CSP554—Big Data Technologies

Prachi Kotadia (A20549927)

## Assignment #4

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Exercise 1) 2 points**

Create a Hive database called “MyDb”.

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



****

****

**A screenshot of a computer program

Description automatically generated**

****

A screenshot of a computer program

Description automatically generated

A screenshot of a computer screen

Description automatically generated

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

A screenshot of a computer program

Description automatically generated

****

**A screenshot of a computer program

Description automatically generated**

**----------------------------------------------------------------------------------------------------------------------------------------------**

**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 black screen with white text

Description automatically generated**

**----------------------------------------------------------------------------------------------------------------------------------------------**

**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.

****

**A screen shot of a black background

Description automatically generated**

**----------------------------------------------------------------------------------------------------------------------------------------------**

**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.

**A screenshot of a computer program

Description automatically generated**

****

A screenshot of a computer screen

Description automatically generated

----------------------------------------------------------------------------------------------------------------------------------------------

**Exercise 5) 2 points**

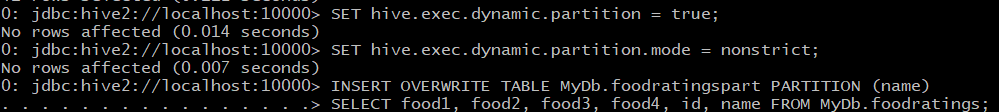
**Assume that the number of food critics is relatively small, say less than 10 and the number of places to eat is very large, say more than 10,000. In a few short sentences explain why using the (critic) name is a good choice for a partition field while using the place id is not.**

1. It is better to utilize the critic's name as a partition field when there are few reviewers because it reduces the number of partitions, which is useful for managing and retrieving data. However, because there are so many locations, employing the place ID would result in a lot of partitions, which could cause performance problems because it can be difficult to manage and query such a large number of partitions, which would slow down query processing.

**----------------------------------------------------------------------------------------------------------------------------------------------**

**Exercise 6) 2 points**

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.





A screen shot of a computer

Description automatically generated

----------------------------------------------------------------------------------------------------------------------------------------------

**Exercise 7) 2 points**

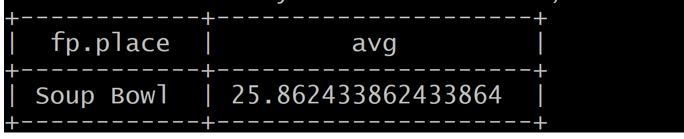
Load the foodplaces<.magic number>.txt file created using TestDataGen from your local file system into the foodplaces table.

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

****

**A screenshot of a computer program

Description automatically generated**

****

**----------------------------------------------------------------------------------------------------------------------------------------------**

**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) The choice between a row format and a column format for big data files depends on the nature of the queries to be performed. Column-based storage is most beneficial for analytics queries that need only a subset of columns over large datasets, while row-based storage is better suited for queries requiring access to most or all columns of each row.

b) "Splittability" refers to the ability to break down a file into smaller, independent chunks that can be processed in parallel, which is crucial for efficient processing of large volumes of data. It enhances parallel processing capabilities, significantly impacting performance.

c) Files stored in a column format can achieve better compression than those in a row format because storing values of the same type together allows for more efficient compression algorithms. This results in significant savings in storage space and improves query performance due to reduced I/O.

d) The Parquet column file format is best used in scenarios where the dataset is wide (i.e., has many columns) and read-heavy workloads are common. It is particularly suited for use cases involving analytics and querying large datasets, where its efficient compression and splittability significantly improve performance and reduce storage costs.

**----------------------------------------------------------------------------------------------------------------------------------------------**

**A screenshot of a computer

Description automatically generated**