```
-------------------------------------------------------------------------------
------------------------------
-------------------------------------------------------------------------------
------------------------------
-- Programming Hive 1 - Lab Exercises

-- 1. In this lab session, we will start working with HiveQL
-- 2. File full_text.txt is available under D2L -> Datasets and Scripts -> Twitter;
Use Filiezilla to copy file onto the virtual machine
-- 3. File dayofweek.txt is available under D2L -> Datasets and Scripts -> Twitter;
Use Filiezilla to copy file onto the virtual machine
-- 4. To avoid confusion, please always include database name 'twitter.' as part of
your hive table name. If you
--    don't specify the database name while you're not in the twitter database (use
twitter), you will not find the
--    the corresponding table. By default you're in a database called "default"
--    e.g.,  twitter.full_text
-- 5. Lines that begin with "--" are comments
-- 6. Lines that begin with "##" are Hive or Hadoop commands.  Commands could span
multiple lines. Only the beginning is marked with "##".
-------------------------------------------------------------------------------
------------------------------
-------------------------------------------------------------------------------
------------------------------
-- Copy files into HDFS. We will be loading these files into hive tables and
performing queries.
----------------------------
----------------------------
--Copy full_text.txt from local (virtual box) to a HDFS location
/user/root/twitter. You need the /user/root/twitter directory first.
## hadoop fs -mkdir /user/root/twitter
## hadoop fs -put full_text /user/root/twitter


--Copy dayofweek.txt from local (virtual box) to a HDFS location /user/root/twitter
## hadoop fs -put dayofweek.txt /user/root/twitter
-------------------------------------
-------------------------------------
-- Working with Hive Database
-------------------------------------
-------------------------------------

-- list available databases
## show databases;

-- create a database for the twitter related analysis
## create database twitter;
```

```
-- change to the twitter database
## use twitter;

-- check if twitter database has been listed
## show databases;

-- show database details
## describe database extended twitter;


---------------------------------------
---------------------------------------
-- Working with Hive Tables
---------------------------------------
---------------------------------------

-- create an empty full_text hive table
## create table twitter.full_text (

          id string,
          ts string,
          lat_lon string,
          lat string,
          lon string,
          tweet string)
row format delimited
fields terminated by '\t' ;

-- load data into the twitter.full_text table
## load data inpath '/user/root/twitter/full_text.txt'
     overwrite into table twitter.full_text;


-- show table schema
## describe twitter.full_text;

-- show extended table detail
## describe extended twitter.full_text;

-- use 'dfs -ls' command in hive to list HDFS directory
-- you should see a directory call "twitter.db"
-- hive databases are just HDFS directories
-- each hive table is an HDFS file

## dfs -ls /apps/hive/warehouse;


-- display contents of full_text table
```

```
## select id, ts from twitter.full_text limit 5;

-- create a new table from an existing table
## create table twitter.full_text_2 as
     select *
     from twitter.full_text;



--------------------------------------
--------------------------------------
-- Hive Functions
--------------------------------------
--------------------------------------

-----------------
-- DATE function
-----------------

-- cast string to timestamp

## create table twitter.full_text_ts as
     select id, cast(concat(substr(ts,1,10), ' ', substr(ts,12,8)) as timestamp) as
ts, lat, lon, tweet
     from full_text;

## describe twitter.full_text_ts;

-- Extract year, month and day from timestamp
## select ts, unix_timestamp(ts) as unix_timestamp, to_date(ts) as dt, year(ts) as
year, month(ts) as month, day(ts) as day
     from twitter.full_text_ts
     limit 5;




-------------------
-- STRING function
-------------------
## select id, ts, trim(lower(tweet)) as tweet
     from twitter.full_text_ts
     limit 5;

## select id, ts, trim(upper(tweet)) as tweet
     from twitter.full_text_ts
     limit 5;
```

```
## select id, ts, length(tweet) as tweet
     from twitter.full_text_ts
     limit 5;

## select id, ts, sentences(tweet) as tokens
     from twitter.full_text_ts
     limit 5;



-- Find twitter handles mentioned in a tweet

## select id, ts, regexp_extract(lower(tweet), '@user_[A-Za-z0-9_][A-Za-z0-9_]*',0)
as patterns
     from twitter.full_text_ts
     limit 5;

-- Another Regex to extract @mentions
## select id, ts, regexp_extract(lower(tweet), '(.*)@user_(\\S{8})([:| ])(.*)',2)
as patterns
     from twitter.full_text_ts
     limit 5;

-- Find hashtags mentioned in the tweet
## select id, ts, regexp_extract(lower(tweet),
'#[A-Za-z0-9_]+[A-Za-z][A-Za-z0-9_]*',0)
   from twitter.full_text_ts
     limit 5;

-- Finding top 10 users who tweet long tweets
-- maximum tweet length is 140 characters.. but output of this query shows tweets
with length > 140
## select t.id, t.len, t.tweet
   from (select id, tweet, length(tweet) as len from twitter.full_text_ts) t
   order by len desc
   limit 10;


-- removing the mentions will improve the results
## select t.id, t.len, t.trimmed_tweet
    from (select id, regexp_replace(tweet, "@user_[A-Za-z0-9_][A-Za-z0-9_]*", "")
as trimmed_tweet, length(regexp_replace(tweet, "@USER_\\w{8}", " ")) as len
from                  twitter.full_text_ts) t
order by len desc
limit 10;

-- Yet another way to trim tweets
```

```
## select t.id, t.len, t.trimmed_tweet
    from (select id, regexp_replace(tweet, "@USER_\\w{8}", "") as trimmed_tweet,
length(regexp_replace(tweet, "@USER_\\w{8}", " ")) as len from
twitter.full_text_ts) t
order by len desc
limit 10;

------------------------
-- CONDITIONAL function
------------------------
-- Find users who like to tw-eating
-- not a great example. just for you to practice case when conditional function

## select * from
    (select id, ts, case when hour(ts) = 7 then 'breakfast'
                         when hour(ts) = 12 then 'lunch'
                         when hour(ts) = 19 then 'dinner'
                    end as tw_eating,
           lat, lon
    from twitter.full_text_ts) t
where t.tw_eating in ('breakfast','lunch','dinner')
limit 10;



--------------------------------------
--------------------------------------
-- WHERE Clause - Filtering Data
--------------------------------------
--------------------------------------

-- Find all tweets by a user
-- Hive is very slow for this type of query because for even one record it still
scans through the entire table
-- this is because MapReduce works in a streaming fashion
-- for fast retrieval, you can either use relational database or new technologies
such as Apache Impala (Cloudera)

## select id, ts, lat, lon, tweet
     from twitter.full_text_ts
     where id='USER_ae406f1d';



-- Find 5 tweets on a specific date
## select *
     from twitter.full_text_ts
     where to_date(ts) = '2010-03-07'
```

```
     limit 5;

-- Calculate # of tweets on a specific date
## select count(*)
     from twitter.full_text_ts
      where to_date(ts) = '2010-03-07'

-- Find all tweets tweeted from NYC vicinity (using bounding box -74.2589, 40.4774,
-73.7004, 40.9176)
-- The square bounding box won't give us very accurate results. We may end up
retrieving tweets in New Jersey as well.
-- A better approach is to use geo function plugins for hive. We will re-visit this
when we introduce Pig

## select distinct lat, lon
from twitter.full_text_ts
where lat > 40.4774 and lat < 40.9176 and
      lon > -74.2589 and lon < -73.7004
limit 20;


---------------------------------------
---------------------------------------
-- GROUP BY - Aggregation Functions
---------------------------------------
---------------------------------------

-- Calculate # of tweets per user

## create table twitter.tweets_per_user as
select id, COUNT(*) as cnt
from twitter.full_text_ts
group by id;


---------------------------------------
---------------------------------------
-- ORDER BY
---------------------------------------
---------------------------------------

-- Find top 10 tweeters in NYC
## select id, count(*) as cnt
from twitter.full_text_ts
where lat > 40.4774 and lat < 40.9176 and
      lon > -74.2589 and lon < -73.7004
group by id
order by cnt desc
```

```
limit 15;

---------------------------------------
---------------------------------------
-- DISTINCT
---------------------------------------
---------------------------------------

-- Find # of distinct days this dataset cover

## select count(distinct to_date(ts))
from twitter.full_text_ts;


---------------------------------------
---------------------------------------
-- JOIN
---------------------------------------
---------------------------------------

-- prepare lookup table 'dayofweek'
-- dayofweek lookup file is available at D2L -> Dataset and Scripts -> Twitter

## create table twitter.dayofweek (datev string, dayofweek string)
row format delimited
fields terminated by '\t';

## load data inpath '/user/root/twitter/dayofweek.txt'
overwrite into table twitter.dayofweek;


-- Find Weekend Tweets
-- INNER JOIN

## create table twitter.weekend_tweets as
select a.id, a.ts, b.dayofweek, a.lat, a.lon, a.tweet
from twitter.full_text_ts a JOIN twitter.dayofweek b
     ON to_date(a.ts) = b.datev AND b.dayofweek IN ('Saturday','Sunday');
```