**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 **09.04.01/07 Интеллектуальные системы анализа, обработки и интерпретации больших данных**

О Т Ч Е Т

|  |  |
| --- | --- |
| **по лабораторной работе № 10** |  |

**Дисциплина:** Spark

**Название:** Языки программирования для работы с большими данными

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | ИУ6-22М |  | Д.М. Карасёв\_\_ |
|  | (Группа) | (Подпись, дата) | (И.О. Фамилия) |
| Преподаватель |  | П.В. Степанов | |
|  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Цель:** ознакомиться с базовыми принципами языка Java для работы с большими данными

**Вариант 1:**

1. Выбрать любой датасет на kaggle.com

2. Сделать 10 выборок данных по выбранной предметной области

**Решение:**

Добавим Dataset c сайта kaggle.com. Dataset представлен на рисунке 1.

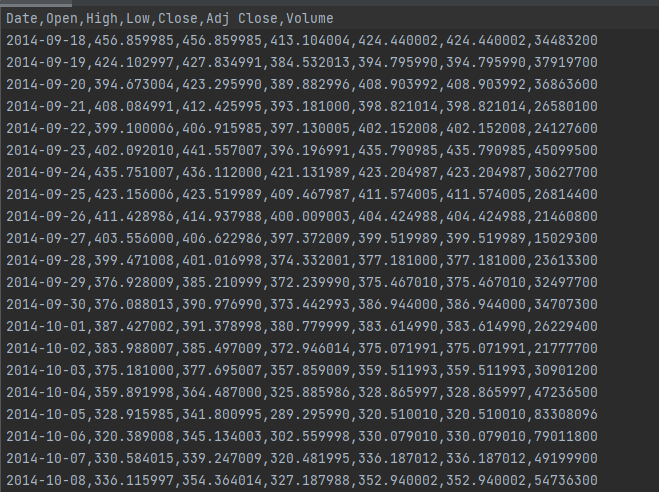


Рисунок 1 - Dataset

package org.example;  
  
import org.apache.spark.SparkConf;  
import org.apache.spark.api.java.JavaSparkContext;  
import org.apache.spark.sql.\*;  
  
public class Main {  
 public static void main(String[] args) {  
 SparkConf conf = new SparkConf().setMaster("local[\*]").setAppName("LR10");  
 JavaSparkContext context = new JavaSparkContext(conf);  
 SparkSession sparkSession = SparkSession.*builder*().appName("LR10").getOrCreate();  
  
 Dataset<Row> dataset = sparkSession.read().format("csv").option("header", "true").load("Bitcoin\_prices.csv");  
  
 dataset.createOrReplaceTempView("bitcoin");  
  
 sparkSession.sql("select \* from bitcoin").show();  
  
 sparkSession.sql("select AVG(Volume) from bitcoin").show();  
  
 sparkSession.sql("select \* from bitcoin where Date between '2014-09-18' and '2014-10-18' ORDER BY Date").show();  
  
 sparkSession.sql("select distinct Volume from bitcoin").show();  
  
 sparkSession.sql("select Date, SUM(Volume) OVER (ORDER BY Date) as VolumeSum from bitcoin").show();  
  
 sparkSession.sql("select \* from bitcoin LIMIT 10").show();  
  
 sparkSession.sql("select \* from bitcoin order by Open asc").show();  
  
 sparkSession.sql("select FIRST(Date), Volume from bitcoin group by Volume").show();  
  
 sparkSession.sql("select \* from bitcoin where Date between '2020-01-01' and '2021-01-01' and Open > 15000").show();  
  
 sparkSession.sql("select \* from bitcoin where Open > 50000 order by 1 asc").show();  
  
 sparkSession.stop();  
  
 }  
}

На рисунке 2 представлен результат выполнения.

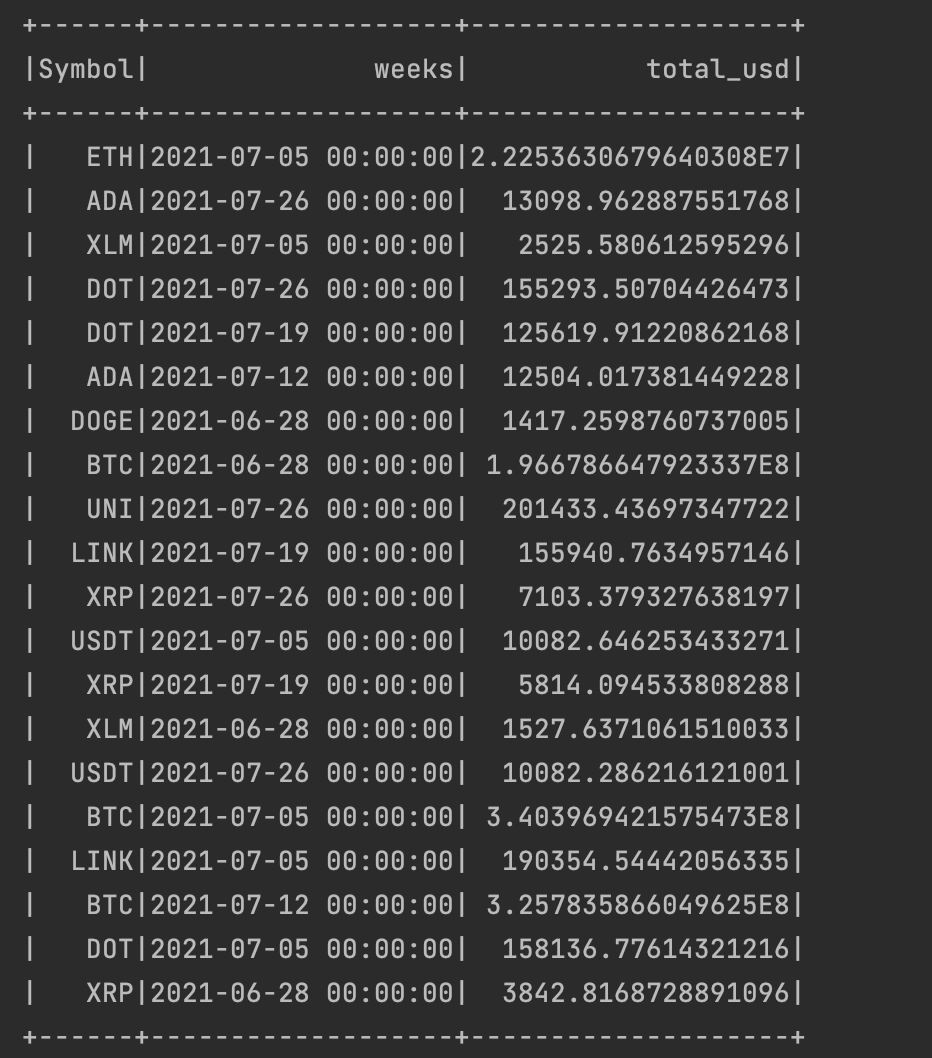


Рисунок 2 – Результат выполнения

**Вывод:** были разработаны классы и методы согласно вариантам.