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 И СИСТЕМЫ УПРАВЛЕНИЯ**

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

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/12 Интеллектуальный анализ больших данных в системах поддержки принятия решений.**

О Т Ч Е Т

по лабораторной работе № 10

Вариант № 5

Название: Spark

Дисциплина: Языки программирования для работы с большими данными

Студент ИУ6-23М (Группа) _____ (Подпись, дата) А.О.Крейденко (И.О. Фамилия)

Преподаватель _____ (Подпись, дата) П.В. Степанов (И.О. Фамилия)

Москва, 2024

Цель: ознакомиться с работой Spark в языке программирования Java.

Задание: сделать 10 выборок данных по выбранной предметной области

Был выбран датасет с данными об оценках приложений в Google Store.

Код класса Main:

```
package org.example;

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

public class Main {
    public static void main(String[] args) {
        // Инициализация Spark сессии
        SparkSession spark = SparkSession.builder()
                .appName("GooglePlayStoreAnalysis")
                .master("local")
                .getOrCreate();

        // Чтение данных из CSV файла
        Dataset<Row> df = spark.read()
                .option("header", "true")
                .option("inferSchema", "true")
                .csv("googleplaystore.csv");

        // Удаление строк с NaN значениями в любых столбцах
        df = df.na().drop();

        // Создание представления для SQL-запросов
        df.createOrReplaceTempView("googleplaystore");

        // 1. Показать первые 5 строк
        spark.sql("SELECT * FROM googleplaystore LIMIT 5").show();
```

```
// 2. Подсчитать количество приложений для каждой категории

spark.sql("SELECT Category, COUNT(*) as count FROM googleplaystore GROUP BY Category ORDER BY count DESC").show();


// 3. Найти топ-10 самых популярных приложений по количеству установок

spark.sql("SELECT * FROM (SELECT *, CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT) AS InstallsInt FROM googleplaystore) ORDER BY InstallsInt DESC LIMIT 10").show();


// 4. Найти топ-10 бесплатных приложений с наибольшим количеством установок

spark.sql("SELECT * FROM (SELECT *, CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT) AS InstallsInt FROM googleplaystore WHERE Type = 'Free') ORDER BY InstallsInt DESC LIMIT 10").show();


// 5. Найти топ-10 платных приложений с наибольшим количеством установок

spark.sql("SELECT * FROM (SELECT *, CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT) AS InstallsInt FROM googleplaystore WHERE Type = 'Paid') ORDER BY InstallsInt DESC LIMIT 10").show();


// 6. Подсчитать количество приложений для каждого уровня контента (Content Rating)

spark.sql("SELECT `Content Rating`, COUNT(*) as count FROM googleplaystore GROUP BY `Content Rating` ORDER BY count DESC").show();


// 7. Подсчитать общее количество установок для каждой категории

spark.sql("SELECT Category, SUM(CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT)) as TotalInstalls FROM googleplaystore GROUP BY Category ORDER BY TotalInstalls DESC").show();


// 8. Найти приложения с самой высокой и самой низкой оценкой (по количеству установок)
```

```
        spark.sql("SELECT * FROM (SELECT *, CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT) AS InstallsInt FROM googleplaystore) ORDER BY InstallsInt DESC LIMIT 1").show();

        spark.sql("SELECT * FROM (SELECT *, CAST(REGEXP_REPLACE(Installs, '[+,]', '') AS INT) AS InstallsInt FROM googleplaystore) ORDER BY InstallsInt ASC LIMIT 1").show();


        // 9. Подсчитать средний размер (Size) приложения для каждой категории

        spark.sql("SELECT Category, AVG(CASE " +

                "WHEN Size LIKE '%k' THEN CAST(SUBSTRING(Size, 1, LENGTH(Size)-1) AS FLOAT) / 1024 " +

                "WHEN Size LIKE '%M' THEN CAST(SUBSTRING(Size, 1, LENGTH(Size)-1) AS FLOAT) " +

                "ELSE NULL END) as AverageSize " +

                "FROM googleplaystore GROUP BY Category ORDER BY AverageSize DESC").show();


        // 10. Подсчитать общее количество отзывов (Reviews) для каждой категории

        spark.sql("SELECT Category, SUM(CAST(Reviews AS INT)) as TotalReviews FROM googleplaystore GROUP BY Category ORDER BY TotalReviews DESC").show();


        // Остановка Spark сессии

        spark.stop();

    }

}
```

Часть вывода программы показана на рисунке 1.

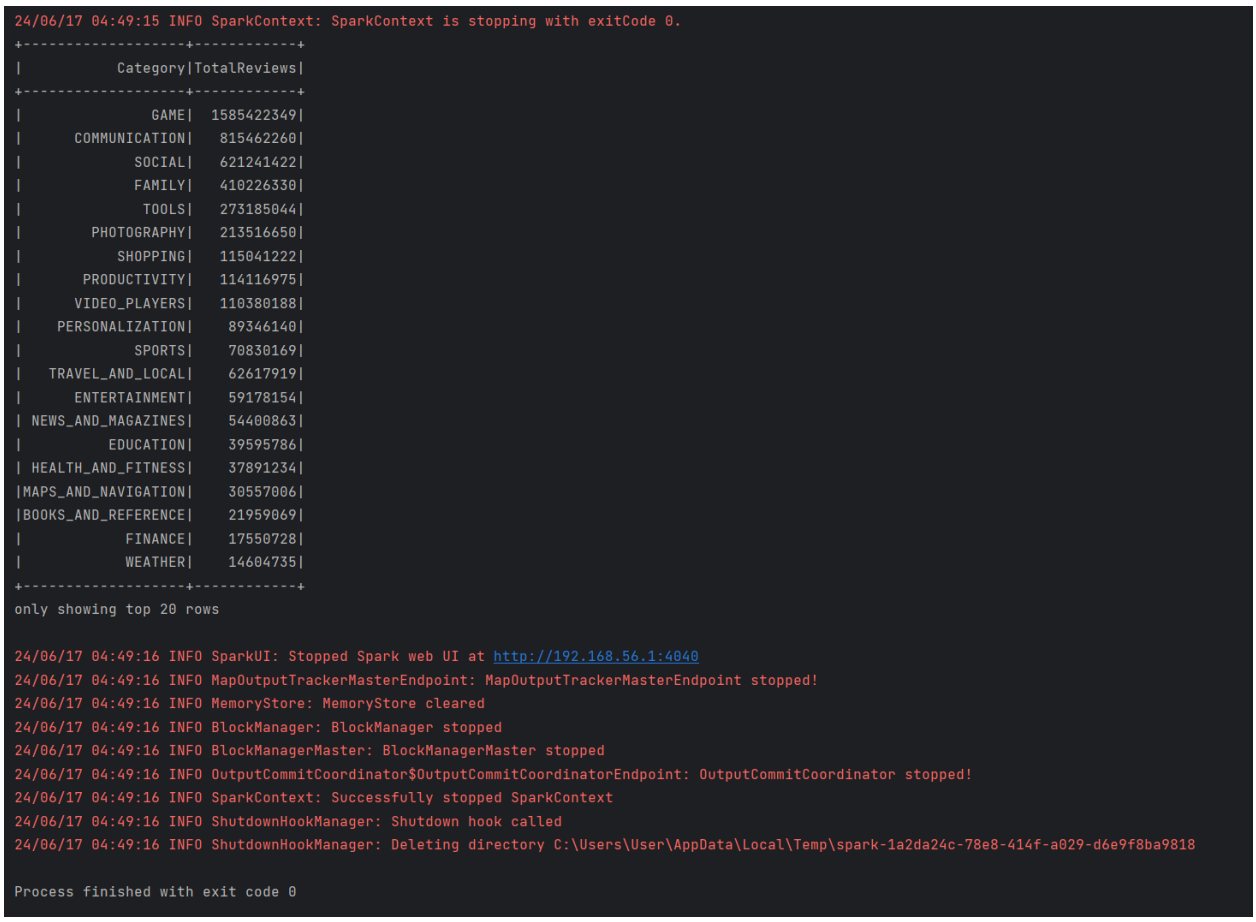

Рисунок 1 – Работа программы

Вывод: был изучен Spark в языке программирования Java.