**ICS 411: Introduction to Big Data Management**

**Homework Assignment 4**

Due: See Syllabus

**Data Analysis using Hive and Impala (there are two questions)**

**Question 1: Weather Data Set:**

In this question, you will use weather data from a select number of research stations throughout the continent of Antarctica. The data files are in a zipped file called question1Data.zip.

1. Create a database in Hive called weatherDatabase
2. Create a table called locations (location is a reserved word in Hive) that has the following attributes and load the table from the file location.csv:
   1. A station column, of type string
   2. A latitude column, of type integer
   3. A longitude column, of type integer
3. Create a partitioned table called windspeed that is partitioned by station name. The tablehas the following attributes. Load data into the partitioned table from the given **8** station files, called **wind\_<station\_name>.csv**.
   1. A year column, of type integer
   2. A month column, of type string
   3. A knots column, of type float
4. Create an external table called temperature that has the following attributes:
   1. A station column, of type string
   2. A year column, of type integer
   3. A month column, of type string
   4. A celsius column, of type float

This table holds the average monthly temperatures, in degrees Celsius, from eight different weather stations in Antarctica from several decades. Load the data into this table from the file temperature.csv.

1. Create a partitioned table called partitioned\_temperature that is partitioned on the station name. Fill in the partitioned table from with data from temperature table.
2. Run each of the following queries twice, once on Hive and once on Impala and record the running time in each case.
   1. SELECT \*

FROM locations;

Hive .03s

Impala 0.13s

* 1. SELECT count(\*)

FROM locations;

Hive 19.4777s

Impala 0.17s

* 1. SELECT station, count(\*)

FROM windspeed

GROUP BY station;

Hive 18.78s

Impala 5.71s

* 1. SELECT station,avg(knots) as avgknots

FROM windspeed

GROUP BY station

ORDER BY avgknots;

Hive 37.859s

Impala 0.29s

* 1. SELECT \*

FROM windspeed

LIMIT 20;

Hive 0.048s

Impala 0.16s

* 1. SELECT \*

FROM temperature

WHERE year = 2000;

Hive 0.039s

Impala 3.92s

* 1. SELECT \*

FROM partitioned\_temperature

WHERE year = 2000;

Hive .04s

Impala 0.15s

* 1. SELECT T.station, T.year, T.month, W.knots, T.Celieus, W.knots

FROM temperature T join windspeed W

ON (T.station = W.station and T.year = W.year and T.month = W.month)

Hive 18.466s

Impala 1.29s

* 1. SELECT T.station, T.year, T.month, W.knots, T.celsius, W.knots

FROM partitioned\_temperature T join windspeed W

ON T.station = W.station and T.year = W.year and T.month = W.month;

Hive 18.329s

Impala 1.2s

1. Write Hive SQL queries to answer the following:
   1. Count how many rows are there in the **temperature** table which do not have **NULL** values for the **Celsius** column.

SELECT station, count(Celsius)

FROM temperature

WHERE Celsius is not null

GROUP BY station;

Hive 20.502s

Impala 0.26s

* 1. Find out what the average temperature in Antarctica was in 1970.

SELECT AVG(celsius) AS avgTemp

FROM partitioned\_temperature

WHERE year = 1970;

Hive 20.355s

Impala 0.14s

* 1. Find out what the hottest and coldest temperatures were recorded in Antarctica.

SELECT MIN(celsius) AS minTemp, MAX(celsius) AS maxTemp

FROM partitioned\_temperature;

Hive 20.282s

Impala 0.14s

* 1. Knots is a measure of speed. To convert knots to Kilometer Per Hour (KPH), multiply by 1.852. Write an SQL query to display the station, speed in knots, and equivalent speed in KPH.

SELECT station, knots, knots\*1.852 AS KPH

FROM windspeed;

Hive 0.089s

Impala 0.64s

1. The name of the weather station at the South Pole is called Clean Air, because very little man-made pollution can be found there. Write an SQL query to find out what temperatures were recorded at the South Pole such that the output is sorted by temperatures. When is it Summer and when it is winter in South pole?

SELECT month, celsius

FROM partitioned\_temperature

WHERE station = 'Clean\_Air' AND celsius IS NOT NULL

ORDER BY celsius ASC;

1. Assume we need to combine all information in one table. Create a table called weatherTable that include station, temperature in Celsius and Fahrenheit, wind speed in knots and KPH, latitude, and longitude. Use the weatherTable you to answer the following questions:
   * 1. What is the average wind speed at 90 degrees latitude (the South Pole)?
     2. What were the coldest and warmest temperatures recorded for each station?

**What to submit:** one document that includes the SQL queries that you wrote to create the database and tables. For question 6, include a table with three columns as follows to report the running time for the queries.

|  |  |  |
| --- | --- | --- |
| Query | Time on Hive | Time on Impala |

**Question 2: Dualcore Company Data Set:**

In this question you will be working on a company database that includes 6 tables, namely, employees, customers, orders, suppliers, products, and order\_details. You are given an SQL script to create the database and populate the tables with data in MySQL. You are asked to complete the following steps:

1. Run the dualcoreDatabase.sql script to create the database in MySQL.
2. Write Sqoop commands to import all tables from MySQL to Hive such that the orders table in Hive should be partitioned by year of order’s date and the customers table is clustered into 10 clusters based on customer id. You will need to set up the Hive partitioned Orders and the Customers bucket tables first. Load the data via Sqoop into temp tables, then insert-select into the final Hive Orders and Customers tables.
3. Show the contents of the following HDFS directory (include contents of all sub folders):

/user/hive/warehouse/dualcoreDatabase.db

hadoop fs -ls /user/hive/warehouse/dualcore.db

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 05:37 /user/hive/warehouse/dualcore.db/customers**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 04:10 /user/hive/warehouse/dualcore.db/employees**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 04:13 /user/hive/warehouse/dualcore.db/order\_details**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 08:42 /user/hive/warehouse/dualcore.db/orders**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 04:08 /user/hive/warehouse/dualcore.db/products**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 16:37 /user/hive/warehouse/dualcore.db/ratings**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 16:18 /user/hive/warehouse/dualcore.db/ratings\_2012**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 16:17 /user/hive/warehouse/dualcore.db/ratings\_2013**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 04:07 /user/hive/warehouse/dualcore.db/suppliers**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 05:27 /user/hive/warehouse/dualcore.db/temp\_customers**

**drwxrwxrwx - cloudera supergroup 0 2020-11-14 08:23 /user/hive/warehouse/dualcore.db/temp\_orders**

1. For each of the following questions, write an SQL query to find the required output. Run your query three time: in MySQL , Hive, and Impala and record the time taken by each:
   1. Find the three most common jobs (i.e., the top three jobs in terms of the number of employees).
   2. Find the number of orders in each month
   3. Find the average order size (i.e., average number of products in each order).
   4. Find the top 2 states with the maximum number of customers.
   5. Find the top 2 state with the maximum number of orders.
   6. Find the number of customers in each zip code.
   7. Find the total number of orders in each zip code.
   8. Find the number of orders for each product.
   9. For each customer, find the total number of orders made by that customer.
2. You are given two additional files that includes customer ratings for various products (file names are: ratings\_2012.txt and ratings\_2013,txt). Create a partitioned table in Hive to include the ratings data. The table should include two partitions, one for 2012 and the other one for 2013.
3. In this question, you will run a sequence of queries to infer conclusions from customers’ ratings. Use the ratings table you created in step 5 to answer the following questions:
   1. What are the 5 products that got the maximum number of ratings?
   2. What are the 5 products with the maximum average rating?
   3. What are the two products with the minimum average rating?
   4. Display product id and users’ comments for all ratings that include the substring ‘ten times more’ for product with id 1274673
   5. Find all distinct comments containing the word “red” that are associated with product with id 1274673.
   6. Write a query that will display the record for product ID 1274673 in the products table.
   7. The previous query should have shown that the product was a “16GB USB Flash Drive (Red)” from the “Orion” brand. Write another query identify similar products (i.e., name includes ‘16GB USB Flash Drive’ and from the same brand ‘Orion’).

**The conclusion is that analyzing users’ ratings of product helped Dualcore uncover a pricing error.**

**What to submit:** a document that includes Sqoop and SQL commands to answer the questions. For question 3, include a table of four columns as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| SQL | Time in MySQL | Time in Hive | Time in Impala |