Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ **ИНФОРМАТИКА, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**
МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/07 Интеллектуальные системы анализа, обработки и интерпретации больших данных**

О Т Ч Е Т

по лабораторной работе №10

Название: Лабораторная работа №10

Дисциплина: Языки программирования для работы с большими данными

Студент	ИУ6-22М		М.И. Замула
	(Группа)	(Подпись, дата)	(И.О. Фамилия)
Преподаватель			П.В. Степанов
		(Подпись, дата)	(И.О. Фамилия)

Москва, 2023 г.

Вариант: 2

Датасеты

1) https://www.kaggle.com/vitaliymalcev/russian-passenger-air-service-20072020
2) https://www.kaggle.com/dwdkills/russian-demography
3) https://www.kaggle.com/trolukovich/russian-schools-geodata

Примеры

1) Первый – подсчет количества слов в текстовом файле
2) Второй – статистика из первой ссылки (статистика по пассажирским авиаперевозкам в России за 2007-2020 гг)

https://github.com/pavlentytest/FileReadDemo

Задание

1) Выбрать любой датасет на kaggle.com
2) Сделать 10 выборок данных по выбранной предметной области

Выполнение

```
package lab10;

package org.example;

import org.apache.spark.sql.SparkSession;
import org.apache.spark.sql.Dataset;
import org.apache.spark.sql.Row;

public class Lab10 {
    public static void main(String[] args) {
        // Подгружаем файл
        SparkSession spark = SparkSession
                .builder()
                .appName("Java Spark SQL basic example")
                .config("spark.master", "local")
                .getOrCreate();
        // Создаем датасет из csv файла
        Dataset<Row> df =
spark.read().option("header","true").csv("src/lab10/sample.csv");
        // Создаем временную таблицу twitter_support
        df.createOrReplaceTempView("twitter_support");
        // Заголовки в csv файле:
        //
tweet_id,author_id,inbound,created_at,text,response_tweet_id,in_response_to_t
weet_id
        // Используем SQL и .show() для 10 примеров выборки и аггрегации
        // Вывести все твиты, отправленные пользователем с ID 119280:
        spark.sql("SELECT * FROM twitter_support WHERE author_id =
119280").show();
        // Вывести все твиты пользователей, "входящих" в компанию,
оказывающей поддержку клиентов в Twitter (inbound=TRUE):
        spark.sql("SELECT * FROM twitter_support WHERE inbound =
True").show();
        // Вывести количество твитов, отправленных каждым пользователем
```

```
        spark.sql("SELECT author_id, COUNT(*) AS tweet_count FROM
twitter_support GROUP BY author_id").show();
        // Вывести все твиты, на которые не был дан ответ (response_tweet_id
- NULL):
        spark.sql("SELECT * FROM twitter_support WHERE response_tweet_id IS
NULL").show();
        // Получить список всех твитов, содержащих определенное слово в
тексте:
        spark.sql("SELECT * FROM twitter_support WHERE text LIKE
'hi'").show();
        // Найти все твиты, отправленные за последние 10 дней:
        spark.sql("SELECT * FROM twitter_support WHERE created_at >=
DATE_SUB(NOW(), INTERVAL 10 DAY);").show();
        // Найти все твиты, содержащие менее 30 символов:
        spark.sql("SELECT * FROM twitter_support WHERE CHAR_LENGTH(text) <
30;").show();
        // Вывести все твиты, в которых менее 10 символов, которые отправлены
до полудня любого дня
        spark.sql("SELECT * FROM twitter_support WHERE LENGTH(text) < 10 AND
TO_CHAR(TO_DATE(created_at, 'DY MON DD HH24:MI:SS TZD YYYY'), 'HH24') <
'12'").show();
        // Получить среднее количество символов в твитах (text):
        spark.sql("SELECT AVG(LENGTH(text)) FROM twitter_support").show();
        //Найти твиты, которые являются ответом на другой твит:
        spark.sql("SELECT * FROM twitter_support WHERE
in_response_to_tweet_id IS NOT NULL").show();


    }
}
```