ASSIGNMENT 27.6

Using the below given two datasets you need to give a demo on the below joins in hive.

 Bucket Map join

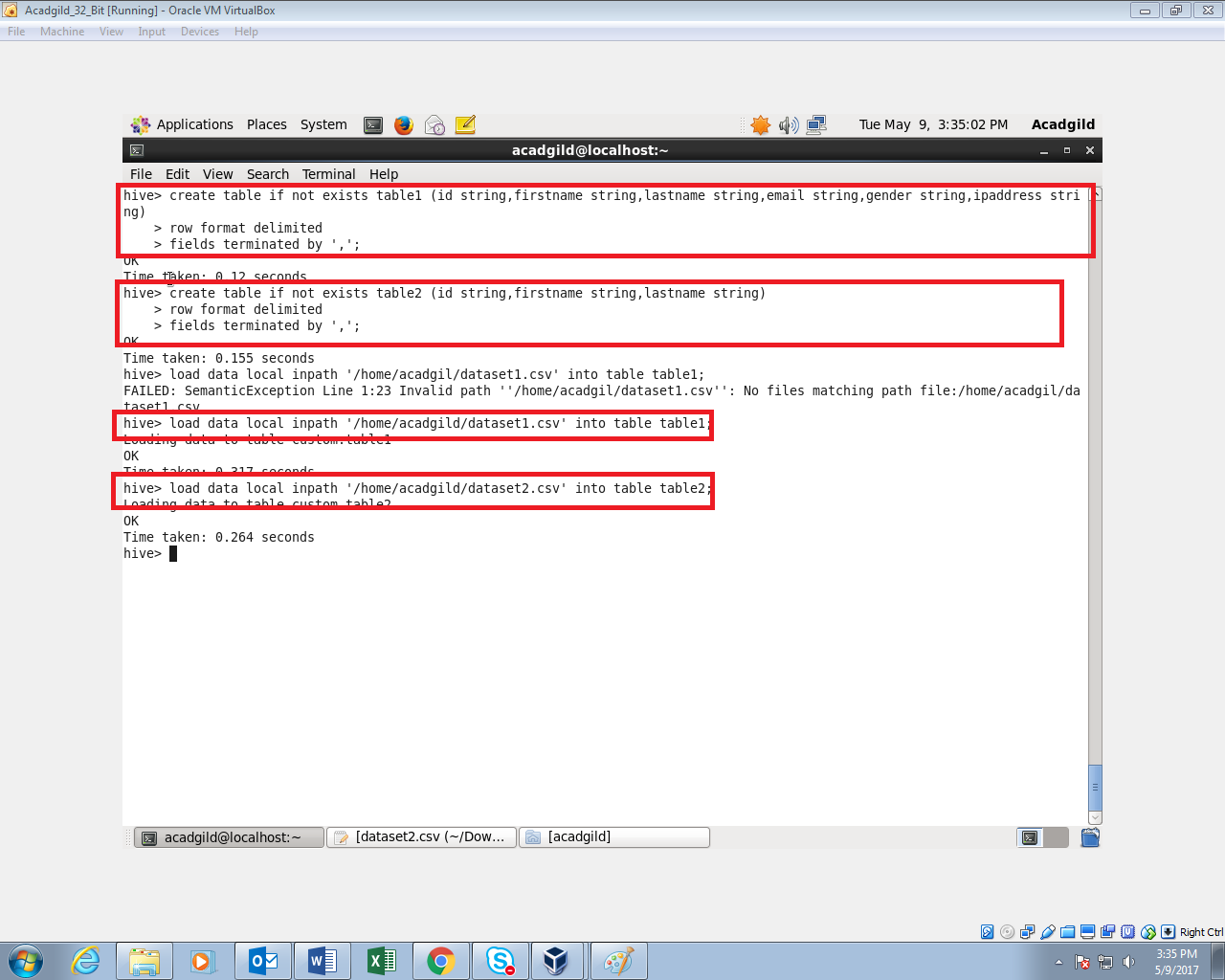
 Sort-Merge Bucket join

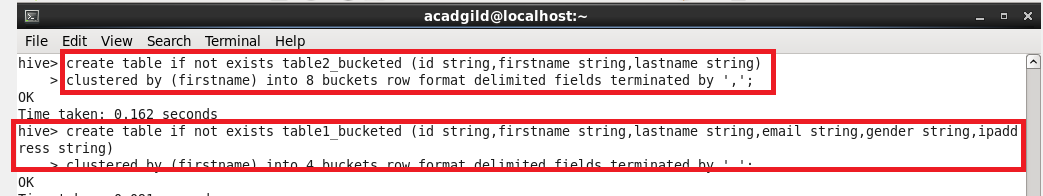
 Sort-Merge Bucket Map join

 Left semi join

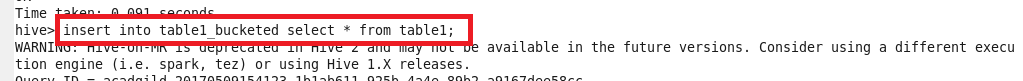
BUCKET MAP JOIN:

* Created 2 tables and loaded the data into the tables

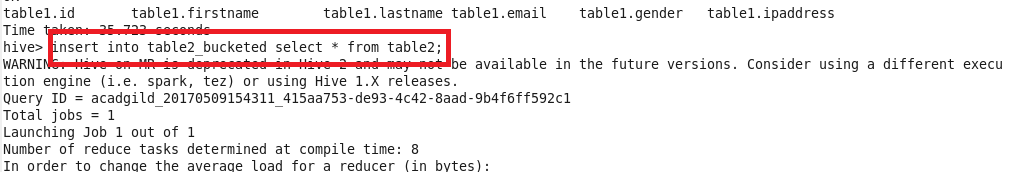
To perform bucketing, we need to have bucketed tables.so I’m creating 2 bucketed tables. One with 4 buckets and one with 8 buckets



Inserting the data in table1 into the bucketed table table1\_bucketed



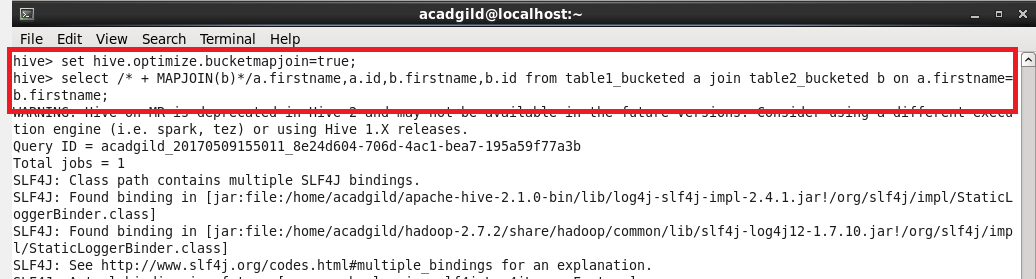
Inserting the data in table2 into the bucketed table table2\_bucketed



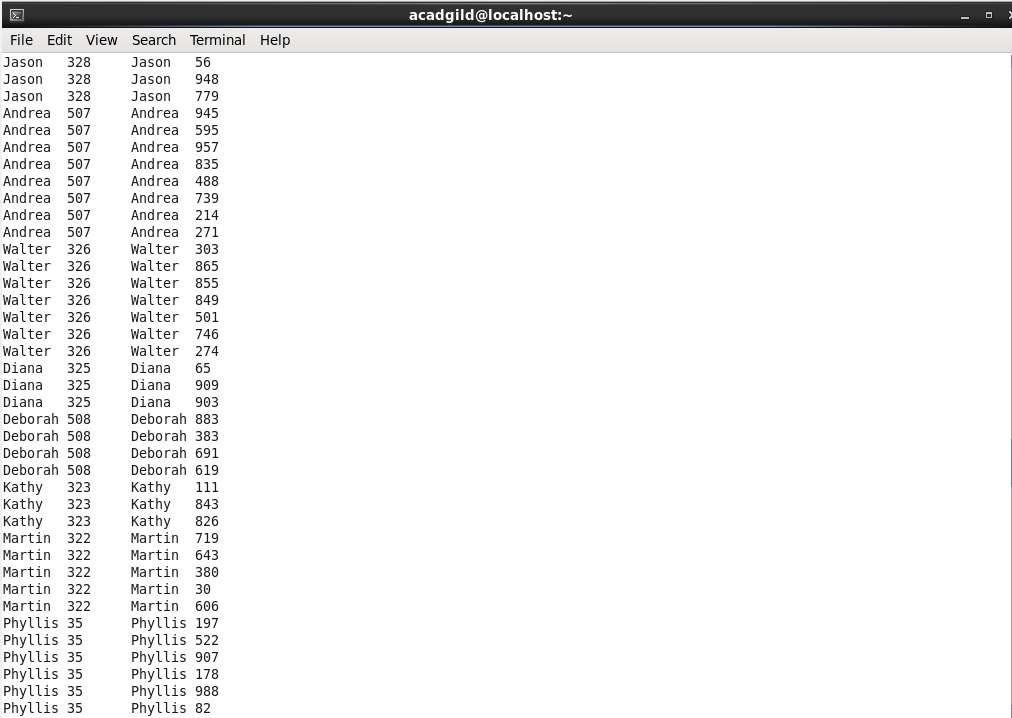
For performing Bucket-Map join, we need to set this property in the Hive shell.

set hive.optimize.bucketmapjoin = true

and writing query for extracting data.



OUTPUT:



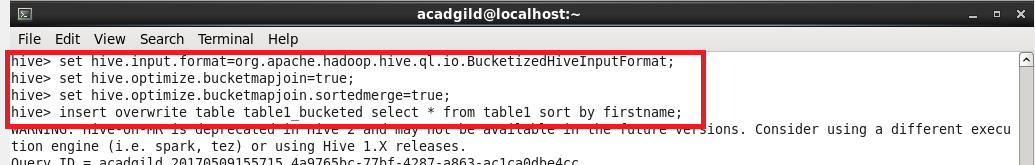
SORT-MERGE BUCKET JOIN:

Sort-Merge bucket join is like reduce side join. Before joining we’ll have to sort the data and then only merge.

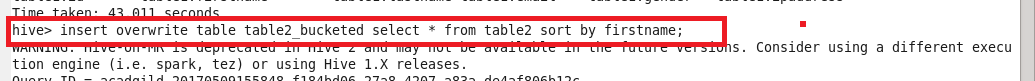
Moreover in this join we should set

set hive.auto.convert.join= false;

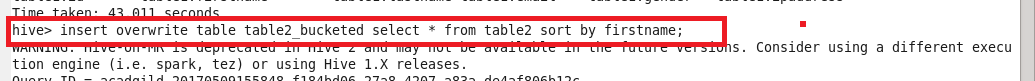
because we want reducer to run.



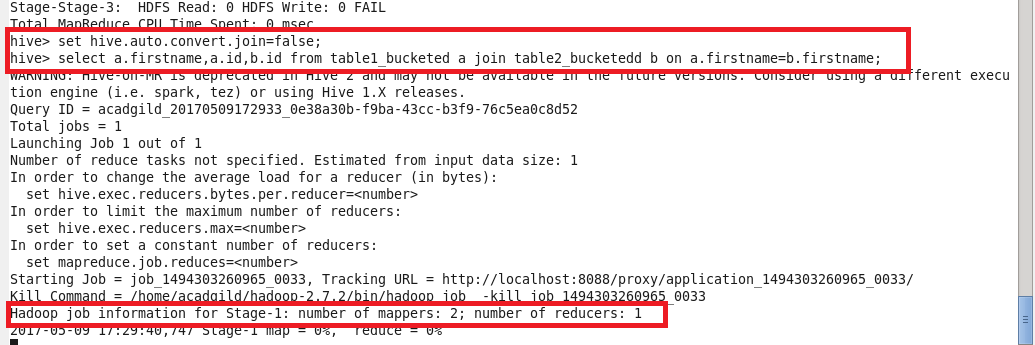
Sorting the contents of bucketed table 2



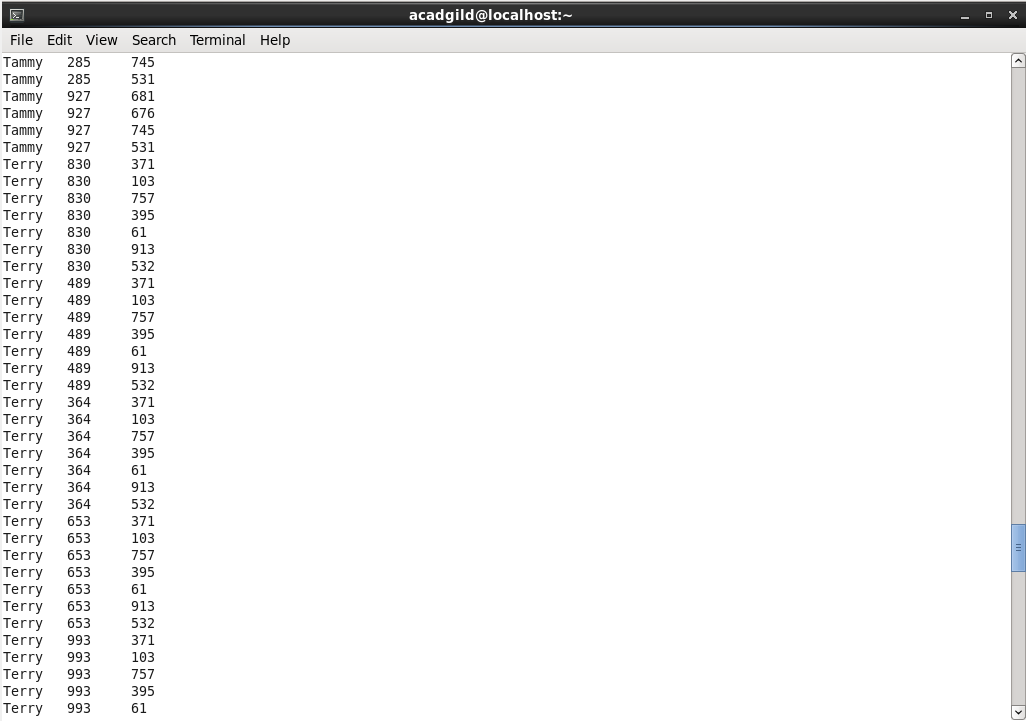
Sorting the contents of bucketed table 1



Writing query for extracting data



OUTPUT:

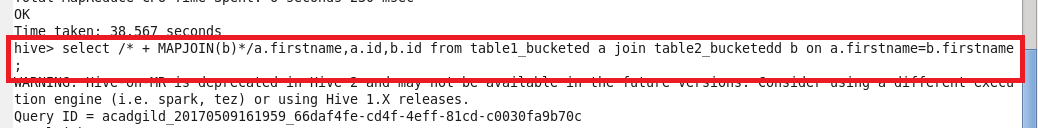


SORT-MERGE BUCKET MAP JOIN:

since it is sort-merge bucket map join we need to set

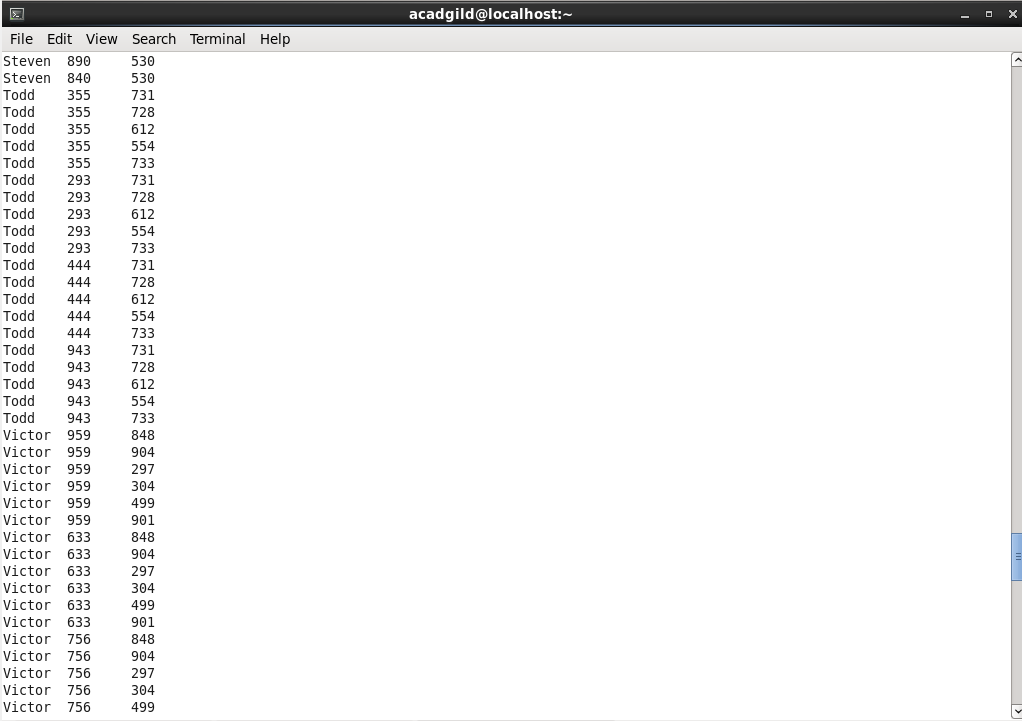
set hive.auto.convert.join= true;

but without setting this also we can run a map only job by typing like this



This query runs a map only job. I’ve used the same bucketed tables that I’ve created previously for this join

The output will be like this,



LEFT SEMI JOIN:

Left semi join is more or less like sub query in sql.

The left semi join is used in place of the IN/EXISTS sub-query in Hive. In a traditional RDBMS, the IN and EXISTS clauses are widely used whereas in Hive, the left semi join is used as a replacement of the same.

Lef semi join eliminates the use of where and in keywords that we use in typical sub queries.

QUERY:



OUTPUT:

