Raj Kamal Yadav
2016076

Rajat Bansal
2016260

INDRAPRASTHA INSTITUTE of INFORMATION TECHNOLOGY DELHI

Data Format

- id serial Primary Key Integer
- log_lvl VARCHAR
- timestamp VARCHAR
- dwnlder_id VARCHAR
- retrval_stage VARCHAR
- repo VARCHAR
- mssg VARCHAR

Apache Spark + PSQL

```
import java.util.Properties

:require /home/bansal/Downloads/postgresql-42.2.9.jar
Class.forName("org.postgresql.Driver") != null
val url = "jdbc:postgresql://localhost:5432/postgres"
val connectionProperties = new Properties()
connectionProperties.setProperty("Driver", "org.postgresql.Driver")
connectionProperties.setProperty("user", "postgres")
connectionProperties.setProperty("password", "abcdefg")
val person = "(SELECT COUNT(*) FROM schemma_bda_1.mytable) as q1"
val personDf = spark.read.jdbc(url, person, connectionProperties)
personDf.show()
```

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
3	26.41	2.44
4	27.52	2.79
5	26.38	3.86
6	26.49	2.51
7	25.74	3.75
8	31.20	4.15
9	26.24	3.50

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
10	27.28	6.29
11	381.92	31.82
13	28.26	4.18
14	26.32	2.50

Apache Spark + MongoDB

```
import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.{DataFrame, Encoders}
import com.mongodb.spark._
import sys.process._
import org.apache.spark.sql._
import org.apache.spark._
import com.mongodb.spark.config._
import org.bson.Document
import scala.util.parsing.json.JSONObject
import org.apache.spark.sql



val spark= SparkSession.builder().appName("MongoSparkDataFrame").master("local[*]").config("spark.mongodb.input.uri","mongodb://127.0.0.1/mydb.tab").config
("spark.mongodb.output.uri","mongodb://127.0.0.1/mydb.tab").getOrCreate()

val sc= spark.sparkContext

val readConfig = ReadConfig(Map("collection" -> "tab", "readPreference.name" -> "secondaryPreferred"), Some(ReadConfig(sc)))
/*val readConfig2 = ReadConfig(Map("collection" -> "interesting", "readPreference.name" -> "secondaryPreferred"), Some(ReadConfig(sc)))*/

val customRdd = MongoSpark.load(sc, readConfig)
/*val customRdd2 = MongoSpark.load(sc, readConfig2)*/
import spark.implicits._
val df = customRdd.toDF
/*val df2 = customRdd2.toDF*/
df.show()
/*df2.show()*/
df.createOrReplaceTempView("records")
/*df2.createOrReplaceTempView("interesting")*/
```

Q3.

```
ans3: org.apache.spark.sql.DataFrame = [count(1): bigint]
+--------+
|count(1)|
+--------+
|  132158|
+--------+
```

Q4.

```
ans4: org.apache.spark.sql.DataFrame = [count(DISTINCT repo): bigint]
+--------------------+
|count(DISTINCT repo)|
+--------------------+
|                6252|
+--------------------+
```

Q5.

```
ans5: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|85528|
| ghtorrent-4|19046|
|ghtorrent-18|18950|
|ghtorrent-10|18926|
|ghtorrent-40|18911|
|ghtorrent-39|18616|
|ghtorrent-38|18614|
|ghtorrent-47|18605|
| ghtorrent-1|18465|
|ghtorrent-24|18452|
+------------+-----+
```

Q5.

```
ans5: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|85528|
| ghtorrent-4|19046|
|ghtorrent-18|18950|
|ghtorrent-10|18926|
|ghtorrent-40|18911|
|ghtorrent-39|18616|
|ghtorrent-38|18614|
|ghtorrent-47|18605|
| ghtorrent-1|18465|
|ghtorrent-24|18452|
+------------+-----+
```

Q6.

```
ans6: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|79623|
|ghtorrent-21| 1378|
|ghtorrent-40| 1134|
|ghtorrent-18|  368|
|ghtorrent-42|  357|
| ghtorrent-9|  356|
| ghtorrent-4|  352|
|ghtorrent-25|  342|
|ghtorrent-22|  333|
| ghtorrent-6|  332|
+------------+-----+
```

Q7.

```
ans7: org.apache.spark.sql.DataFrame = [C: bigint, H: string]
+------+---+
|     C|  H|
+------+---+
|255916| 01|
+------+---+
```

Q8.

```
ans8: org.apache.spark.sql.DataFrame = [c: bigint, repo: string]
+-----+--------------------+
|    c|                repo|
+-----+--------------------+
|79524| greatfakeman/Tabchi|
| 4084|mithro/chromium-i...|
| 2575| shuhongwu/hockeyapp|
| 2299|obophenotype/huma...|
| 1149|kubernetes/kubern...|
+-----+--------------------+
```

Q9.

```
ans9: org.apache.spark.sql.DataFrame = [cnt: bigint, fstrng: string]
+-----+--------------------+
|  cnt|              fstrng|
+-----+--------------------+
|79523|quest. URL: https...|
+-----+--------------------+
```

Q10.

```
Time taken without indexing: 72712.648838ms.
```

Q11.

```
ans11: org.apache.spark.sql.DataFrame = [count(DISTINCT repo): bigint]
+--------------------+
|count(DISTINCT repo)|
+--------------------+
|               95191|
+--------------------+

Time taken without indexing: 436922.31861ms.
```

Q12.

```
ans13: org.apache.spark.sql.DataFrame = [count(1): bigint]
+--------+
|count(1)|
+--------+
|   87938|
+--------+
```

Q13.

```
ans14: org.apache.spark.sql.DataFrame = [c: bigint, repp: string]
+---+-----------+
|  c|       repp|
+---+-----------+
|740|hello-world|
|309|       test|
|166|       demo|
| 88|       Test|
| 47|          -|
| 26|      hello|
| 24|   Ruby_k59|
| 20|    website|
| 16|   TestRepo|
| 15|    angular|
+---+-----------+
```

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
3	39.67	26.18
4	26.62	21.57
5	45.54	46.61
6	100.13	95.30
7	60.51	53.25
8	53.39	44.36
9	65.76	56.11

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
10	5.91	4.83
11	11.75	14.22
13	53	55.84
14	88.82	85.52

Apache Spark + HDFS

```
import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.{DataFrame, Encoders}
import com.mongodb.spark._
import sys.process._
import org.apache.spark.sql._
import org.apache.spark.sql
import org.apache.spark._

import spark.implicits._
val df= spark.read.csv("hdfs://localhost:9000/mytable.csv").toDF
val df2= spark.read.csv("hdfs://localhost:9000/interesting.csv").toDF
df.show()
df2.show()
df.createOrReplaceTempView("mytable")
df2.createOrReplaceTempView("interesting")
```

```
+-------------+-----+
|          _c3|    c|
+-------------+-----+
| ghtorrent-13|85528|
|  ghtorrent-4|19046|
| ghtorrent-18|18950|
| ghtorrent-10|18926|
| ghtorrent-40|18911|
| ghtorrent-39|18616|
| ghtorrent-38|18614|
| ghtorrent-47|18605|
|  ghtorrent-1|18465|
| ghtorrent-24|18452|
+-------------+-----+

Time taken: 38073.956329ms.
time6: Long = 34677398599879
person6: org.apache.spark.sql.DataFrame = [_c3: string, c: bigint]
+-------------+-----+
|          _c3|    c|
+-------------+-----+
| ghtorrent-13|79623|
| ghtorrent-21| 1378|
| ghtorrent-40| 1134|
| ghtorrent-18|  368|
|  ghtorrent-9|  356|
| ghtorrent-42|  355|
|  ghtorrent-4|  352|
| ghtorrent-25|  342|
| ghtorrent-22|  333|
|  ghtorrent-6|  332|
+-------------+-----+

Time taken: 41751.060037ms.
time7: Long = 34719503807823
person7: org.apache.spark.sql.DataFrame = [C: bigint, H: string]
+------+---+
|     C|  H|
+------+---+
|255916| 01|
+------+---+

Time taken: 36944.022853ms.
time8: Long = 34756877129099
person8: org.apache.spark.sql.DataFrame = [c: bigint, _c5: string]
+-----+--------------------+
|    c|                 _c5|
+-----+--------------------+
|79524| greatfakeman/Tabchi|
| 4084|mithro/chromium-i...|
| 2575| shuhongwu/hockeyapp|
| 2299|obophenotype/huma...|
| 1149|kubernetes/kubern...|
+-----+--------------------+

Time taken: 43459.329619ms.
```

```
iamawesome-inspiron-5559: /usr/local/spark/spark-3.0.0-preview2-bin-hadoop2.7/bin
time3: Long = 35495603426558
person3: org.apache.spark.sql.DataFrame = [count(1): bigint]
+--------+
|count(1)|
+--------+
|  132158|
+--------+

Time taken: 89960.903852ms.
time4: Long = 35586059821262
person4: org.apache.spark.sql.DataFrame = [count(DISTINCT _c5): bigint]
+--------------------+
|count(DISTINCT _c5)|
+--------------------+
|                6252|
+--------------------+

Time taken: 102179.426643ms.
time5: Long = 35688455756136
person5: org.apache.spark.sql.DataFrame = [_c3: string, c: bigint]
+-------------+-----+
|          _c3|    c|
+-------------+-----+
| ghtorrent-13|85528|
|  ghtorrent-4|19046|
| ghtorrent-18|18950|
| ghtorrent-10|18926|
| ghtorrent-40|18911|
| ghtorrent-39|18616|
| ghtorrent-38|18614|
| ghtorrent-47|18605|
|  ghtorrent-1|18465|
| ghtorrent-24|18452|
+-------------+-----+

Time taken: 50887.906048ms.
time6: Long = 35739538067859
person6: org.apache.spark.sql.DataFrame = [_c3: string, c: bigint]
+-------------+-----+
|          _c3|    c|
+-------------+-----+
| ghtorrent-13|79623|
| ghtorrent-21| 1378|
| ghtorrent-40| 1134|
| ghtorrent-18|  368|
|  ghtorrent-9|  356|
| ghtorrent-42|  355|
|  ghtorrent-4|  352|
| ghtorrent-25|  342|
| ghtorrent-22|  333|
|  ghtorrent-6|  332|
+-------------+-----+

Time taken: 92600.49709ms.
time7: Long = 35832390698440
person7: org.apache.spark.sql.DataFrame = [C: bigint, H: string]
+------+---+
```

Q3.

```
ans3: org.apache.spark.sql.DataFrame = [count(1): bigint]
+--------+
|count(1)|
+--------+
|  132158|
+--------+
```

Q4.

```
ans4: org.apache.spark.sql.DataFrame = [count(DISTINCT repo): bigint]
+--------------------+
|count(DISTINCT repo)|
+--------------------+
|                6252|
+--------------------+
```

Q5.

```
ans5: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|85528|
| ghtorrent-4|19046|
|ghtorrent-18|18950|
|ghtorrent-10|18926|
|ghtorrent-40|18911|
|ghtorrent-39|18616|
|ghtorrent-38|18614|
|ghtorrent-47|18605|
| ghtorrent-1|18465|
|ghtorrent-24|18452|
+------------+-----+
```

Q5.

```
ans5: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|85528|
| ghtorrent-4|19046|
|ghtorrent-18|18950|
|ghtorrent-10|18926|
|ghtorrent-40|18911|
|ghtorrent-39|18616|
|ghtorrent-38|18614|
|ghtorrent-47|18605|
| ghtorrent-1|18465|
|ghtorrent-24|18452|
+------------+-----+
```

Q6.

```
ans6: org.apache.spark.sql.DataFrame = [dwnlder_id: string, c: bigint]
+------------+-----+
|  dwnlder_id|    c|
+------------+-----+
|ghtorrent-13|79623|
|ghtorrent-21| 1378|
|ghtorrent-40| 1134|
|ghtorrent-18|  368|
|ghtorrent-42|  357|
| ghtorrent-9|  356|
| ghtorrent-4|  352|
|ghtorrent-25|  342|
|ghtorrent-22|  333|
| ghtorrent-6|  332|
+------------+-----+
```

Q7.

```
ans7: org.apache.spark.sql.DataFrame = [C: bigint, H: string]
+------+---+
|     C|  H|
+------+---+
|255916| 01|
+------+---+
```

Q8.

```
ans8: org.apache.spark.sql.DataFrame = [c: bigint, repo: string]
+-----+--------------------+
|    c|                repo|
+-----+--------------------+
|79524| greatfakeman/Tabchi|
| 4084|mithro/chromium-i...|
| 2575| shuhongwu/hockeyapp|
| 2299|obophenotype/huma...|
| 1149|kubernetes/kubern...|
+-----+--------------------+
```

Q9.

```
ans9: org.apache.spark.sql.DataFrame = [cnt: bigint, fstrng: string]
+-----+--------------------+
|  cnt|              fstrng|
+-----+--------------------+
|79523|quest. URL: https...|
+-----+--------------------+
```

Q10.

```
Time taken without indexing: 72712.648838ms.
```

Q11.

```
ans11: org.apache.spark.sql.DataFrame = [count(DISTINCT repo): bigint]
+--------------------+
|count(DISTINCT repo)|
+--------------------+
|               95191|
+--------------------+

Time taken without indexing: 436922.31861ms.
```

Q12.

```
ans13: org.apache.spark.sql.DataFrame = [count(1): bigint]
+--------+
|count(1)|
+--------+
|   87938|
+--------+
```

Q13.

```
ans14: org.apache.spark.sql.DataFrame = [c: bigint, repp: string]
+---+-----------+
|  c|       repp|
+---+-----------+
|740|hello-world|
|309|       test|
|166|       demo|
| 88|       Test|
| 47|          -|
| 26|      hello|
| 24|   Ruby_k59|
| 20|    website|
| 16|   TestRepo|
| 15|    angular|
+---+-----------+
```

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
3	39.04	89.96
4	35.65	102.17
5	38.07	50.88
6	41.75	92.60
7	36.94	63.54
8	43.45	47.75
9	61.0	57.0

Query No.	Executor 1(./bin/spark-shell --num-executors 1) (In seconds)	Executor 2(./bin/spark-shell --num-executors 2) (In seconds)
10	---	
11	---	
13	49.54	58.14
14	46.13	63.53

Learnings from the Assignment

- We learnt about RDD's and why they are faster than normal databases(can be queried in parallel and reside in main memory.)
- Learnt that Mongodb supports secondary indexing but provides its own primary index by default whereas HDFS doesn't support indexing.
- The most amazing thing we learnt was that although MongoDB alone is a lot faster than PSQL, with Apache Spark PSQL obtained a speedup of 5-10 times due to parallelization of queries on all cores of all cluster nodes making it faster than Mongo which is based on NoSQL technology and doesn't make use of the speedup.

Challenges faced in solving the assignment

- Connecting HDFS and MongoDB with Apache Spark.
- Absence of Indexing(secondary indexing in the case of MongoDB and no indexing in HDFS) from inside the Apache Spark environment, have to collect the results from the offline environment.
- Absence of certain operations like write(Update, Alter, Create) in Apache Spark.
- Unable to install Hive on the top of Hadoop environment to execute queries in SQL-like environment, instead used the Apache Spark for the same.
- Instead of using the log file for importing the data so big we used the csv from the previous assignment for the same for all the parts in the assignment.
-

Resources Used

StackOverFlow for How to measure the time taken for the execution of the Query.