DATASTAX ACADEMY

Exercise 7.1 – Spark SQL

The purpose of this exercise is to practice using Spark SQL to solve simply queries.

In this exercise, you will:

- Understand how to start the Spark SQL client
- Gain familiarity with the Spark SQL syntax

Part 1: Start the Spark SQL client

Starting the Spark SQL client is simple.

1. Open a terminal window
2. Type `dse spark-sql`
3. After a brief delay, the `spark-sql` command line should appear as follows:

```
ubuntu@ds420-vm:~$ dse spark-sql
The log file is at /home/ubuntu/.spark-sql-shell.log
spark-sql>
```

Part 2: Spark SQL Syntax

Review the Spark SQL syntax. Below is a list of the statements and clauses that can be used in a Spark SQL query:

```
SELECT [DISTINCT] [column names]|[wildcard]
FROM [keyspace name.]table name
[JOIN clause table name ON join condition]
[WHERE condition]
[GROUP BY column name]
[HAVING conditions]
[ORDER BY column names [ASC | DSC]]
```

Note that just like relational SQL and unlike `CQL`, `GROUP BY`, `HAVING` and `ORDER BY` are supported. Also, note that you must specify the keyspace for your query to work as expected.

Below is a short list of available functions that you may find useful as part of this exercise:

`AVG(field_name)`: Returns the average of all values in the column

`COUNT(* or field_name)`: Returns the count of all items in the column or row

`SUM(field_name)`: Returns the sum of all values in the column

Part 3: Writing the Queries

Write queries that answer the following questions:

1. How many user ratings are there?

```
SELECT count(*) FROM killrvideo_spark.video_ratings_by_user;
```

2. Who are the top 10 raters (in terms of quantity of ratings)?

```
SELECT userid, count(*) as numRatings
FROM killrvideo_spark.video_ratings_by_user
GROUP BY userid
ORDER BY numRatings DESC LIMIT 10;
```

3. What are the average ratings of the top 10 raters?

```
SELECT userid, count(*) as numRatings, avg(rating) as avgRatings
FROM Killrvideo_spark.video_ratings_by_user
GROUP BY userid
ORDER BY numRatings DESC LIMIT 10;
```

4. Which raters are posting the most negative reviews? We are looking for raters with more than seven reviews and the average of those reviews is less than 2.5.

```
SELECT userid, count(rating) as theCount, AVG(rating) as theAvg
FROM killrvideo_spark.video_ratings_by_user
GROUP BY userid
HAVING theCount > 7 AND theAvg < 2.5
ORDER BY theCount DESC, theAvg ASC;
```

5. What are the names of raters posting the most negative reviews?

```
SELECT firstname, lastname, count(rating) as theCount, AVG(rating) as
   theAvg FROM killrvideo_spark.users NATURAL JOIN
   killrvideo_spark.video_ratings_by_user
GROUP BY userid, firstname, lastname
HAVING count(rating) > 7 AND AVG(rating) < 2.5;
```