|  |  |
| --- | --- |
|  | **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**

**Отчет**

**по лабораторной работе № 10**

**Название:** Spark

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

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

Москва, 2023

1. ****Введение и задание****

**Целью лабораторной работы является формирование навыков использования Spark с языком программирования Scala, изучение соответствующих синтаксических конструкций и особенностей работы.**

**Задание.**

**1) Выбрать любой датасет на kaggle.com**

**2) Cделать 10 выборок данных по выбранной предметной области**

1. ****Ход работы****

**В качестве датасета был выбран Russian passenger air service 2007-2020.**

**Выполнение задания лабораторной работы:**

**import org.apache.spark.sql.SparkSession**

**import org.apache.spark.{SparkConf, SparkContext}**

**object CounterDemo {**

**def main(args: Array[String]): Unit = {**

**val conf = new SparkConf().setAppName("lab10").setMaster("local[\*]")**

**val sc = new SparkContext(conf);**

**val spark = SparkSession.builder.appName("lab10").getOrCreate()**

**val df = spark.read.format("com.databricks.spark.csv").option("header",true).load("russian\_air\_service\_CARGO\_AND\_PARCELS.csv")**

**df.createOrReplaceTempView("air")**

**spark.sql("select \* from air").show()**

**spark.sql("select count(\*) from air").show()**

**spark.sql("select count(\*) from air group by `Airport name`").show()**

**spark.sql("select distinct `Airport name` from air where `Airport name` like '%Moscow%'").show()**

**spark.sql("select \* from air where `Airport name` like '%Sheremetyevo%'").show()**

**spark.sql("select Year, sum(January), sum(February), sum(March), sum(April), sum(May), sum(June), sum(July), sum(August), sum(September), sum(October), sum(November), sum(December) from air where `Airport name` like '%Sheremetyevo%' group by Year").show()**

**spark.sql("select `Airport name`, Year, `Whole year` from air where Year = 2019 order by cast(`Whole year` as int) desc").show()**

**spark.sql("select `Airport name`, substring\_index(substring\_index(`Airport coordinates`, ',', 1), '(', -1) as Latitude, substring\_index(substring\_index(`Airport coordinates`, ',', -1), ')', 1) as Longitude from airports;").show()**

**spark.sql("select a1.`Airport name`, a1.`Whole year` as PassengerTraffic\_Year1, a2.`Whole year` as PassengerTraffic\_Year2, (a2.`Whole year` - a1.`Whole year`) as TrafficGrowth from airports as a1 join airports as a2 on a1.`Airport name` = a2.`Airport name` where a1.Year = 2018 and a2.Year = 2019 order by TrafficGrowth desc").show()**

**spark.sql("SELECT Month, AVG(January) AS Avg\_January, AVG(February) AS Avg\_February, AVG(March) AS Avg\_March, AVG(April) AS Avg\_April, AVG(May) AS Avg\_May, AVG(June) AS Avg\_June, AVG(July) AS Avg\_July, AVG(August) AS Avg\_August, AVG(September) AS Avg\_September, AVG(October) AS Avg\_October, AVG(November) AS Avg\_November, AVG(December) AS Avg\_December FROM ( SELECT 'January' AS Month, January FROM airports UNION ALL SELECT 'February' AS Month, February FROM airports UNION ALL SELECT 'March' AS Month, March FROM airports UNION ALL SELECT 'April' AS Month, April FROM airports UNION ALL SELECT 'May' AS Month, May FROM airports UNION ALL SELECT 'June' AS Month, June FROM airports UNION ALL SELECT 'July' AS Month, July FROM airports UNION ALL SELECT 'August' AS Month, August FROM airports UNION ALL SELECT 'September' AS Month, September FROM airports UNION ALL SELECT 'October' AS Month, October FROM airports UNION ALL SELECT 'November' AS Month, November FROM airports UNION ALL SELECT 'December' AS Month, December FROM airports ) AS subquery GROUP BY Month").show()**

**}**

**}**

1. ****Заключение****

**В процессе выполнения лабораторной работы были сформированы навыки использования Spark с языком программирования Scala, были изучены соответствующие синтаксические конструкции и особенности работы.**