|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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**

**Отчет**

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

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-21М |  |  | В.В. Ерискин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание**

Загрузить данные из дата сета используя среду разработки **IntelliJ** **IDEA** и объектно-ориентированный язык программирования **Scala,** сделать несколько запросов по обработке данных.

**Решение**:

На локальном диске компьютера был размещен файл с набором данных об воздушных перевозках в РФ, данные приводились в разрезе месяцев за период с 2007 года января по июль 2020 г., по горизонтали расположены аэропорты , по вертикали сумма выручки за оказанные услуги, а также даны географические координаты аэропортов. Формат предоставленных данных файл формата csv, он представлен на рисунке ниже:

**A screenshot of a computer

Description automatically generated**

**Сформируем запрос на чтение данных из файла на языке программирования Scala:**

import org.apache.spark.sql.SparkSession

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

def main(args: Array[String]): Unit = {  
  
val conf = new SparkConf().setAppName("CounterDemo").setMaster("local[\*]")  
val sc = new SparkContext(conf)  
*val logFile = "C:/Big\_data/readme.md"*val spark = SparkSession.builder().appName(name = "TestApp").getOrCreate()  
val dataFile = spark.read.format("com.databricks.spark.csv").option("header", true).  
 load("\\Users\\vitaly\\Desktop\\BigData\\russian\_passenger\_air\_service\_2.csv")dataFile  
 .createOrReplaceTempView("Air\_serv")

Теперь над найденным дата сетом можно проводить различные математические манипуляции, от простейшего нахождения суммы до построения сложных прогностических моделей.

*Найдем аэропорты с наибольшим объемом оказанных услуг:   
spark.sql("SELECT Whole year, Airport name FROM Air\_serv ORDER BY Whole year DESC").show()  
Найдем аэропорты с наибольшим объемом оказанных услуг за 2017 г.:   
spark.sql("SELECT Whole year = 2017, Airport name FROM Air\_serv ORDER BY Whole year DESC").show()  
Или посчитать объем оказанных услуг конкретного аэропорта за определенный период:   
spark.sql ("SELECT Whole year = 2017 FROM Demography WHERE Whole year=2017").show()*

*spark.stop()*

**Вывод:**

Задание выполнено, программы компилируются и успешно выполняются.