



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 Интеллектуальные системы анализа,  
обработки и интерпретации больших данных

## ОТЧЕТ

по лабораторной работе № 10

Вариант 13

Название: Spark

Дисциплина: Языки программирования для работы с большими данными

Студент

ИУ6-22М

(Группа)

(Подпись, дата)

В.А.Ловцов

(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 выборки данных по выбранной предметной области

Был выбран набор данных об авиаперевозках.

Код класса Main:

```
package atomic.engineering

import org.apache.spark.sql.SparkSession

object SparkDemo {

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

        val spark =
SparkSession.builder().master("local[1]").appName("AtomicEngin
eering").getOrCreate()

        println("name:" + spark.sparkContext.appName)

        val path =
"src/russian_air_service_CARGO_AND_PARCELS.csv"

        val df = spark.read.option("header", "true").csv(path)

        // df.show()

        df.createOrReplaceTempView("data")

        var sqlDF = spark.sql("SELECT * FROM data limit 10")

        sqlDF.show()

        sqlDF.printSchema
```

```
sqlDF = spark.sql("SELECT COUNT(*) AS row_count FROM  
data")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT DISTINCT Year FROM data")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT * FROM data WHERE `Airport  
name` LIKE 'B%'")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT `Airport name`, Year FROM  
data LIMIT 10")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT * FROM data WHERE February >  
600")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT Year, SUM(September) AS total  
FROM data GROUP BY Year")
```

```
sqlDF.show()
```

```
sqlDF = spark.sql("SELECT * FROM data ORDER BY June  
DESC limit 10")
```

```

sqlDF.show()

sqlDF = spark.sql("SELECT AVG(October) AS average,
MAX(October) AS maximum FROM data")

sqlDF.show()

sqlDF = spark.sql("SELECT COUNT(DISTINCT `Airport
name`) AS airports FROM data")

sqlDF.show()

sqlDF = spark.sql("SELECT `Airport name`, SUM(December)
AS total_load FROM data GROUP BY `Airport name` ORDER BY
total_load DESC LIMIT 10")

sqlDF.show()

spark.stop()

}

}

```

Работа программы показана на рисунке 1.

Airport name	Year	January	February	March	April	May	June	July	August	September	October	November	December	Whole year	Airport coordinates
Abakan	2020	43.58	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('91.3997...))
Aikhal	2020	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('111.543...))
Loss	2020	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('125.398...))
Amderma	2020	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('61.5774...))
Anadyr (Carbon)	2020	122.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('177.738...))
Anapa (Vitjazovo)	2020	49.08	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('37.3415...))
Apatite (Khibiny)	2020	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('33.5819...))
Arkhangelsk (Vask...)	2020	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('40.7067...))
Arkhangelsk (Talagy)	2020	96.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('40.7148...))
Astrakhan (Narima...)	2020	53.55	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(Decimal('47.9998...))

Рисунок 1 – Работа программы

**Вывод:** был изучен Spark, написана программа для осуществления запросов к данным