CS 5620 – Take-Home Final Exam

First Name: ....KISHORE....KUMAR.....

Last Name: .....ANDEKAR.....

Student 700 Number: .....700744713.....

Instructions:

1. PLEASE ANSWER ALL QUESTIONS.
2. PLEASE WRITE THE QUESTION NUMBER AND ITEM NEXT TO EACH OF YOUR ANSWERS (Examples: Question 2., Question 3.c., Question 4.g. etc.)
3. WRITE YOUR ANSWERS ON CLEAN SHEETS.
4. WRITE BRIEFLY AND NEATLY.
5. PLEASE SCAN YOUR ANSWER SHEET THEN UPLOAD IT INTO BLACKBOARD. YOU ARE ONLY ALLOWED ONE ATTEMPT. NO EMAIL SUBMISSION WILL BE ACCEPTED
6. PLEASE WRITE YOUR FIRST NAME, LAST NAME, AND NUMBER.
7. PLEASE TURN IN YOUR GENUINE ANSWERS. DON'T SHARE YOUR ANSWERS WITH ANYONE. ANY VIOLATION FOUND WILL RESULT IN FAILING THE COURSE AND BEING REPORTED TO THE DEPARTMENT.
8. IF YOU HAVE NO ACCESS TO A SCANNER, YOU CAN USE A PHONE APP TO SCAN YOUR ANSWER SHEETS. ALL OF YOUR ANSWERS MUST BE SUBMITTED IN ONE PDF FILE ON BLACKBOARD.
9. GOOD LUCK!

Q.1. What are the three configuration modes for running Hive cli service with respect to the metastore service and the metastore database? Briefly, state the difference between them.

Q.2. **True** or **False**: In order to run Hive queries in a Hadoop cluster, Hive must be installed on every node in the cluster. Justify your answer.

Q.3. Write the necessary commands in Hive to find the total length of all lines of a file that exists in HDFS. Assume that the filename is `transactions.txt` and it is stored under the user's home directory. The function `length(<input-str-argument>)` can be used to find the length of strings in Hive. You need to create and populate one or more tables to solve the problem, the fewer the number of tables the better is your answer. You should be able to figure out the commands without actually using Hive console.

Q1. Hive Metastore is the central repository of Apache Hive MetaData.
There are three Configuration Modes for running Hive cli services are
1. Embedded Metastore 2. Local Metastore 3. Remote Metastore

1. Embedded Metastore:

In Hive by default, Both Metastore service and Hive service runs in the same JVM by using Embedded Derby Database. This Mode also has a limitation that, as only one embedded Derby database can access the database files at any one time. If we try to start the second session it produces an error when it attempts to open a connection to the Metastore.

2. Local Metastore:

To overcome this Limitation of Embedded Metastore, for Local Metastore was introduced. This mode allows us to have many Hive sessions i.e., many users can use the metastore at the same time. This can be achieved by using JDBC databases like MySQL which runs in a separate JVM than that of Hive and Metastore services which are running in the same JVM.

3. Remote Metastore:

In Remote Mode, metastore runs on its own separate JVM, not in the Hive service JVM. If other processes want to communicate with the Metastore server they can communicate using Thrift Network API's.

Q2. False, Hive is an Hadoop client and it runs on the top of the Hadoop, so we don't need Hive to be installed in all node of Hadoop cluster.

Q3. Given File "transactions.txt" is stored under home directory. We will populate the data into a table called count.

> Create table count(line STRING) location '/user/hive/warehouse/transactions.txt';

Now table count will have all contents of transactions file. we will split each line using line delimiter '\n'

split(line,'\n') // It will split lines into array of strings

> select split(length(line),'\n') from count;

The above command will display length of characters of each lines.

To find total length of all lines we use sum

> select split(sum(length(line),'\n')) from count;

Q4. 4a. default> show databases;

4b. default> show tables;

4c. default> show tables in companydb;

4d. default> select * from companydb.employees;

4e. default> select * from companydb.products limit 5;

4f. default> SET hive.execution.engine

4g. default> SET hive.metastore.warehouse.dir

4h. default> hdfs dfs -cat /user

4i. default> pwd

4j. default> LOAD DATA LOCAL INPATH 'foo.txt' INTO TABLE mytable;

4K. default> describe extended mytable

or

default> describe formatted mytable.

Q5. Explode Function is used to split output in individual tokens rather than a List

explode (split ('Welcome to Programming Hive !', ' '))

W
e
l
c
o
m
e
t
o
P
r
o
g
r
a
m
m
i
n
g
H
i
v
e
!

Q6. Schema on Write: In traditional Database, database has control over storage. Here data is being created against schema when written into the database during Load.

Schema on Read: Hive has no control over storage and database is not verified during load time, rather it is verified while processing the Query. Hive uses this process called Schema on Read.

→ Schema on Read is more efficient than schema on write. So Hive uses schema on Read.

Q7. External Keyword tells Hive this table is external and Location clause is required to tell Hive where it is Located.

For External, Hive doesn't assume its own the data. Therefore, dropping the table doesnot delete the data, although the metadata for the table will be deleted.

```
Create external table if not exists mytable(
        table_id INT,
        table_name STRING)
        ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
        LOCATION '/data/dataset-2020';
```

Q8. 8a: Create table Customers (cust_id INT, cust_name STRING, street STRING, city STRING, zip INT, region STRING)
Partitioned By (country STRING);

8b. Create table customers_data (cust_id INT, cust_name STRING, street STRING, city string, zip INT, region STRING)
Partitioned By (Country STRING);

```
Load Data Local Inpath '/data/customers/usa' into table customers_data;
    Partition ( country = 'USA');
```

Load Data Local Inpath '/data/customers/canada' into table customers_data;
Partition (country = 'canada');

Load Data Local Inpath = '/data/customers/Mexico' into table customers_data;
Partition (Country = 'Mexico');

set hive.exec.dynamic.partition = true;
set hive.exec.dynamic-partition.mode = nonstrict
Insert into table customers_data Partition (country);
Select cust_id, cust_name, street, city, zip, region, country from customers_data;

8.c. Yes, we can use dynamic partitioning to avoid creating id's of partitions as we having many countries i.e., USA, Canada and Mexico. Also, we have loaded data using single SQL query using dynamic partition.

8.d. Based on filter condition using where clause in partition table, we need to scan thousands of records. Partitions can dramatically improve Query Performance.

Q9. lines = sc.textFile('data/logfiles/*')
lines.count()

Q10. python_lines = lines.filter(lambda line: 'error' in line.lower())
python_lines.count()

Q11. character_count = lines.map(lambda : line : len(line))
character_count.sum()

Q12. numbers = sc.parallelize(list(range(0,100)))
numbers.sum()

Q13. Lazy Evaluation in Spark means whenever we call a transformation on RDD operation is not performed immediately. It doesnot allocate memory for any data sets until they are computed. Spark uses Lazy Evaluation to reduce no of parse takes to takeover data by grouping operations together. Lazy evaluation applies only to actions not to transformations.

example: Actions logs.count()

Transformation logs = sc.textFile("file1.txt")