Spring 2019 CS157C:Introduction to NoSQL Databases

Programming Assignment 4

Instructor: Dr. Kim

Due: Saturday, April 27 11:59 pm

Configure a VM in a way that prompts from the VM show your initial or any unique identifier - This will be strictly enforced. If you are not using a VM for this assignment, which is allowed, show me the way that the screenshot(s) are from you.

Part I - 10 points:

1. Install Cassandra in your VM. The following is the instruction for the Ubuntu VM created by Vagrantfile. (You are allowed not to use the Ubuntu VM.)

```
% sudo apt-get update -y
% sudo apt-get upgrade -y
% sudo reboot

1. Install Java 8
https://www.liquidweb.com/kb/how-to-install-oracle-java-8-on-ubuntu-14-04-lts/

2. http://cassandra.apache.org/download/
3. sudo service cassandra start
4. call cqlsh
```

   Note: If you are running into an error, check the swap space as shown below and allocate a swap space.

```
grep swap /var/log/cassandra/system.log
WARN  [main] 2017-03-30 05:11:30,694 SigarLibrary.java:174 - Cassandra server running in degraded mode. Is swap disabled? : true,  Address space adequat

sudo fallocate -l 4G /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
sudo swapon -s
sudo vi /etc/fstab
/swapfile   none    swap    sw    0   0
sudo service cassandra start
```

2. Run cqlsh and get a screenshot as shown below: <--- This screenshot will give you 10 points. (all or none)

```
Last login: Sat Apr  1 17:21:39 2017 from 10.0.2.2
vagrant@vagrant-ubuntu-trusty-64:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.10 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh>
```

Part II - 90 points

- In the part of the assignment, we will exercise creating a table, populating the table through data wrangling , and manipulate data using CQL DML commands.

---

1. Create the following Cassandra table with a primary key consisting of state, city and zip. The state serves as a partition key and the city and zip together serve as a clustering key.

```
create table citylist
(
  city varchar,
  loc List,
  pop int,
  state varchar,
  zip varchar,
  primary key (state,city,zip)
) ;
```

2. Get a JSON data from http://media.mongodb.org/zips.json. You may use the wget command.
3. Populate the table with this data. Notice that the format of the given JSON data does not follow the given schema. Therefore, you need to figure out how to wrangle the given data to populate the given table. This process may involves converting JSON data to CSV and cleaning and reorganizing them using regular expressions. It is a good practice to work out your logic with a small datasets before moving to large data set. Briefly describe how you wrangle the data.
4. Get the screenshot of the first 15 of the table content.
5. Write a CQL command that gets all rows where city is 'NEW WORK'. Since the city is not the partition key, your query will not be executed without one of the following methods:
   - Method 1: ALLOW FILTERING
   - Method 2: With a secondary index built on the city column

- Method 3: With a materialized view of which partition key is city (You need to define a materialized view and select * from it.)
    1. For each method, show the first 15 results and show the execution time of the query. (It is your job to find the way to measure the execution time of a CQL query.)
    2. Analyze the execution time and describe your rationale behind of your analysis.
6. Write a CQL command that gets **all the zip codes of the NEW YORK city** with a population greater than **20,000** and less than **30,000**. Answer which method(s) of the previous question work(s) for this query? Also, show the first 15 results of the query.
7. Define another schema for citylist2 table in a way that data can be queried in the **ascending order of city and descending order of zip code**
8. Populate citylist2 with the given JSON data and get the table content.
9. Get all the zip codes of the state California. Show the first 15 results.
10. Get all the zip codes of the city San Jose of which population is greater than 5,000 and less than 10,000. Show the first 15 results.

Your report should be organized as shown below.

```
Part I

2.  cqlsh prompt screenshot

Part II
3. Describe your data wrangling method.
4. Screenshot of results
5.

Method 1:
 A CQL query to get the result
 Result
 Execution Time
Method 2:
 CQL command to create an index and a CQL query to get the result.
 Result
 Execution Time
Method 2:
 CQL command to create a materialized view and a CQL query to get the result
 Result
 Execution Time

Your analysis goes here.

6. Which method(s) work for this question?
   CQL query to get the result
   Result
7. Schema declaration
8. CQL query to get the result
   Result
9. CQL query to get the result
   Result
```

Write your report in yourlast4_digits_SID.pdf and zip it into hw4.zip. Submit hw4.zip through the course web site.