BDA

Assignment 2

Reshan Faraz (PhD19006)

Prakrati Gupta(MT20014)

Answer 1 : Loading the dependency.

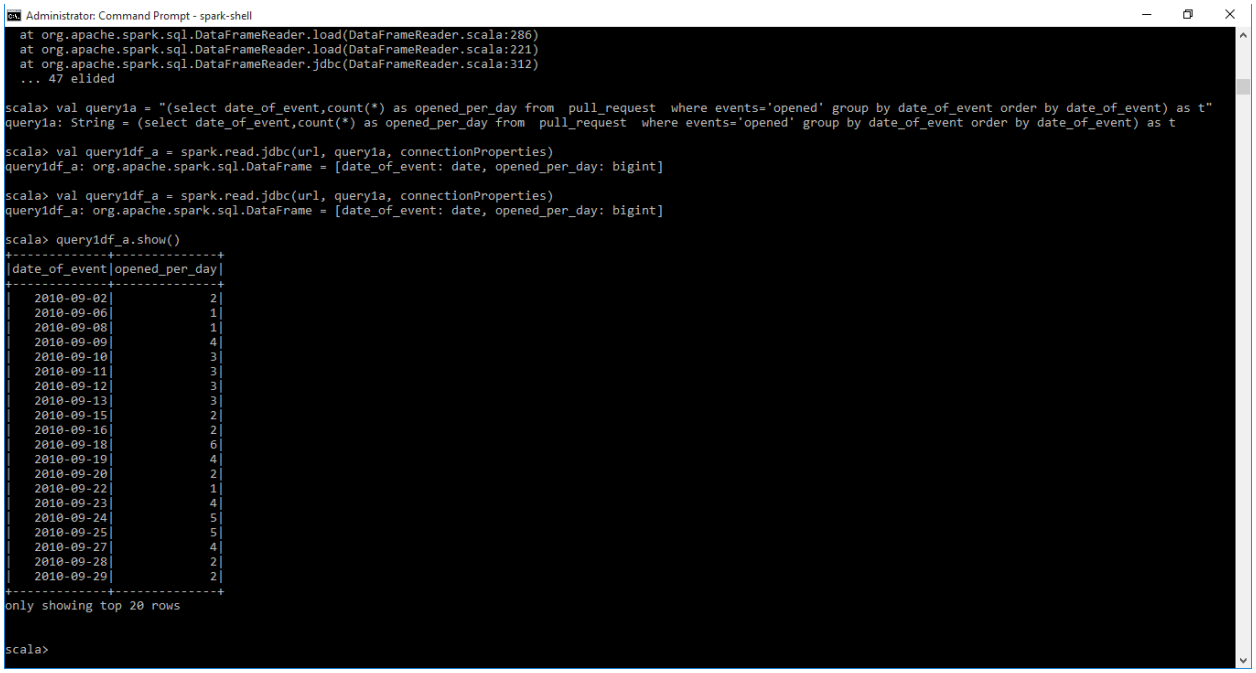
```
scala> :require D:\myy.jar
Added 'D:\myy.jar' to classpath.

scala> import java.util.Properties
import java.util.Properties
```

```
scala> val connectionProperties = new Properties()
connectionProperties: java.util.Properties = {}

scala> connectionProperties.setProperty("Driver", "org.postgresql.Driver")
res14: Object = null
```

```
scala> val url = "jdbc:postgresql://localhost:5432/BDA?user=postgres&password=[redacted]"
url: String = jdbc:postgresql://localhost:5432/BDA?user=postgres&password=[redacted]
```



We executed all the query from .scala file (attached).

In the spark shell just run the following command to execute all the queries

1 - *:load <path of HelloWorld.scala>*

2 - *HelloWorld.main(Array())*

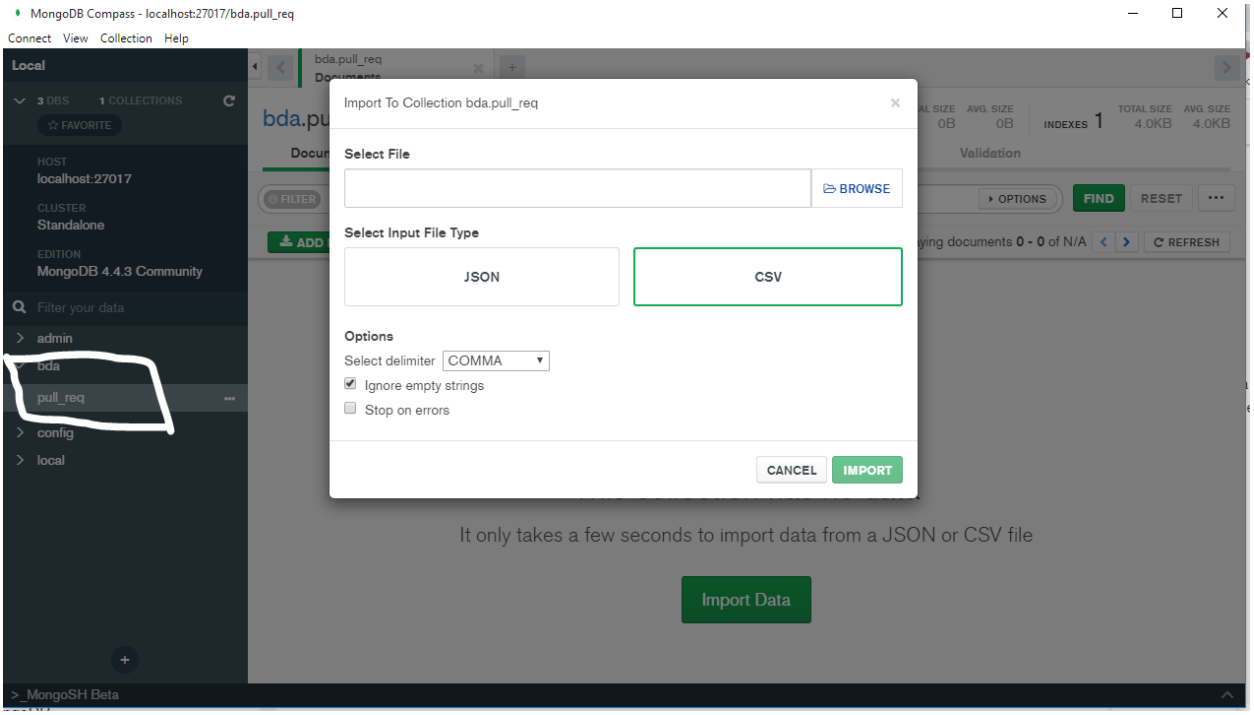
Executor 1 : 9 Second

Executor ID ▲	Address	Status	RDD Blocks	Storage Memory	Disk Used	Cores	Active Tasks	Failed Tasks	Complete Tasks	Total Tasks	Task Time (GC Time)	Input	Shuffle Read	Shuffle Write	Thread Dump
driver	DESKTOP-C8BK0QK:50826	Active	0	15.6 KiB / 366.3 MiB	0.0 B	4	0	0	15	15	9 s (2 s)	0.0 B	0.0 B	0.0 B	Thread Dump

---

Answer 2 :

After installing mongoDb we import the csv data in database name using mongoCompass :


MongoDB Compass - localhost:27017/bda.pull_req
Connect View Collection Help
Local
3 DBS 1 COLLECTIONS
FAVORITE
HOST
localhost:27017
CLUSTER
Standalone
EDITION
MongoDB 4.4.3 Community
Filter your data
admin
bda
pull_req
config
local
bda.pull_req
Import To Collection bda.pull_req
Select File
BROWSE
Select Input File Type
JSON
CSV
Options
Select delimiter COMMA
Ignore empty strings
Stop on errors
CANCEL
IMPORT
FILTER
OPTIONS
FIND
RESET
ADD
REFRESH
Validation
It only takes a few seconds to import data from a JSON or CSV file
Import Data
>_MongoSH Beta

Specify Fields and Types

	4 String	datanoise String	opened String	2010-09-02 3:34:17 String
1	5	marsuboss	opened	2010-09-02 7:14:12
2	6	m3talsmith	opened	2010-09-06 16:07:08
3	7	sferik	opened	2010-09-08 19:09:56
4	8	sferik	opened	2010-09-09 4:33:23
5	8	dtrasbo	discussed	2010-09-09 4:44:25
6	8	brianmario	discussed	2010-09-09 4:53:01
7	8	sferik	discussed	2010-09-09 14:22:03
8	9	rsim	opened	2010-09-09 19:18:48
9	10	simonjefford	opened	2010-09-09 19:36:42
10	11	bcardarella	opened	2010-09-09 20:48:38

Import completed 273,088 (100%)

DONE

```
> show dbs;
admin   0.000GB
bda_1   0.008GB
config  0.000GB
local   0.000GB
> use bda_1
switched to db bda_1
> show collections
a2
> db.a2.find().pretty()
{
        "_id" : ObjectId("6049cd08cfdf4d129cf4d750"),
        "num_req" : 4,
        "name_a" : "datanoise",
        "type_c" : "opened",
        "date_o" : ISODate("2010-09-01T22:04:17Z")
}
{
        "_id" : ObjectId("6049cd08cfdf4d129cf4d751"),
        "num_req" : 5,
```

```
scala> import com.mongodb.spark._
import com.mongodb.spark._

scala> import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.SparkSession

scala> val spark = SparkSession.builder()
spark: org.apache.spark.sql.SparkSession.Builder = org.apache.spark.sql.SparkSession$Builder@59f4df4e

scala> spark.master("local")
res0: org.apache.spark.sql.SparkSession.Builder = org.apache.spark.sql.SparkSession$Builder@59f4df4e

scala> spark.appName("MongoSparkConnectorIntro")
res1: org.apache.spark.sql.SparkSession.Builder = org.apache.spark.sql.SparkSession$Builder@59f4df4e

scala> spark.config("spark.mongodb.input.uri", "mongodb://127.0.0.1/bda_1.a2")
res2: org.apache.spark.sql.SparkSession.Builder = org.apache.spark.sql.SparkSession$Builder@59f4df4e

scala> spark.config("spark.mongodb.output.uri", "mongodb://127.0.0.1/bda_1.a2")
res3: org.apache.spark.sql.SparkSession.Builder = org.apache.spark.sql.SparkSession$Builder@59f4df4e

scala> spark.getOrCreate()
21/03/11 23:11:22 WARN SparkSession$Builder: Using an existing SparkSession; some spark core configurations may not take
 effect.
res4: org.apache.spark.sql.SparkSession = org.apache.spark.sql.SparkSession@48012514

scala> val rdd = MongoSpark.load(sc)
```

Answer 3 :

Loading the CSV to the hadoop file system

```
C:\Users\reshanriq> hdfs dfs -put "C:\Users\reshanriq\Downloads\Neww\a.csv"  "C:\hadoop\data\datanode"
The filename, directory name, or volume label syntax is incorrect.

C:\Users\reshanriq>cd "C:\hadoop\data\datanode"

C:\hadoop\data\datanode>ls
a.csv

C:\hadoop\data\datanode>
```

hdfs dfs -put "C:\Users\reshanriq\Downloads\Neww\a.csv" "C:\hadoop\data\datanode"

```
>>> from pyspark.sql import SparkSession
>>> sparkSession = SparkSession.builder.appName("example-pyspark-read-and-write").getOrCreate()
>>> df = spark.read.format("csv").load("/a.csv")
>>> df.write.csv("/ss.csv")
>>> df_load = sparkSession.read.csv('/ss.csv')
>>> df_load.show()
+----+-----------+-----------+-------------------+
| _c0|        _c1|        _c2|                _c3|
+----+-----------+-----------+-------------------+
|4254|jonleighton|synchronize|2012-04-27 10:09:08|
|4367|jonleighton|synchronize|2012-04-27 10:09:08|
|4396|jonleighton|synchronize|2012-04-27 10:09:08|
|4428|jonleighton|synchronize|2012-04-27 10:09:08|
|4431|jonleighton|synchronize|2012-04-27 10:09:09|
|4452|jonleighton|synchronize|2012-04-27 10:09:09|
|4456|jonleighton|synchronize|2012-04-27 10:09:09|
|4496|jonleighton|synchronize|2012-04-27 10:09:10|
|4526|jonleighton|synchronize|2012-04-27 10:09:10|
|4540|jonleighton|synchronize|2012-04-27 10:09:11|
|4625|jonleighton|synchronize|2012-04-27 10:09:11|
|4631|jonleighton|synchronize|2012-04-27 10:09:11|
|4688|jonleighton|synchronize|2012-04-27 10:09:12|
|4728|jonleighton|synchronize|2012-04-27 10:09:13|
|4748|jonleighton|synchronize|2012-04-27 10:09:13|
|4759|jonleighton|synchronize|2012-04-27 10:09:13|
|4785|jonleighton|synchronize|2012-04-27 10:09:13|
|4795|jonleighton|synchronize|2012-04-27 10:09:13|
|4833|jonleighton|synchronize|2012-04-27 10:09:15|
|4835|jonleighton|synchronize|2012-04-27 10:09:15|
+----+-----------+-----------+-------------------+
```