Discussion IST769 Unit C – Hadoop

## Agenda

1. Your Questions
2. Go over Problem Set
3. Unit Coursework Activities

FRONT MATTER

Start your Hadoop Engines!

**PS:>** **docker-compose up -d** **jupyter namenode datanode hive-server hive-metastore hive-metastore-postgresql**

Other things we will need (later)

1. To access the Linux shell where the Hadoop and Beeline clients are installed, we must connect to the running instance of the hive-server service. Type the following at the PowerShell Prompt:  
**PS:> docker-compose exec hive-server bash**

2. To see a list of Hadoop commands, from the Linux shell, type:  
**root@hive-server: # hadoop fs**

3. To connect the Hive beeline client from the Linux shell, type:  
**root@hive-server: # beeline -u jdbc:hive2://hive-server:10000/default**

4. To access HFDS over the web from the windows host:  
[http://localhost:50070](http://localhost:50070/)

5. To access Jupyter Lab from your Windows host  
[http://localhost:8888](http://localhost:8888/)  
The password is SU2orange!

ON WINDOWS port conflict with 50070:

Open power as administrator

**net stop winnat**

DEMO IN CLASS

beeline> !sh hadoop fs -mkdir /user/root

beeline> !sh hadoop fs -mkdir /user/root/grades

CTRL+D Exit beeline

# hadoop fs -put /datasets/grades/\*.tsv /user/root/grades/

MAKES EXTERNAL TABLE:

beeline>

create external table grades\_ext (

year int,

term string,

course string,

credits int,

letter string

) row format delimited

fields terminated by '\t'

location '/user/root/grades';

beeline > select \* from grades\_ext;

>> moving stocks data for demo:

hadoop fs -put /datasets/stocks/stocks.csv /user/root/grades/

>> removing stocks.csv

hadoop fs -rm /user/root/grades/stocks.csv

Internal table

create table grades\_int stored as parquet as

select \* from grades\_ext;

OPTIONAL

Microsoft Hive ODBC driver: (64 bit)  
<https://www.microsoft.com/en-us/download/details.aspx?id=40886>

Power BI Desktop  
<https://powerbi.microsoft.com/en-us/downloads/>

## 1. Your Questions

1. Could you please Internal-External Tables?
2. Movement of source data when loaded from local file system into Hive-managed table.
3. Location Hive Internal/External table at the HDFS path /user/hive/warehouse
4. Which hive table requires use of the Load Command (Hive Internal/External)
5. True or False,
   1. No data are stored within the hive metastore
   2. The hive client is responsible for parsing the query’s execution plan.
   3. On hadoop, the hive data is always stored in HDFS

## DEMO

PS>  **docker-compose exec hive-server bash**

root@hive-server:~# hadoop fs -mkdir /user/root

root@hive-server:~# hadoop fs -mkdir /user/root/grades

root@hive-server:~# hadoop fs -put /datasets/grades/\*.tsv /user/root/grades

root@hive-server:~# hadoop fs -ls /user/root/grades

**root@hive-server:~#**

jdbc:hive2://hive-server:10000/default> create external table grades (

year int,

semester string,

course string,

credits int,

letter string

) row format delimited fields terminated by '\t' location '/user/root/grades';

jdbc:hive2://hive-server:10000/default> select \* from grades;

jdbc:hive2://hive-server:10000/default> create table grades\_internal stored as parquet as select \* from grades;

jdbc:hive2://hive-server:10000/default> show tables;

jdbc:hive2://hive-server:10000/default> select \* from grades\_internal;

jdbc:hive2://hive-server:10000/default> !sh hadoop fs -ls /user/hive/warehouse/

<http://localhost:50070/>

Connecting to Hive from Power BI. NOTE: The default Hive connector requires SSL, ours is not setup like that so we must use ODBC (Open Database Connectivity)

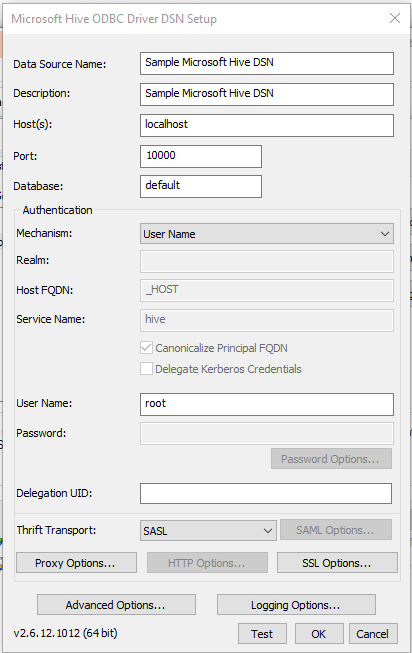
**Control Panel ⇒ Administrative Tools ⇒ ODBC 64 Bit ⇒ System DSN**

Host: localhost

Port: 10000

Database: default

User name: root



## 2. Coursework Activities - Hadoop

1. Create a folder in hadoop called **exams** then load /datasets/exam-scores/\*.csv into that folder on HDFS.
2. Create an external Hive table called examscores. The challenge here is figuring out how to read the files with headers!
3. Write a hive SQL query to get the average exam score by exam version
4. Create a folder in hadoop called **stocks** then load /datasets/json-samples/stocks.json into that folder on HDFS.
5. Create an external Hive table called stocks. The challenge here is figuring out how to read the JSON format!
6. Write a Hive SQL Query to output stocks with a price > 100

## 