ввода текста JTextArea размещено менеджером граничного размещения BorderLayout .EAST c права на экранной форме.
2. Таблица JTable состоит из четырёх столбцов("Автор","Книга","Год издания","Наличие") . Таблица находится в центре (на всем свободном месте) экранной формы.

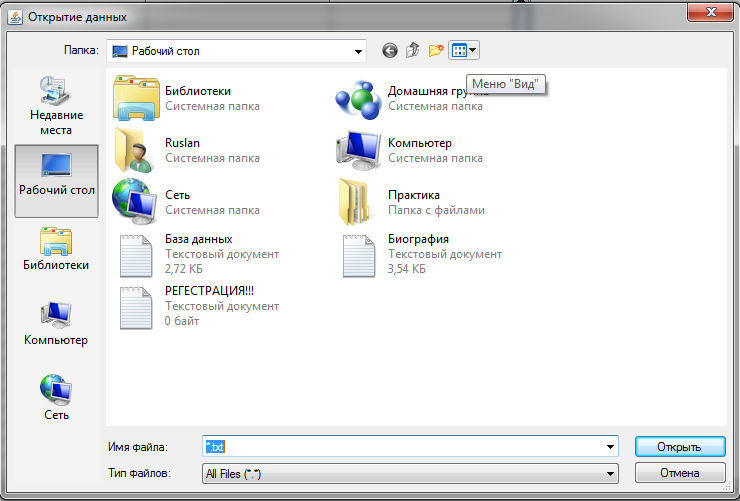
Макет экранной формы представлен на рисунке 1.

Рисунок 1

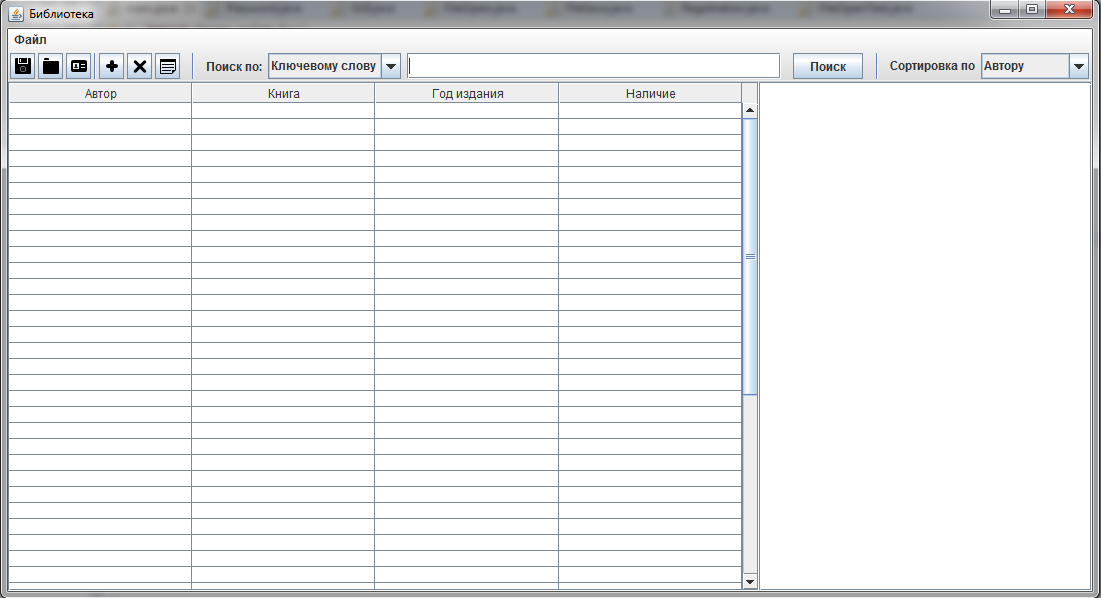


**Описание действий реализованными слушателями**

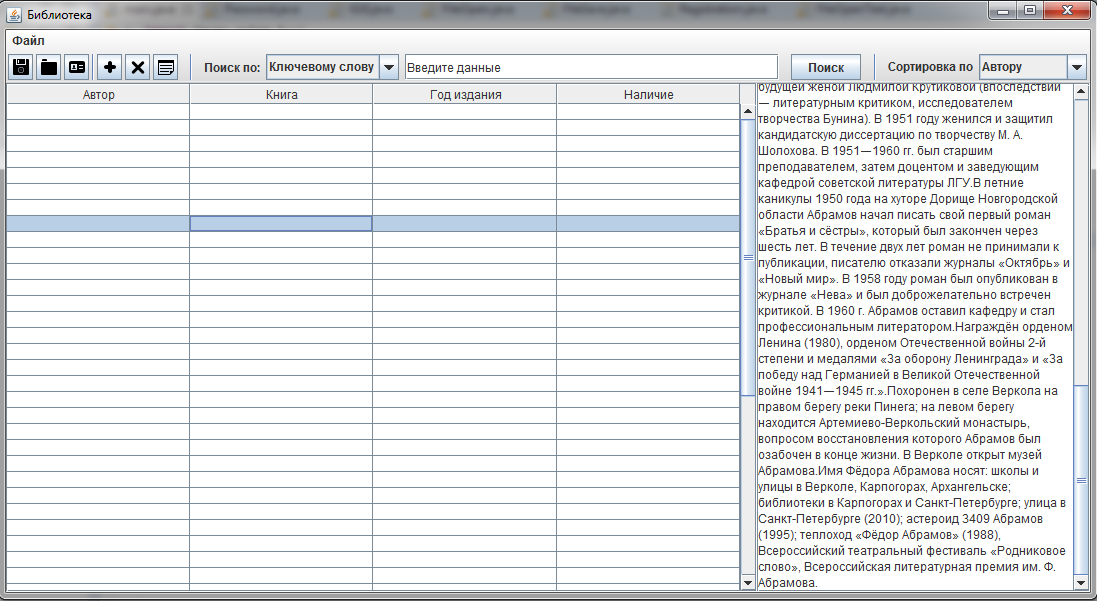
Источники событии сomboBox, extMenuIt, button\_open, button\_save будут вызвать интерфейс слушателя ActionListener метода actionPerformed. При нажатии на сomboBox - в search\_field выведится сообщение, extMenuIt - закроется программа, button\_open - откроется окно для загрузки файла в таблицу, button\_save - откроется окно для сохранения файла.



Источники событии search\_field будет вызван интерфейс слушателя MouseListener метода mouseClicked. При нажатии правой кнопки мыши на области search\_field, будет удалена строка "Введите данные" (другие строки удаляться не будут).

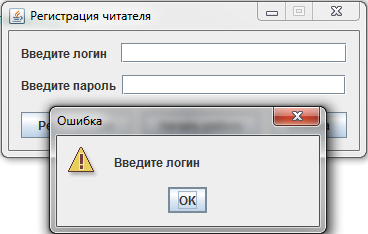


Источники событии на selModel будет вызван интерфейс слушателя ListSelectionListener метода valueChanged. При нажатии на любую строку таблицы, в textArea будет загружены данные из определенного файла.

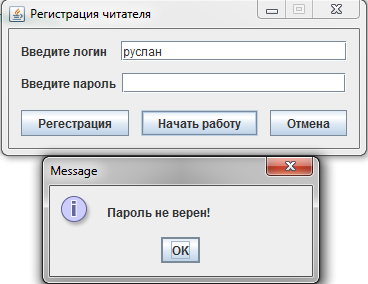


**Перечень ситуаций, которые контролируются с помощью исключений**

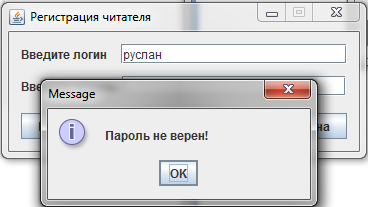
1. Если нажали в классе Password на кнопку button\_ok и при этом поле FieldLogin пустое, то сработает исключение NullPointerException.



1. Если нажали в классе Password на кнопку button\_ok и при этом поле FieldLogin заполнено, а поле passwField нет, то сработает исключение myException.



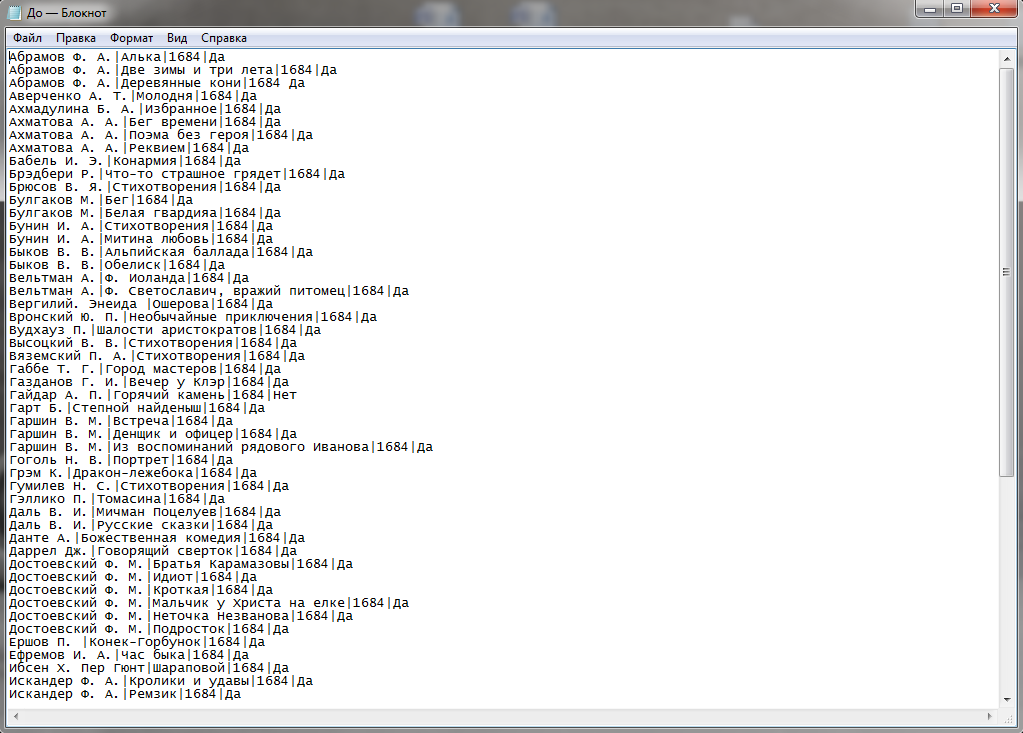
1. Если нажали в классе Password на кнопку button\_ok и при этом поля FieldLogin и passwField заполнены, но поле passwField не соответствует паролю, тогда срабатывает исключение myException.



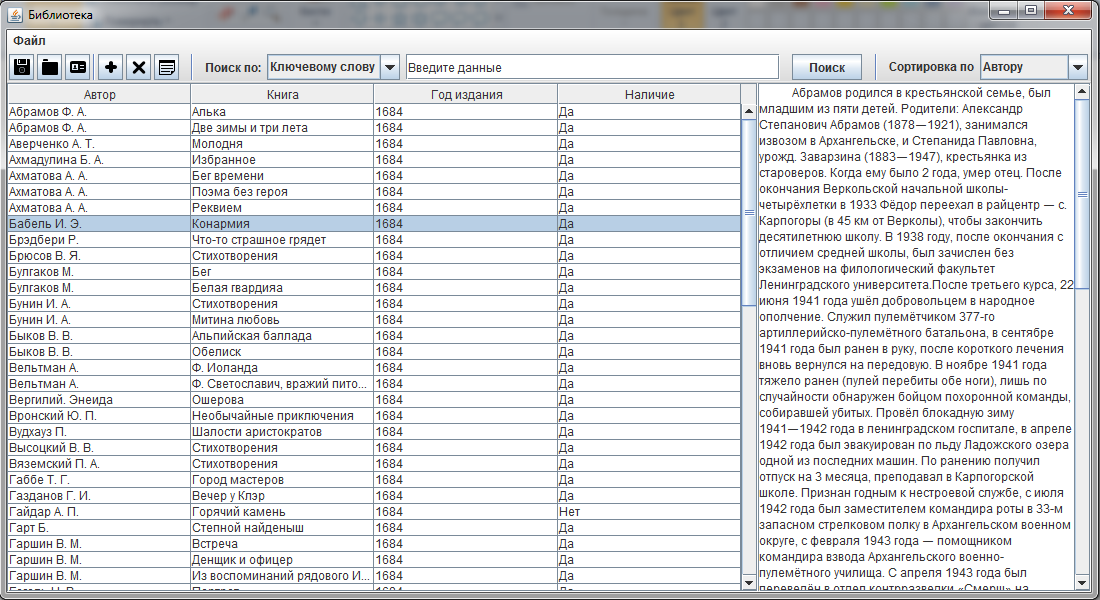
1. В классе FileOpen если не найден файл, который нужна открыть, то срабатывает исключение FileNotFoundException, если же произошла ошибка при считывание файла, то срабатывает исключение IOExceptionю.
2. В классе FileSave если произошла ошибка при записи в файл, то срабатывает исключение IOExceptionю.

**Скриншоты, иллюстрирующие процесс загрузки данных в файл и выгрузки из него.**

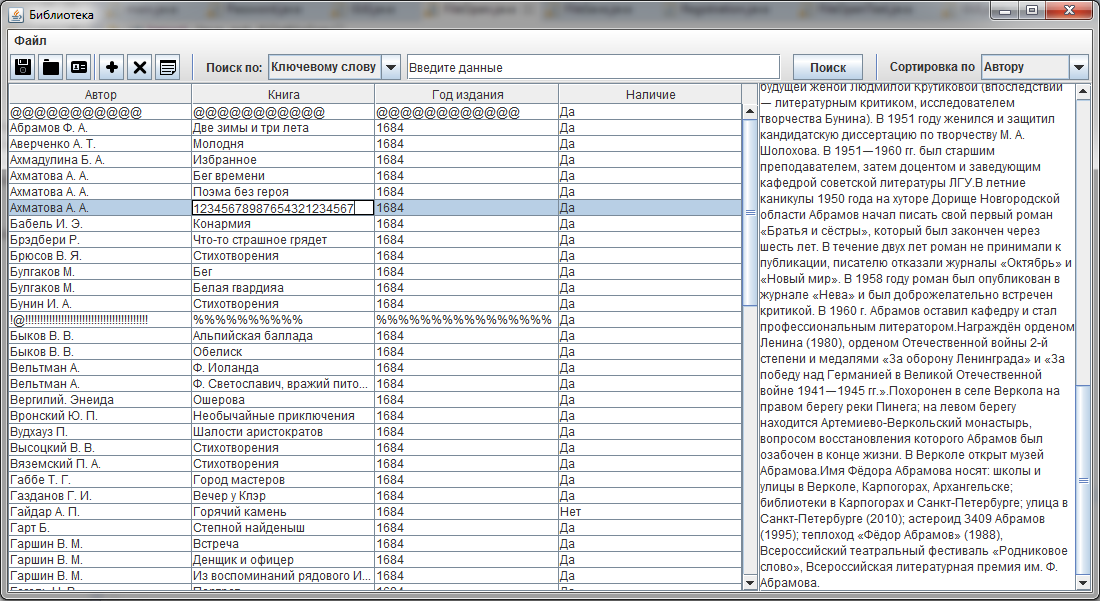
Загрузка из файла «До.txt»



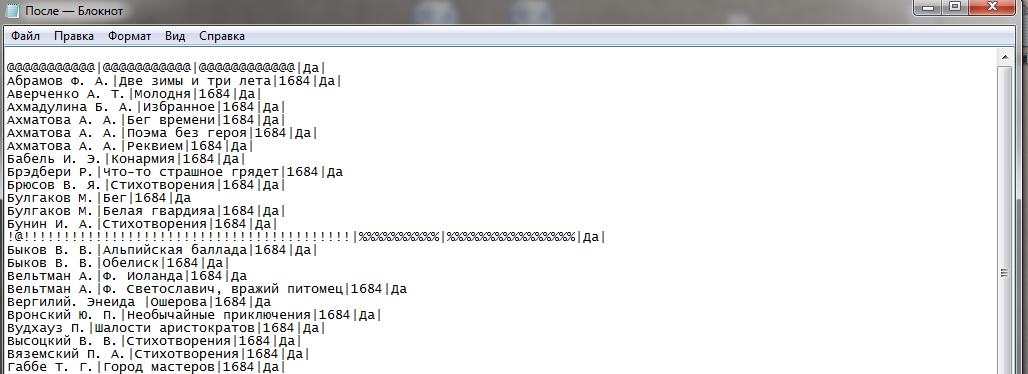
После загрузки из файла «До.txt»



После изменения значений в таблице



Сохраняем измененную таблицу в файлу «После.txt»



**Текст программы**

///////////////////////// Class GUI ////////////////////////////////

import javax.swing.Box;

import javax.swing.ImageIcon;

import javax.swing.JComponent;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.border.EmptyBorder;

import javax.swing.event.ListSelectionEvent;

import javax.swing.event.ListSelectionListener;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.MouseEvent;

import java.awt.event.MouseListener;

import javax.swing.\*;

public class GUI extends JFrame{

//Объявляем массивы

String elements[];

Object headers[];

Object data[][];

String sorting[];

// Объявляем контейнеры

JMenuBar menuBar;

JMenu fileMenu;

JMenuItem saveMenuIt,openMenuIt,newMenuIt,extMenuIt;

JTable table;

DefaultTableModel tableModel;

JLabel search\_label,sortingLabel;

JComboBox сomboBox,sortingCombolBox;

JToolBar toolBar;

JTextField search\_field;

JButton button\_search;

JButton button\_save,button\_open,button\_add,button\_edit,button\_delete,button\_profile;

JTextArea textArea;

Box box1,box2,box3;

// Объявляем классы

iHandler ihandler = new iHandler();

Mouse Ms = new Mouse();

LisT Ls = new LisT();

public GUI(){

super("Библиотека");

//Размер экрана

setSize(1100,600);

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

// Инцилизация массивов

elements = new String[] {"Ключевому слову","Автору","Книге","Году издания","В наличие"};

sorting = new String[] {"Автору","Книге","Году издания","Наличию"};

headers = new Object[] {"Автор","Книга","Год издания","Наличие"};

data = new Object[50][4] ;

//Инцилизация контейнеров

search\_label = new JLabel("Поиск по:");

сomboBox = new JComboBox(elements);

toolBar = new JToolBar("Инструментальная панель");

search\_field = new JTextField("Введите данные");

button\_search = new JButton("Поиск");

tableModel = new DefaultTableModel(data, headers);

table =new JTable(tableModel);

sortingLabel = new JLabel("Сортировка по");

sortingCombolBox = new JComboBox(sorting);

textArea = new JTextArea(20,30);

textArea.setLineWrap(true);// Перенос текста

textArea.setWrapStyleWord(true);// Перенос слова целеком на соседнию строку

//textArea.append("Область для ввода текстового содержимого ");

menuBar = new JMenuBar();

fileMenu = new JMenu("Файл");

openMenuIt = new JMenuItem("Открыть");

saveMenuIt = new JMenuItem("Сохранить");

newMenuIt = new JMenuItem("Новое");

extMenuIt = new JMenuItem("Выйти");

// Создаём кнопки и задаём им значки

button\_save = new JButton(new ImageIcon("./16x16/29.png"));

button\_open = new JButton(new ImageIcon("./16x16/46.png"));

button\_add = new JButton(new ImageIcon("./16x16/20.png"));

button\_edit = new JButton(new ImageIcon("./16x16/36.png"));

button\_delete = new JButton(new ImageIcon("./16x16/17.png"));

button\_profile = new JButton(new ImageIcon("./16x16/11.png"));

// Настройка подсказок для кнопок

button\_save.setToolTipText("Сохранить список книг");

button\_open.setToolTipText("Открыть список книг");

button\_add.setToolTipText("Добавить информацию");

button\_edit.setToolTipText("Редактировать");

button\_delete.setToolTipText("Удалить информацию о книге");

button\_profile.setToolTipText("Профиль администратора");

// Изменяем размер кнопок

button\_save.setPreferredSize(new Dimension(25,25));

button\_open.setPreferredSize(new Dimension(25,25));

button\_add.setPreferredSize(new Dimension(25,25));

button\_edit.setPreferredSize(new Dimension(25,25));

button\_delete.setPreferredSize(new Dimension(25,25));

button\_profile.setPreferredSize(new Dimension(25,25));

;

//Создаём меню

Menu();

//Создаём меня поиска сортировки

Bar();

// Все созданные и расположенные элементы выводим на окно

getContentPane().add(toolBar, BorderLayout.NORTH);

getContentPane().add(new JScrollPane(textArea), BorderLayout.EAST);

getContentPane().add(new JScrollPane(table));

сomboBox.addActionListener(ihandler);

//search\_field.addActionListener(ihandler);

extMenuIt.addActionListener(ihandler);

button\_open.addActionListener(ihandler);

button\_save.addActionListener(ihandler);

search\_field.addMouseListener(Ms);

ListSelectionModel selModel = table.getSelectionModel();

selModel.addListSelectionListener(Ls);

}

private void Menu(){

// Создаём меню

fileMenu.add(newMenuIt);

fileMenu.add(openMenuIt);

fileMenu.add(saveMenuIt);

fileMenu.addSeparator();

fileMenu.add(extMenuIt);

//Загружаем меню

menuBar.add(fileMenu);

menuBar.add(Box.createHorizontalGlue());

setJMenuBar(menuBar);

}

private void Bar(){

// Вставляем кнопки в панель инструментов

box1 = Box.createHorizontalBox();

box1.add(button\_save);

box1.add(Box.createHorizontalStrut(3));

box1.add(button\_open);

box1.add(Box.createHorizontalStrut(3));

box1.add(button\_profile);

box1.add(Box.createHorizontalStrut(3));

box1.add(new JSeparator(SwingConstants.VERTICAL));

box1.add(Box.createHorizontalStrut(3));

box1.add(button\_add);

box1.add(Box.createHorizontalStrut(3));

box1.add(button\_delete);

box1.add(Box.createHorizontalStrut(3));

box1.add(button\_edit);

// Создаём поисковую строку

box2 = Box.createHorizontalBox();

box2.add(search\_label);

box2.add(Box.createHorizontalStrut(6));

box2.add(сomboBox);

box2.add(Box.createHorizontalStrut(6));

box2.add(search\_field);

box2.add(Box.createHorizontalStrut(6));

box2.add(button\_search);

box2.add(Box.createHorizontalStrut(6));

box2.add(button\_search);

// Создаём строку для сортировки

box3 = Box.createHorizontalBox();

box3.add(sortingLabel);

box3.add(Box.createHorizontalStrut(6));

box3.add(sortingCombolBox);

// Объединяем все три строки

toolBar.add(box1, BorderLayout.WEST);

toolBar.add(Box.createHorizontalStrut(12));

toolBar.add(new JSeparator(SwingConstants.VERTICAL));

toolBar.add(Box.createHorizontalStrut(12));

toolBar.add(box2);

toolBar.add(Box.createHorizontalStrut(12));

toolBar.add(new JSeparator(SwingConstants.VERTICAL));

toolBar.add(Box.createHorizontalStrut(12));

toolBar.add(box3, BorderLayout.EAST);

//Лишаем возможность перемещать панель

toolBar.setFloatable(false);

}

public class iHandler implements ActionListener{

public void actionPerformed(ActionEvent e) {

if(e.getSource() == сomboBox){

//search\_field.setText("По " + (String) сomboBox.getSelectedItem());

search\_field.setText("Введите данные");

}

if(e.getSource() == search\_field){

search\_field.setText("");

}

if(e.getSource() == extMenuIt){

dispose();

}

if(e.getSource() == button\_open){

FileOpen file = new FileOpen("Открытие данных",tableModel);

}

if(e.getSource() == button\_save){

FileSave file = new FileSave("Сохранение данных",tableModel);

}

}

}

public class Mouse implements MouseListener{

public void mouseClicked(MouseEvent e) {

// Если строка в строке search\_field, есть первичная запись из elements, то она удаляется

if((search\_field.getText().contains("Введите данные")))

/\*

if((search\_field.getText().contains(elements[0])) || (search\_field.getText().contains(elements[1]))

||(search\_field.getText().contains(elements[2])) || (search\_field.getText().contains(elements[3]))

||(search\_field.getText().contains(elements[4])))\*/

search\_field.setText("");

}

public void mouseEntered(MouseEvent e) {}

public void mouseExited(MouseEvent e) {}

public void mousePressed(MouseEvent e) {}

public void mouseReleased(MouseEvent e) {}

}

public class LisT implements ListSelectionListener{

public void valueChanged(ListSelectionEvent e) {

textArea.setText("");

FileOpenText text = new FileOpenText(textArea,"2.txt");

}

}

}

///////////////////////// Password ////////////////////////////////

import javax.swing.\*;

import javax.swing.border.EmptyBorder;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class Password extends JFrame {

Handler handler = new Handler();

Box box1,box2,box3,mainBox;

JButton button\_ok,button\_cancel,button\_registration;

JTextField FieldLogin;

JLabel LableLogin,passwLabel;

JPasswordField passwField;

public Password(){

super("Регистрация читателя");

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

setResizable(false);

// Настраиваем первую горизонтальную панель (для ввода идентификатора или ФИО)

box1 = Box.createHorizontalBox();

LableLogin = new JLabel("Введите логин");

FieldLogin = new JTextField(20);

box1.add(LableLogin);

box1.add(Box.createHorizontalStrut(14));

box1.add(FieldLogin);

box2 = Box.createHorizontalBox();

passwLabel = new JLabel ("Введите пароль");

passwField = new JPasswordField (20);

box2.add(Box.createHorizontalGlue());

box2.add(passwLabel);

box2.add(Box.createHorizontalStrut(6));

box2.add(passwField);

// Настраиваем вторую горизонтальную панель (с кнопками)

box3 = Box.createHorizontalBox();

button\_ok = new JButton("Начать работу");

button\_cancel = new JButton("Отмена");

button\_registration = new JButton("Регестрация");

box3.add(Box.createHorizontalGlue());

box3.add(button\_registration);

box3.add(Box.createHorizontalStrut(12));

box3.add(button\_ok);

box3.add(Box.createHorizontalStrut(12));

box3.add(button\_cancel);

// Размещаем три горизонтальные панели на одной вертикальной

mainBox = Box.createVerticalBox();

mainBox.setBorder(new EmptyBorder(12,12,12,12));

mainBox.add(box1);

mainBox.add(Box.createVerticalStrut(12));

mainBox.add(box2);

mainBox.add(Box.createVerticalStrut(17));

mainBox.add(box3);

setContentPane(mainBox);

pack();

passwField.addActionListener(handler);

button\_ok.addActionListener(handler);

button\_cancel.addActionListener(handler);

button\_registration.addActionListener(handler);

}

public class Handler implements ActionListener{//implements - взаймсвенный метод

public void actionPerformed(ActionEvent e) {

// Метод действия(нажатия кнопки) преевод : действи выполнено

if((e.getSource() == button\_ok) || (e.getSource() == passwField)){

try{

//Проверка ввода ФИО

String strLogin;

strLogin = FieldLogin.getText();

if(strLogin.length() == 0) throw new NullPointerException();

//Проверка ввода пароля

String strPassw;

strPassw = passwField.getText();

if(strPassw.length() == 0) throw new myException();

//Проверка на правильность пароля

String str;

str=passwField.getText();

if(!str.equals("2340612340")) throw new myException();

//JOptionPane.showMessageDialog(null,"Вход разрешен!");

//Запись в файл и запускаем другое окно

GUI r = new GUI();

r.setVisible(true);

r.setLocationRelativeTo(null);

dispose();

} catch (NullPointerException ex) {

JOptionPane.showMessageDialog(null,"<html><p> Введите логин </p> ",

"Ошибка", JOptionPane.WARNING\_MESSAGE);

} catch (myException meEx) {

JOptionPane.showMessageDialog(null,meEx.getMessage());

};

}

if(e.getSource() == button\_cancel){

dispose();

}

if(e.getSource() == button\_registration){

Registration list = new Registration("Регестрация");

list.setVisible(true);

list.setLocationRelativeTo(null);

}

}

}

private class myException extends Exception{

public myException(){

super("Пароль не верен!");

passwField.setText("");

}

}

}

/////////////////////////////////// FileSave ////////////////////////////////

import java.awt.FileDialog;

import java.io.BufferedWriter;

import java.io.FileWriter;

import java.io.IOException;

import javax.swing.JFrame;

import javax.swing.table.DefaultTableModel;

public class FileSave extends JFrame {

public FileSave(String str,DefaultTableModel table){

FileDialog sava = new FileDialog(this,str,FileDialog.SAVE);

sava.setFile("\*.txt");// Установка начального каталога

sava.setVisible(true);

//Определяем имя каталога или файла

String fileNameSave = sava.getDirectory() + sava.getFile();

if(fileNameSave == null) return; // Пользователь нажал отмена

try{

BufferedWriter writer = new BufferedWriter(new FileWriter(fileNameSave));

for(int i = 0; i < table.getRowCount(); i++){

writer.write("\r\n");

for(int j = 0; j < table.getColumnCount(); j++){

writer.write((String)table.getValueAt(i,j));

if(j<4) writer.write("|");

}

}

writer.close();

}catch (IOException e) {

e.printStackTrace();

}

}

}

/////////////////////////////////// FileOpen ////////////////////////////////

import java.awt.FileDialog;

import java.io.BufferedReader;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import javax.swing.JFrame;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

public class FileOpen extends JFrame{

public FileOpen(String str,DefaultTableModel table){

FileDialog open = new FileDialog(this,str,FileDialog.LOAD);

open.setFile("\*.txt");// Установка начального каталога

open.setVisible(true);

//Определяем имя каталога или файла

String fileNameOpen = open.getDirectory() + open.getFile();

if(fileNameOpen == null) return; // Пользователь нажал отмена

try{

// Считываем строку и заносим в буфер

BufferedReader reader = new BufferedReader(new FileReader(fileNameOpen));

int rows = table.getRowCount();

for(int i = 0; i < rows; i++)

table.removeRow(0);

String author;

do{

author = reader.readLine();

if(author != null){

String[] str1 = author.split("\\|");

if(str1.length == 4)

table.addRow(new String[] {str1[0],str1[1],str1[2],str1[3]} );

}

}while(author != null);

reader.close();

}catch(FileNotFoundException e) {

e.printStackTrace();

}catch (IOException e) {

e.printStackTrace();

}

}

}

//////////////////////////////////////// main ///////////////////////////////////

**import** javax.swing.\*;

**public** **class** main {

**public** **static** **void** main(String arts[]){

GUI r = **new** GUI();

r.setVisible(**true**);

r.setLocationRelativeTo(**null**);

}

}