Минобрнауки России

Федеральное государственное автономное образовательное учреждение

высшего образования «Санкт-Петербургский государственный

электротехнический университет «ЛЭТИ»

им В. И. Ульянова (Ленина)»

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

**Лабораторная работа № 7**

**«Разработка интерфейса пользователя» по дисциплине**

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

Выполнил Авдеев Н. А.

Факультет КТИ

Группа № 3312

Санкт-Петербург

2024

**Цель работы:**

Знакомство со способами формирования отчетов с использованием конструктора Jaspersoft iReport Designert.

**Распечатки XML-файлов:  
<?xml version="1.0" encoding="UTF-8"?>**

**<coffees>**

**<coffee>**

**<type>Арабика</type>**

**<country>Бразилия</country>**

**<roast>Средняя</roast>**

**<grade>85</grade>**

**</coffee>**

**<coffee>**

**<type>Робуста</type>**

**<country>Вьетнам</country>**

**<roast>Темная</roast>**

**<grade>70</grade>**

**</coffee>**

**<coffee>**

**<type>Либерика</type>**

**<country>Либерия</country>**

**<roast>Светлая</roast>**

**<grade>80</grade>**

**</coffee>**

**</coffees>**

**Скриншоты, иллюстрирующие работу JasperReport:**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

**Исходные тексты генерирования отчёта:**

public static void printReport(String xmlFilePath, String templatePath, String outputPath, String format) {

            try {

                // Создаем источник данных

                JRXmlDataSource dataSource = new JRXmlDataSource(xmlFilePath, "/coffees/coffee"); // Пример XPath

                // Компилируем шаблон

                JasperReport jasperReport = JasperCompileManager.compileReport(templatePath);

                // Заполняем отчет данными

                JasperPrint print = JasperFillManager.fillReport(jasperReport, new HashMap<>(), dataSource);

                // Выбираем экспортер в зависимости от формата

                JRExporter exporter = null;

                switch (format.toLowerCase()) {

                    case "pdf":

                        exporter = new JRPdfExporter();

                        break;

                    case "html":

                        exporter = new JRHtmlExporter();

                        break;

                    case "excel":

                        exporter = new JRXlsxExporter();

                        break;

                    default:

                        throw new IllegalArgumentException("Unsupported format: " + format);

                }

                // Устанавливаем параметры экспорта

                exporter.setParameter(JRExporterParameter.OUTPUT\_FILE\_NAME, outputPath);

                exporter.setParameter(JRExporterParameter.JASPER\_PRINT, print);

                // Экспортируем отчет

                exporter.exportReport();

            } catch (JRException e) {

                JOptionPane.showMessageDialog(null, "Ошибка при генерации отчета: " + e.getMessage());

            }

        }

**Вывод:**

Реализовано приложение для генерации отчетов в различных форматах (PDF, HTML, Excel) на основе данных из XML-файлов. Приложение использует мощный инструмент Jaspersoft Studio для создания гибких и визуально привлекательных шаблонов отчетов. Благодаря механизмам обработки исключений, программа обеспечивает стабильную работу и информирует пользователя о возникших ошибках.

**Ссылка на репозиторий**:

https://github.com/G3rb3r228/Labs\_OOP\_AvdeevNA-3312/tree/main/Lab07\_Avdeev3312