



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

О Т Ч Е Т

по лабораторной работе № 10

Вариант 12

Название: Scala Spark

Дисциплина: Языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

П.А. Мартынюк

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2022

Цель работы:

Получение навыков работы со Scala Spark.

Выполнение:

Задание:

1. Выбрать любой датасет (взяв датасет из курсового проекта, тема «Поликлиника»)
2. Сделать 10 выборок данных на ваше усмотрение

Листинг выполнения одного из запросов (файл spark.scala)

```
import org.apache.spark.sql.Session

object CounterDemo {
  def main(args: Array[String]): Unit = {
    val conf = new
SparkConf().setAppName("CounterDemo").setMaster("local[*]")
    val sc = new SparkContext(conf);
    val spark = SparkSession.builder.appName("Test app").getOrCreate()
    val path_1 = "/home/polina/Документы/scripts/visits.csv"
    val df_visits = spark.read.option("header", "true").csv(path_1)
    val path_2 = "/home/polina/Документы/scripts/patients.csv"
    val df_patients = spark.read.option("header", "true").csv(path_2)
    df_visits.createOrReplaceTempView("visits")
    df_patients.createOrReplaceTempView("patients")
    spark.sql("SELECT patient_age, count(patient_age) as counter FROM
patients, visits WHERE patients.patient_id = visits.patient_id AND
diagnosis = 'Грипп' GROUP BY patient_age").show()
    spark.stop()
  }
}
```

```
polina@polina-VirtualBox: /usr/local/spark-3.2.1
at scala.concurrent.impl.Promise.liftedTree1$1(Promise.scala:33)
at scala.concurrent.impl.Promise.$anonfun$transform$1(Promise.scala:33)
at scala.concurrent.impl.CallbackRunnable.run(Promise.scala:64)
at java.util.concurrent.ForkJoinTask$RunnableExecuteAction.exec(ForkJoinTask.java:1402)
at java.util.concurrent.ForkJoinTask.doExec(ForkJoinTask.java:289)
at java.util.concurrent.ForkJoinPool$WorkQueue.runTask(ForkJoinPool.java:1056)
at java.util.concurrent.ForkJoinPool.runWorker(ForkJoinPool.java:1692)
at java.util.concurrent.ForkJoinWorkerThread.run(ForkJoinWorkerThread.java:175)

scala> val path_1 = "/home/polina/Документы/scripts/visits.csv"
path_1: String = /home/polina/Документы/scripts/visits.csv

scala> val df_visits = spark.read.option("header", "true").csv(path_1)
df_visits: org.apache.spark.sql.DataFrame = [patient_id: string, visit_id: string ... 6 more fields]

scala>

scala> val path_2 = "/home/polina/Документы/scripts/patients.csv"
path_2: String = /home/polina/Документы/scripts/patients.csv

scala> val df_patients = spark.read.option("header", "true").csv(path_2)
df_patients: org.apache.spark.sql.DataFrame = [patient_id: string, patient_age: string ... 1 more field]

scala> df_visits.createOrReplaceTempView("visits")

scala> df_patients.createOrReplaceTempView("patients")

scala> spark.sql("SELECT patient_age, count(patient_age) as counter FROM patients, visits WHERE patients.patient_id = visits.pat
ent_id AND diagnosis = 'Грипп' GROUP BY patient_age").show()
+-----+-----+
|patient_age|counter|
+-----+-----+
|27|1|
|61|1|
|40|1|
+-----+-----+

scala>
```

Рисунок 1 - Результат выполнения запроса

Ссылка на программное решение:

Программное решение представлено в репозитории распределённой системы управления версиями Git:

<https://github.com/Owlfeather/JavaMagisterCourse/tree/main/Lab10/src>

Вывод:

При выполнении лабораторной работы были получены навыки работы со Scala Spark.