­­­МИНОБРНАУКИ РОССИИ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ЛЭТИ» им.В.И.УЛЬЯНОВА (ЛЕНИНА)

Кафедра вычислительной техники

Отчет по лабораторной работе №7

по дисциплине «Объектно-ориентированное программирование»

Тема: «**Построение отчетов в PDF- и HTML- форматах**»

|  |  |
| --- | --- |
| Студент гр. 9307 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Брызгалова Е.А. |
|  |  |
| Преподаватель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_Павловский М.Г. |

Санкт-Петербург

2021

Оглавление

[Цель работы 3](#_Toc72768353)

[Описание задания 3](#_Toc72768354)

[Проверка работоспособности программы 3](#_Toc72768355)

[Класс отвечающий за создания отчётов 5](#_Toc72768356)

[Выводы 6](#_Toc72768357)

# Цель работы

Знакомство со способами формирования отчетов с использованием библиотеки IText.

# Описание задания

* Скриншоты созданных отчётов.
* Текст документации, сгенерированный Javadoc.
* Фрагменты кода, отвечающие за создание отчётов в формате PDF и HTML.

# Проверка работоспособности программы

При запуске программы, пользователь загружает данные из файла (для примера подготовлен файл 1.xml) или вводит собственные

Для создания отчёта пользователю необходимо нажать кнопку «Отчёт» и в высветившемся окне выбрать формат отчёта (рис1.).

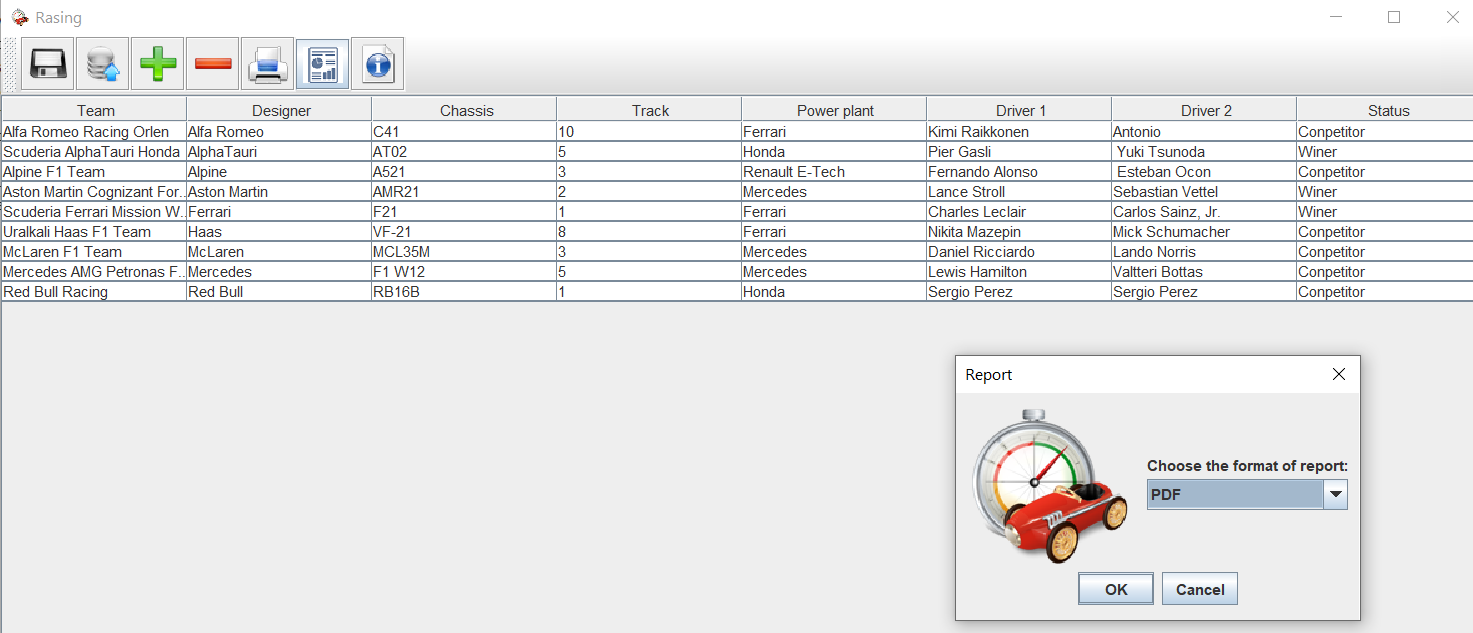


Рис.1

После нажатия кнопки «ОК» генерируется отчёт в выбранном формате. На рис.2 и рис.4 представлены скриншоты отчётов в формате pdf и html.

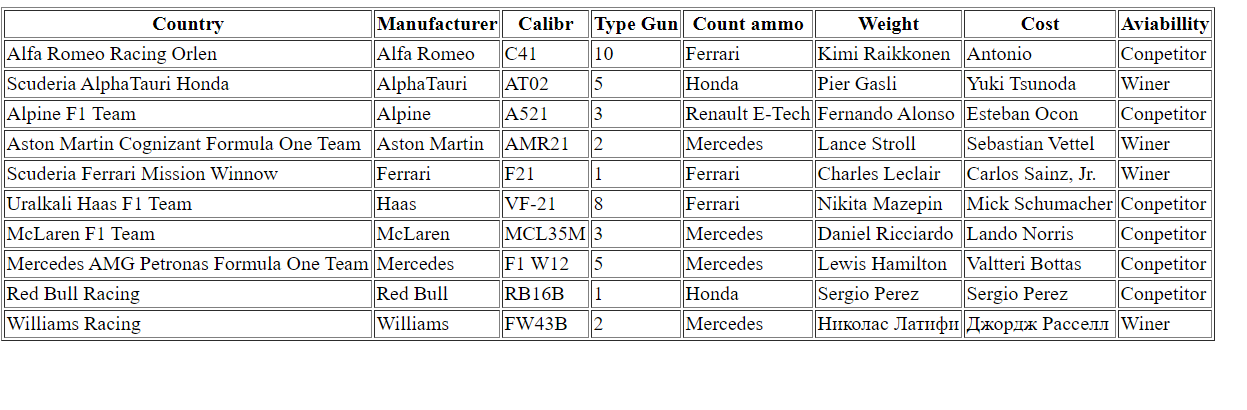


Рис.3 Отчёт в формате html

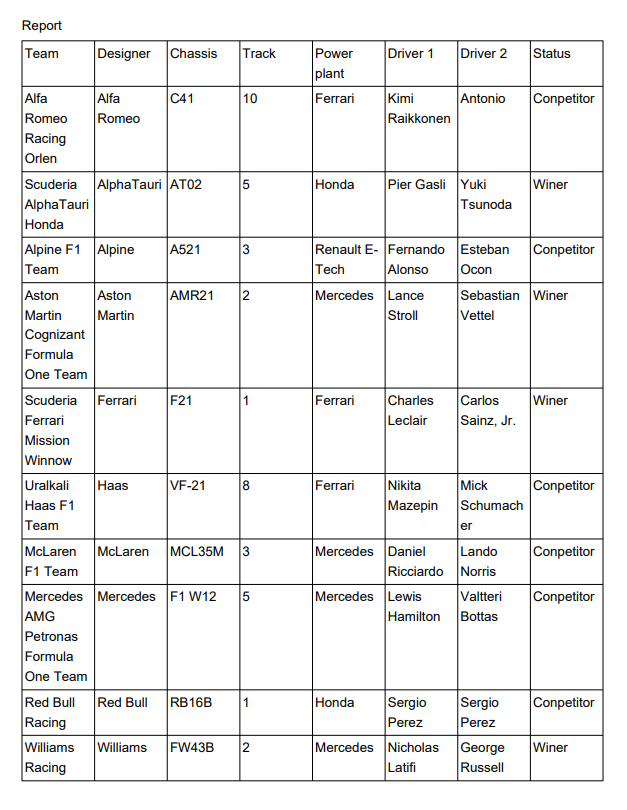


Рис.4 Отчёт в формате pdf

# Класс отвечающий за создания отчётов

import com.itextpdf.kernel.pdf.PdfDocument;

import com.itextpdf.kernel.pdf.PdfWriter;

import com.itextpdf.layout.Document;

import com.itextpdf.layout.element.Cell;

import com.itextpdf.layout.element.Paragraph;

import com.itextpdf.layout.element.Table;

import javax.swing.table.DefaultTableModel;

import java.io.FileNotFoundException;

import java.io.FileWriter;

import java.io.IOException;

import java.io.PrintWriter;

public class Report {

void createPDF(DefaultTableModel model) throws FileNotFoundException {

PdfDocument pdfDoc = new PdfDocument(new PdfWriter("./Report.pdf"));

pdfDoc.addNewPage();

Document doc = new Document(pdfDoc);

Paragraph paragraph = new Paragraph("Report");

doc.add(paragraph);

String column[] = {"Team", "Designer", "Chassis","Track","Power plant","Driver 1", "Driver 2","Status"};

Table table = new Table(column.length);

for(String s: column) {

table.addCell(new Cell().add(s));

}

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

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

table.addCell(new Cell().add(model.getValueAt(i,j).toString()));

}

}

doc.add(table);

doc.close();

}

void createHTML(DefaultTableModel model) {

PrintWriter pw = null;

try {

pw = new PrintWriter(new FileWriter("./Report.html"));

} catch (IOException e) {

e.printStackTrace();

}

pw.println("<TABLE BORDER><TR><TH>Country<TH>Manufacturer<TH>Calibr<TH>Type Gun<TH>Count ammo<TH>Weight<TH>Cost<TH>Aviabillity</TR>");

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

int square = i \* i;

pw.println("<TR><TD>" + (String) model.getValueAt(i,0)

+ "<TD>" + (String) model.getValueAt(i,1)

+ "<TD>" + (String) model.getValueAt(i,2)

+ "<TD>" + (String) model.getValueAt(i,3)

+ "<TD>" + (String) model.getValueAt(i,4)

+ "<TD>" + (String) model.getValueAt(i,5)

+ "<TD>" + (String) model.getValueAt(i,6)

+ "<TD>" + (String) model.getValueAt(i,7));

}

pw.println("</TABLE>");

pw.close();

}

}

# Выводы

В ходе лабораторной работы были получены навыки создания документов в формате pdf и html. Была создана программа с возможностью сохранения отчётов в формате pdf, с помощью библиотеки IText, и в формате html.