

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 № 8
«Организация многопоточных
приложений»

Выполнил: Мохно Д. А.

Факультет КТИ

Группа № 3312

Преподаватель: Павловский М.Г.

Подпись преподавателя _____

Санкт-Петербург

2024 г

Цель работы

знакомство с правилами и классами построения параллельных приложений в языке Java.

Распечатки XML-файлов до загрузки данных в экранную форму и после их выгрузки.

data.xml до внесения изменений:

```
<?xml version="1.0" encoding="UTF-8" ?>
<base>
  <record client="Годунов Е. А." doctor="Смирнов В. П." date="24.09.2024" time="10:00"
symptoms="Боль в горле"/>
  <record client="Иванова А. В." doctor="Кузнецов М. И." date="25.09.2024"
time="11:00" symptoms="Температура"/>
  <record client="Сидоров К. А." doctor="Попов А. Н." date="26.09.2024" time="09:30"
symptoms="Кашель"/>
  <record client="Петрова Л. М." doctor="Соколов Б. Т." date="27.09.2024" time="14:00"
symptoms="Головная боль"/>
  <record client="Семенов И. О." doctor="Лебедев В. У." date="28.09.2024" time="15:15"
symptoms="Боль в спине"/>
  <record client="Николаев Д. Р." doctor="Козлов Г. А." date="29.09.2024" time="08:45"
symptoms="Аллергия"/>
  <record client="Федорова Е. С." doctor="Новиков Д. В." date="30.09.2024"
time="13:00" symptoms="Боль в животе"/>
  <record client="Алексеева М. Б." doctor="Морозов С. Е." date="01.10.2024"
time="10:30" symptoms="Усталость"/>
  <record client="Романов А. И." doctor="Петров Н. Ч." date="02.10.2024" time="12:00"
symptoms="Бронхит"/>
  <record client="Соловьев Д. П." doctor="Волков Р. Ш." date="03.10.2024" time="16:00"
symptoms="Мигрень"/>
</base>
```

data.xml после внесения изменений:

```
<?xml version="1.0" encoding="UTF-8" ?>
<base>
  <record client="Годунов Е. А." doctor="Смирнов В. П." date="24.09.2024" time="10:00"
symptoms="Боль в горле"/>
  <record client="Иванова А. В." doctor="Кузнецов М. И." date="25.09.2024"
time="11:00" symptoms="Температура"/>
  <record client="Сидоров К. А." doctor="Попов А. Н." date="26.09.2024" time="09:30"
symptoms="Кашель"/>
  <record client="Петрова Л. М." doctor="Соколов Б. Т." date="27.09.2024" time="14:00"
symptoms="Головная боль"/>
  <record client="Семенов И. О." doctor="Лебедев В. У." date="28.09.2024"
time="15:15" symptoms="Боль в спине"/>
```

</base>

Скриншот построенного отчета.

Поликлиника				Список
Список.				
ФИО ПАЦИЕНТА	ФИО ВРАЧА	ДАТА ЗАПИСИ	ВРЕМЯ ЗАПИСИ	ЖАЛОБЫ
Годунов Е. А.	Смирнов В. П.	24.09.2024	10:00	Боль в горле
Иванова А. В.	Кузнецов М. И.	25.09.2024	11:00	Температура
Сидоров К. А.	Попов А. Н.	26.09.2024	09:30	Кашель
Петрова Л. М.	Соколов Б. Т.	27.09.2024	14:00	Головная боль
Семенов И. О.	Лебедев В. У.	28.09.2024	15:15	Боль в спине

Текст документации, сгенерированный Javadoc.

PACKAGECLASSUSE TREE INDEX HELP

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHODSEARCH

Class Processor

java.lang.Object[Ⓢ]
Processor

public class Processor
extends Object[Ⓢ]

Класс Processor выполняет операции по работе с файлами и таблицами для сохранения, открытия, редактирования и отображения данных в формате CSV.

Nested Class Summary

Nested Classes

Modifier and Type	Class	Description
(package private) static class	Processor.Parser	Класс Parser предоставляет методы для работы с XML-файлами.

Field Summary

Fields

Modifier and Type	Field	Description
String [Ⓢ]	currentFilePath	Текущий путь к файлу
JTable [Ⓢ]	table	Таблица для отображения данных
DefaultTableModel [Ⓢ]	tableModel	Модель таблицы для управления данными
JFrame [Ⓢ]	window	Основное окно приложения

Constructor Summary

Constructors

Constructor	Description
Processor(JFrame [Ⓢ] frame)	Конструктор, принимающий основное окно приложения.

Method Summary

All MethodsInstance MethodsConcrete Methods

Modifier and Type	Method	Description
void	createTable(String [Ⓢ] [] colNames)	Создает таблицу с заданными именами столбцов.
void	exportFile(String [Ⓢ] format)	Экспортирует файлы.
private net.sf.jasperreports.engine.JasperPrint	getReport(Document [Ⓢ] doc)	Конвертирует таблицу в объект для печати или экспорта
void	openFile()	Открывает диалоговое окно для выбора CSV-файла и загружает данные в таблицу.
void	printFile()	Выводит файл на печать.
String [Ⓢ] [][]	readFile(String [Ⓢ] pathToFile)	Читает содержимое CSV-файла и возвращает его как двумерный массив строк.
void	saveFile()	Сохраняет данные таблицы в текущий CSV/XML-файл.
void	saveFileAs()	Открывает диалоговое окно для выбора нового места и имени для сохранения CSV-файла.
private void	tableFill(String [Ⓢ] [][] text)	Заполняет таблицу данными из двумерного массива строк.
private void	tableFill(Document [Ⓢ] data)	Заполняет таблицу данными из двумерного массива строк.
void	writeFile(String [Ⓢ] pathToFile)	Записывает содержимое таблицы в CSV-файл.
private void	writeFile(Document [Ⓢ] data)	Записывает содержимое таблицы в XML-файл.

Methods inherited from class java.lang.Object[Ⓢ]
clone[Ⓢ], equals[Ⓢ], finalize[Ⓢ], getClass[Ⓢ], hashCode[Ⓢ], notify[Ⓢ], notifyAll[Ⓢ], toString[Ⓢ], wait[Ⓢ], wait[Ⓢ], wait[Ⓢ]

Field Details

window

public JFrame[Ⓢ] window
Основное окно приложения

table


public JTable[Ⓢ] table
Таблица для отображения данных

tableModel

public DefaultTableModel[Ⓢ] tableModel
Модель таблицы для управления данными

currentFilePath

public String[Ⓢ] currentFilePath
Текущий путь к файлу

Constructor Details
<div><div>Processor</div><div><div>Processor(JFrame frame)</div><div>Конструктор, принимающий основное окно приложения.</div><div>Parameters: frame - основное окно приложения JFrame</div></div></div>
Method Details
<div><div>openFile</div><div><div>public void openFile()</div><div>Открывает диалоговое окно для выбора CSV-файла и загружает данные в таблицу.</div></div></div>
<div><div>saveFile </div><div><div>public void saveFile()</div><div>Сохраняет данные таблицы в текущий CSV/XML-файл.</div></div></div>
<div><div>saveFileAs</div><div><div>public void saveFileAs()</div><div>Открывает диалоговое окно для выбора нового места и имени для сохранения CSV-файла.</div></div></div>
<div><div>printFile</div><div><div>public void printFile()</div><div>Выводит сообщение о печати файла. Этот метод можно расширить для добавления функциональности печати.</div></div></div>
<div><div>exportFile</div><div><div>public void exportFile(String format)</div><div>Экспортирует файлы.</div><div>Parameters: format - Формат файла "html" или "pdf"</div></div></div>
<div><div>readFile</div><div><div>public String[][] readFile(String pathToFile)</div><div>Читает содержимое CSV-файла и возвращает его как двумерный массив строк.</div><div>Parameters: pathToFile - путь к файлу для чтения</div><div>Returns: двумерный массив строк, представляющий данные таблицы, или null в случае ошибки чтения файла</div></div></div>
<div><div>writeFile</div><div><div>public void writeFile(String pathToFile)</div><div>Записывает содержимое таблицы в CSV-файл.</div><div>Parameters: pathToFile - путь к файлу для записи</div></div></div>
<div><div>writeFile</div><div><div>private void writeFile(Document data)</div><div>Записывает содержимое таблицы в XML-файл.</div><div>Parameters: data - объект xml документа</div></div></div>
<div><div>tableFill</div><div><div>private void tableFill(String[][] text)</div><div>Заполняет таблицу данными из двумерного массива строк.</div><div>Parameters: text - двумерный массив строк, представляющий данные для таблицы</div></div></div>
<div><div>tableFill</div><div><div>private void tableFill(Document data)</div><div>Заполняет таблицу данными из двумерного массива строк.</div><div>Parameters: data - объект xml документа</div></div></div>
<div><div>createTable</div><div><div>public void createTable(String[] colNames)</div><div>Создает таблицу с заданными именами столбцов.</div><div>Parameters: colNames - массив строк с именами столбцов</div></div></div>

Фрагменты кода, отвечающие за сохранение и чтение данных из XML-файла.

```
/**
 * Экспортирует файлы.
 *
 * @param format Формат файла "html" или "pdf"
 */
public void exportFile(String format) {
    // Этап 1: создание документа XML в отдельном потоке
    Thread loadDataThread = new Thread(() -> {
        try {
            Document doc = Parser.to_xml(tableModel);
            // Этап 2: подготовка отчёта в другом потоке
            Thread generateReportThread = new Thread(() -> {
                JasperPrint print = getReport(doc);
                if (print != null) {
                    // Этап 3: экспорт отчёта в файл в третьем потоке
                    Thread exportFileThread = new Thread(() -> {
                        try {
                            FileDialog fd = new FileDialog(window, "",
                                FileDialog.SAVE);

                            fd.setDirectory(System.getProperty("user.dir"));
                            fd.setFile("Untitled." + format);
                            fd.setVisible(true);
                            String output_file_path = fd.getDirectory() +
                                fd.getFile();

                            if (output_file_path.endsWith("pdf")) {
                                File output_file = new
                                    File(output_file_path);

                                JasperExportManager.exportReportToPdfFile(print,
                                    output_file.getAbsolutePath());
                            } else if (output_file_path.endsWith("html"))
                            {
                                File output_file = new
                                    File(output_file_path);

                                JasperExportManager.exportReportToHtmlFile(print,
                                    output_file.getAbsolutePath());
                            }
                        } catch (JRException e) {
                            e.printStackTrace();
                        }
                    });
                    exportFileThread.start(); // Запуск потока экспорта
                } else JOptionPane.showMessageDialog(window, "Ошибка при
формировании отчёта", "ERROR", JOptionPane.INFORMATION_MESSAGE);
            });
            generateReportThread.start(); // Запуск потока генерации
отчёта
        } catch (Exception e) {
            e.printStackTrace();
        }
    });
    loadDataThread.start(); // Запуск потока загрузки данных
}
```

Вывод

В результате выполнения работы были изучены правила работы с многопоточностью в java и получены практические навыки в программировании на этом языке.

Ссылки

https://github.com/DanyaMokhno/OOP_Labs/tree/main/com.study_oop.Laba_8