Assignment 3 – DS8003

Amir Ghaderi – 500794236

October 27th, 2016

**Question 1:** create table movies\_whole(movieid int, movie\_name string, genre string) row format delimited fields terminated by '\t';

**Question 2:**load data inpath '/user/amir/hive2/action\_comedy\_thriller' overwrite into table movies\_whole;

**Question 3:** create table movies\_part(movieid string, movie\_name string) partitioned by (genre string);

**Question 4:**

load data local inpath 'action' into table movies\_part partition(genre='action');

load data local inpath 'comedy' into table movies\_part partition(genre='comedy');

load data local inpath 'thriller' into table movies\_part partition(genre='thriller');

**Question 5:**

**a) HiveQL:** describe movies\_part;

**Result:**

movieid string

movie\_name string

genre string

# Partition Information

# col\_name data\_type comment

genre string

**b) HiveQL:** show partitions movies\_part;

**Result:**

genre=action

genre=comedy

genre=thriller

**Question 6:** dfs -ls /apps/hive/warehouse/movies\_part;

When a partitioned table is created each partition is placed into it’s own directory inside HDFS.

**Question 7:**

|  |  |  |
| --- | --- | --- |
| **Question** | **Movies\_Part(time(sec))** | **Movies\_Whole(time(sec))** |
| **a)** | **9.183** | **1.923** |
| **b)** | **23.348** | **13.781** |
| **c)** | **17.919** | **9.492** |
| **d)** | **20.083** | **10.410** |
| **e)** | **8.789** | **11.761** |

**7.1 : I think logically they should run faster on the partitioned table.**

**7.2 : They clearly run faster on the Movies\_whole table. I am assuming this is the case because the amount of data we are using is not large enough to benefit from partitions.**

**Question 8:** create table movie\_year\_temp as select movieid, regexp\_extract(movie\_name, '([1-2][0-9][0-9][0-9])',1) as movie\_year, substr(movie\_name,1,length(movie\_name) -6) as movie\_title from movies\_whole;

**Question 9:** create table year\_buckets(movieid int, movie\_year string, movie\_title string) clustered by (movie\_year) into 8 buckets;

**Question 10:** insert into table year\_buckets select movieid, movie\_year, movie\_title from movie\_year\_temp;

**Question 11:** dfs -ls /apps/hive/warehouse/year\_buckets;

When a bucketed table is created each bucket is placed into it’s own file inside HDFS.

**Question 12:** select explode(histogram\_numeric(int(movie\_year),5)) as hist\_year from movie\_year\_temp;

**Results:**

{"x":1600.0,"y":1.0}

{"x":1938.8076923076924,"y":26.0}

{"x":1956.8437499999993,"y":32.0}

{"x":1977.8955223880596,"y":67.0}

{"x":1994.8313413014603,"y":753.0}