Hive - Impala Workshop - Keys

Part 1 – Hive

HDFS

```
hdfs dfs –mkdir /hivelab1

hdfs dfs –put Stations_2017.csv /hivelab1/
```

Hive

beeline –u jdbc:hive2://localhost:10000

```
create database bixi;

use bixi;

create table Stations (
    code string,
    name string,
    latitude double,
    longitude double)
    Row format delimited fields terminated by ',' stored as TextFile
    TBLPROPERTIES ("skip.header.line.count"="1");

load data inpath '/hivelab1/Stations_2018.csv' into table stations;


select * from stations limit 10;

select count(*) from stations;

drop table stations;


create external table Stations (
    code string,
    name string,
    latitude double,
    longitude double)
    Row format delimited fields terminated by ',' stored as TextFile
    location '/hivelab1/'
    TBLPROPERTIES ("skip.header.line.count"="1");

select * from stations limit 10;
select count(*) from stations;


create external table BixiData (
```

```
    start_date string,
    start_station_code string,
    end_date string,
    end_station_code string,
    duration_sec int,
    is_member tinyint)
    Row format delimited fields terminated by ',' stored as TextFile
    location '/hivelab2/'
    TBLPROPERTIES ("skip.header.line.count"="1");

select count(*) from bixidata;

select duration_sec from bixidata order by duration_sec desc limit
5;


select count(*) from bixidata;

select max(duration_sec) from bixidata;
select count(*) FROM bixidata where duration_sec = 7199;
select duration_sec from bixidata order by duration_sec desc limit
20;

select start_station_code, name, COUNT(start_station_code) FROM
bixidata join stations on (start_station_code = code)
GROUP BY start_station_code, name ORDER BY COUNT(*) desc limit 10;

select end_station_code, name, COUNT(end_station_code) FROM bixidata
join stations on (end_station_code = code)
GROUP BY end_station_code, name ORDER BY COUNT(*) desc limit 10;
```

Exercise 2:

```
HDFS

hdfs dfs –mkdir /hivelab3

hdfs dfs –put real_estate.csv /hivelab3/

Hive
```

beeline –u jdbc:hive2://localhost:10000

```
set hive.enforce.bucketing = true;
set hive.enforce.sorting=true;
set hive.exec.dynamic.partition = true;
set hive.exec.dynamic.partition.mode = nonstrict;

CREATE TABLE RealEstate (
Street string, City string, Zip int, State string, Beds int, Baths
int, Sq__ft int, Type string, Price int)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE
tblproperties("skip.header.line.count"="1");

LOAD DATA INPATH '/hivelab3/real_state.csv' INTO TABLE realestate;

select * from realestate limit 10;
```

```
CREATE TABLE RealEstate_Part (
Street string, Zip int, State string, Beds int, Baths int, Sq__ft
int, Type string, Price int)
partitioned by (City string)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;

insert into realestate_part partition (City)
    select Street, Zip, State, Beds, Baths, Sq__ft, Type, Price,
City from realestate;

select * from realestate where city=='ANTELOPE';

CREATE TABLE RealEstate_Bucket (
Street string, Zip int, State string, Beds int, Baths int, Sq__ft
int, Type string, Price int)
partitioned by (City string) clustered by (Street) SORTED BY (Price)
into 4 buckets
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;

insert OVERWRITE TABLE realestate_bucket partition (City)
    select Street, Zip, State, Beds, Baths, Sq__ft, Type, Price,
City from realestate;

select * from realestate_bucket tablesample (bucket 1 out of 4)
where city='ANTELOPE';
select * from realestate_bucket tablesample (bucket 2 out of 4)
where city='ANTELOPE';
select * from realestate_bucket tablesample (bucket 3 out of 4)
where city='ANTELOPE';
select * from realestate_bucket tablesample (bucket 4 out of 4)
where city='ANTELOPE';
```

Part 3 – Impala

```
HDFS

hdfs dfs –mkdir /impalalab1

hdfs dfs –put Pokemon.csv /impalalab1/


--- hive -----

CREATE TABLE pokemon
(
    Number Int,Name String,Type1 String,Type2 String,
    Total Int,HP Int,Attack Int,Defense Int,Sp_Atk Int,
    Sp_Def Int,Speed Int
) row format delimited fields terminated BY ',' lines terminated BY
'\n' tblproperties("skip.header.line.count"="1");


load data inpath '/impalalab1/Pokemon.csv' INTO table pokemon;
```

```
--- impala -----
Invalidate metadata;

HP AVG     69.25875
Select avg(hp) from pokemon;


2-
create table pokemon1 as select *, IF(HP>69.25875, 'powerful',
IF(HP<69.25875, 'Moderate','powerless')) AS power_rate from pokemon;

3-
select COUNT(name),power_rate from pokemon1 group by power_rate;

4-
select name,hp from pokemon1 order by hp desc limit 10;

5-
select name,attack from pokemon1 order by attack desc limit 10;

6-
select name,defense from pokemon1 order by defense desc limit 10;

7-
select name,total from pokemon1 order by total desc limit 10;

8-
select name,(attack-sp_atk) as atk_diff from pokemon1 order by
atk_diff limit 10;

9-
select name,(defense - sp_def) as def_diff from pokemon1 order by
def_diff limit 10;

10-
Select name, speed from pokemon order by speed desc limit 10;
```