CSP554—Big Data Technologies

Assignment #4

Worth: 18 points

For this assignment you will be using your Hadoop environments

The general theme of this week's assignment is to write several Hive command and queries programs to perform various tasks.

I have included code to demo Hive as one of the files—hql.zip—associated with this assignment. There are useful bits to study and reuse, so have a look now.

You must test your exercises against randomly generated data files. You will generate these files using the TestDataGen program. To do so please follow the below steps:

- Download the file TestDataGen.class from the Blackboard. It is one of the attachments to the assignment.
- scp the file over to the home directory (/home/hadoop) on your Hadoop VM
- Log on to your VM using ssh and execute the file using "java TestDataGen"
- This will output a magic number which you should copy down and provide with the results of your assignment.
- It will also place the files foodratings<magic number>.txt and foodplaces<magic number>.txt in your VM home directory
- Use them for your exercises
- Note, each time you execute the TestDataGen program it create new files of test data so you can exercise your program using different combinations of data. Make sure to send the magic number of your final test data set. Also, this means that every student will have different data for their assignment.

The foodratings<magic number>.txt file has six comma separated fields. The first field is the name of a food critic. The second through fifth fields are the ratings each critic gives to four food types at each restaurant they review. The ratings are an integer from 1 through 50. The sixth field is the id of the restaurant.

The foodplaces<.magic number>.txt file has two comma separated fields. The first field is the id of a restaurant. The second field is the name of that restaurant.

Exercise 1) 2 points

Create a Hive database called "MyDb".

Note, after you do this the default database is still 'default." So unless you do something specific about this, if you create a table without qualifying it as belonging to MyDb (MyDb.sometable), it is created in the 'default' database. You can change the default database via a hive command. Try to discover which one and execute it now. Or when you create and use a table you must always qualify its name with the name of the database you created.

Now in MyDb create a table with name foodratings having six columns with the name of the first 'name' and the type of the first a string and the names of the remaining columns food1, food2, food3, food4 and id and indicate their types each as an integer. The table should have storage format TEXTFILE and column separator a ",". That is the underlying format should be a CSV file. The table itself and each column should include a comment just to show me you know how to use comments (it does not matter what it says).

Execute a Hive command of 'DESCRIBE FORMATTED MyDb.foodratings;' and capture its output as one of the results of this exercise.

Then in MyDb create a table with name foodplaces having two columns with first called 'id' with the type of the first an integer, and the second column called 'place' with the type of the second a string. This table should also have storage format TEXTFILE and column separator a ",". That is the underlying format should be a CSV file. No comments are needed for this table.

Execute a Hive command of 'DESCRIBE FORMATTED MyDb.foodplaces' and capture its output as another of the results of this exercise.

Exercise 2) 2 points

Load the foodratings<magic number>.txt file created using TestDataGen from your local file system into the foodratings table.

Execute a hive command to output the min, max and average of the values of the food3 column of the foodratings table. This should be one hive command, not three separate ones.

A copy of the hive command you wrote, the output of this query and the magic number are the result of this exercise.

Exercise 3) 2 points

Execute a hive command to output the min, max and average of the values of the food1 column grouped by the first column 'name'. This should be one hive command, not three separate ones.

The output should look something like:

Mel 10 20 15

Bill 20, 30, 24

…

A copy of the hive command you wrote, the output of this query and the magic number are the result of this exercise.

Exercise 4) 2 points

In MyDb create a partitioned table called ‘foodratingspart’

The partition field should be called ‘name’ and its type should be a string. The names of the non-partition columns should be food1, food2, food3, food4 and id and their types each an integer. The table should have storage format TEXTFILE and column separator a “,”. That is the underlying format should be a CSV file. No comments are needed for this table.

Execute a Hive command of ‘DESCRIBE FORMATTED MyDb.foodratingspart;’ and capture its output as the result of this exercise.

Exercise 5) 2 points

Assume that the number of food critics is relatively small, say less than 10 and the number places to eat is very large, say more than 10,000. In a few short sentences explain why using the (critic) name is a good choice for a partition field while using the place id is not.

Exercise 6) 2 points

Configure Hive to allow dynamic partition creation. Now, use a hive command to copy from MyDB.foodratings into MyDB.foodratingspart to create a partitioned table from a non-partitioned one.

Hint: The ‘name’ column from MyDB.foodratings should be mentioned last in this command (whatever it is).

Provide a copy of the command you use to load the ‘foodratingspart’ table as a result of this exercise.

Execute a hive command to output the min, max and average of the values of the food2 column of MyDB.foodratingspart where the food critic ‘name’ is either Mel or Jill.

The query and the output of this query are other results of this exercise. It should look something like

10 20 15

Exercise 7) 2 points

Load the foodplaces<.magic number>.txt file created using TestDataGen from your local file system into the foodplaces table.

Use a join operation between the two tables (foodratings and foodplaces) to provide the average rating for field food4 for the restaurant ‘Soup Bowl’

The output of this query is the result of this exercise. It should look something like

Soup Bowl 20

Exercise 8) 4 points

Read the article “An Introduction to Big Data Formats” found on the blackboard in section “Articles” and provide short (2 to 4 sentence) answers to the following questions:

a) When is the most important consideration when choosing a row format and when a column format for your big data file?

b) What is “splittability” for a column file format and why is it important when processing large volumes of data?

c) What can files stored in column format achieve better compression than those stored in row format?

d) Under what circumstances would it be the best choice to use the “Parquet” column file format?