**ASSIGNMENT – 27.1**

**Explain the following in brief with an example.**

1. **Map side Join :**

Joins performed by Mapper are called as Map-side Joins.

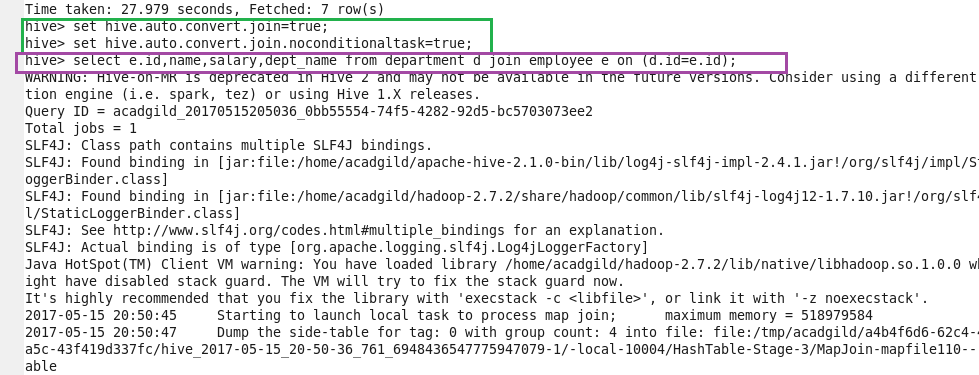
In case one of the dataset is small, map side join takes place.

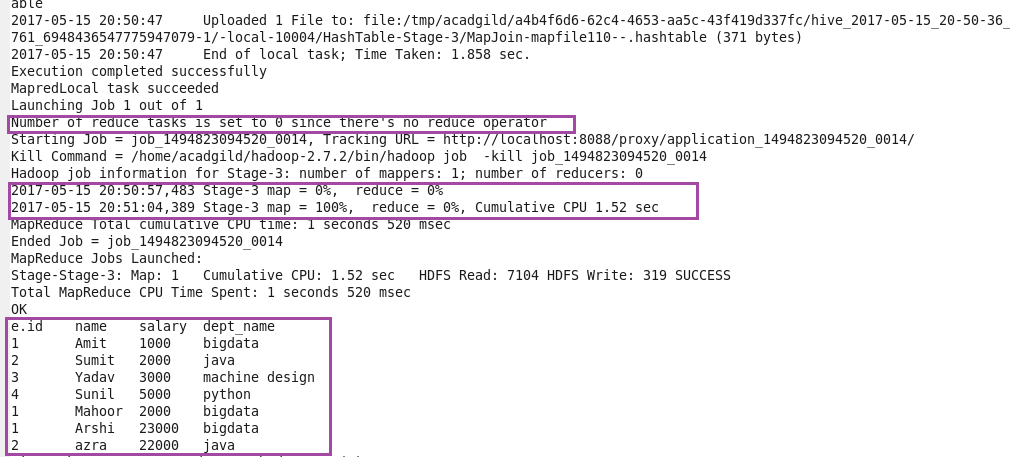
In map side join, a local job runs to create hash-table from content of HDFS file and sends it to every node.

To perform mapside join we need to set configurations:

Set hive.auto.convert.join=true;

Set hive.auto.convert.join.noconditionaltask=true;





See the highlighted part in purple color. You will find that there is no reducer phase performed in this join operation. Hence, the map-side join is faster than regular join operation.

1. **Reduce side Join :**

Joins performed by Reducer are called as Reduce-side Joins. If datasets are large, reduce side join takes place.

for eg : We can see in the Bucket Map join and SMBM join because datasets are large automatically reducer will come into picture.

1. **Bucket Map Join**

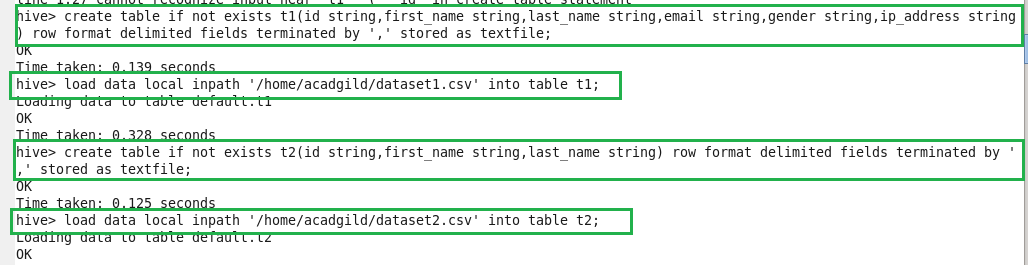
The data must be bucketed on the keys used in the ON clause and the number of buckets for one table must be a multiple of the number of buckets for the other table.

• When these conditions are met, Hive can join individual buckets between tables in the map phase, because it does not have to fetch the entire content of one table to match against each bucket in the other table.

• set hive.optimize.bucketmapjoin=true;

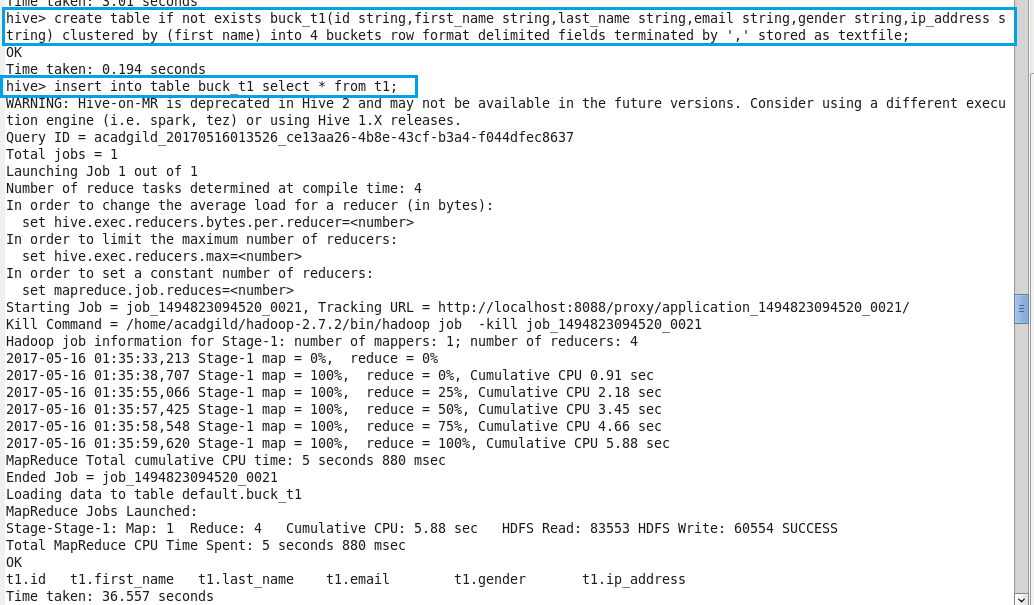
• SET hive.auto.convert.join=true;

Creating tables and loading the data into the created table.

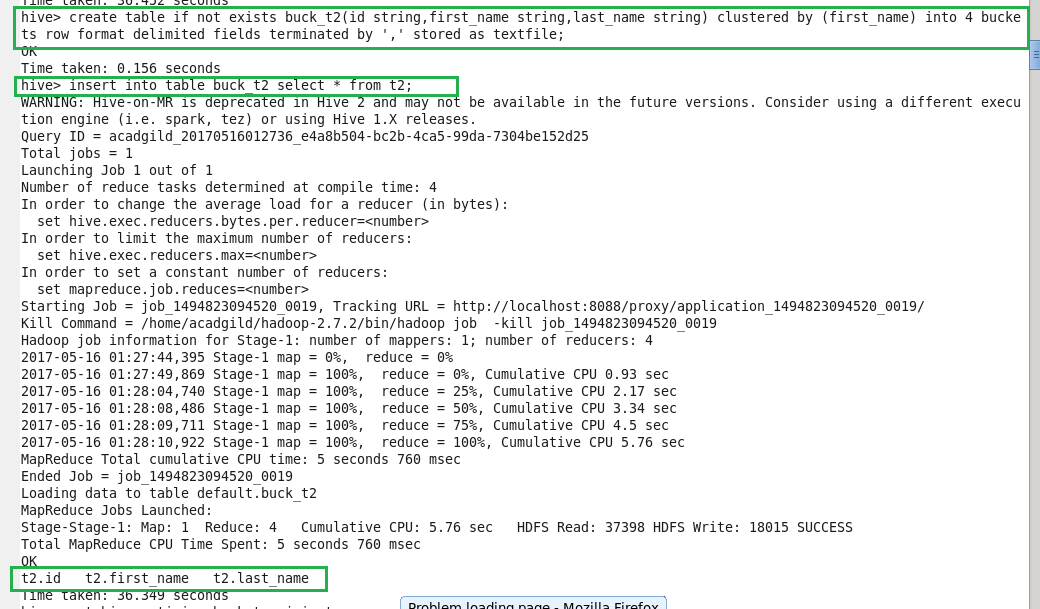


Creating bucketed tables to perform bucket map join and inserting data from the existing tables.

**Created table – buck\_t1**



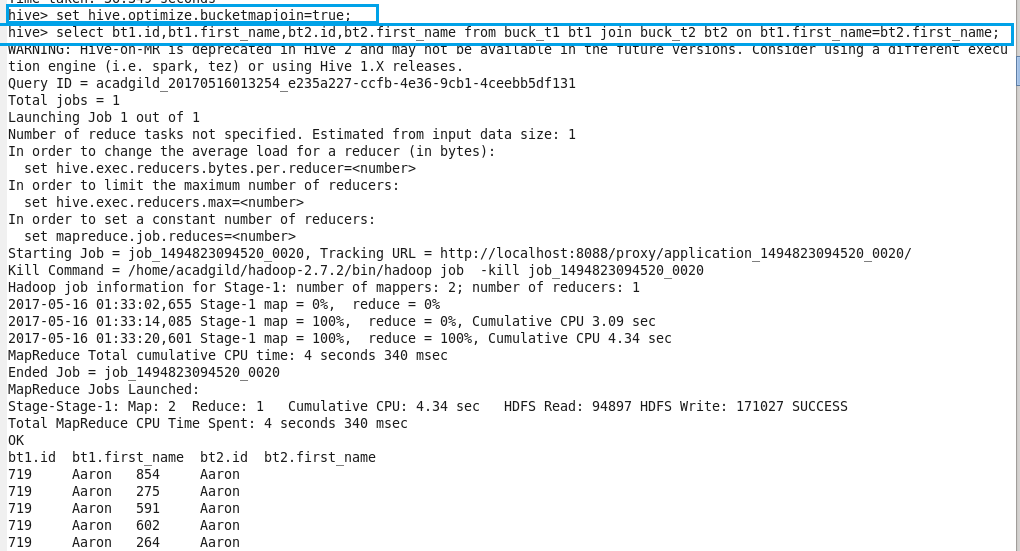
**Created table – buck\_t2**

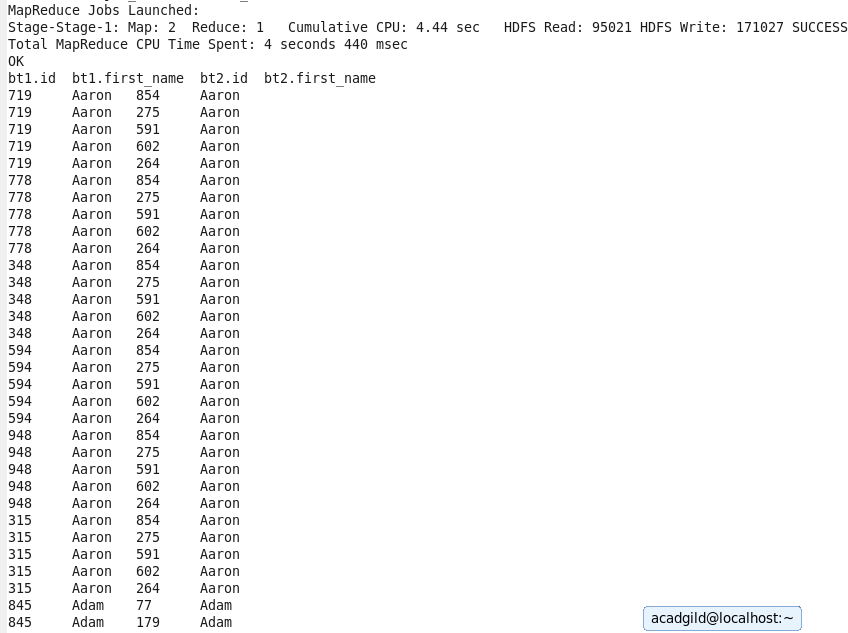


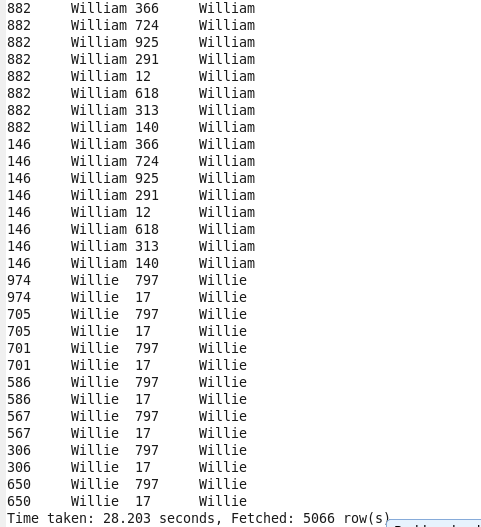
To perform bucket map join we need to set configuration i.e,

Set hive.optimize.bucketmapjoin=true;

With attached output :







**SMBM Join**

**Sort-Merge-Bucket (SMB) Map Join:**

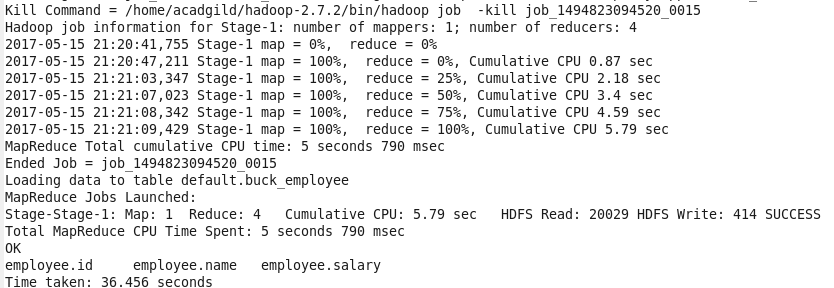
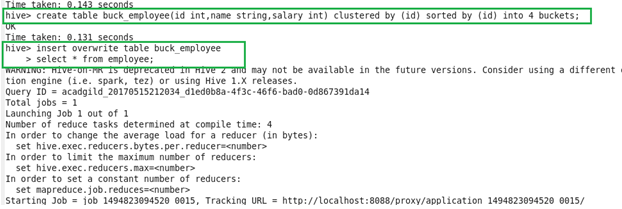
It is another Hive join optimization technique where all the tables need to be bucketed and sorted. In this case joins are very efficient because they require a simple merge of the presorted tables.

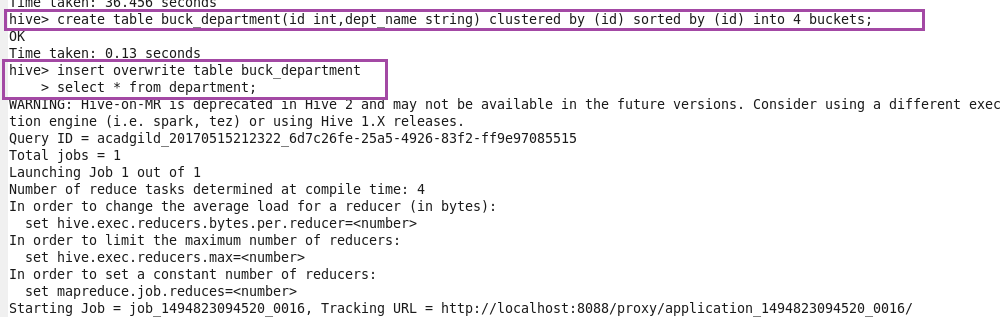
Creating bucketed tables from existing tables i.e, employee and department. Before that we need to set the configurations.

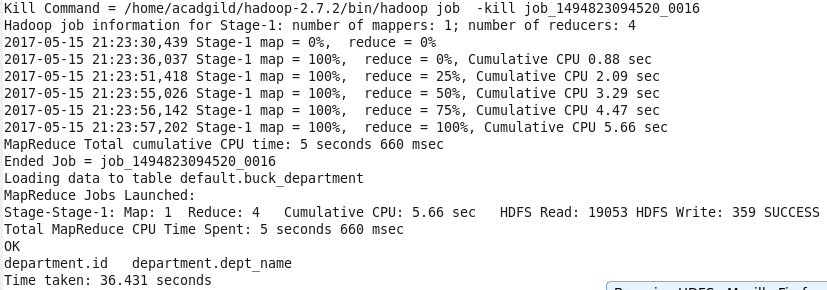
Set hive.enforce.bucketing=true;

Set hive.enforce.sorting=true;

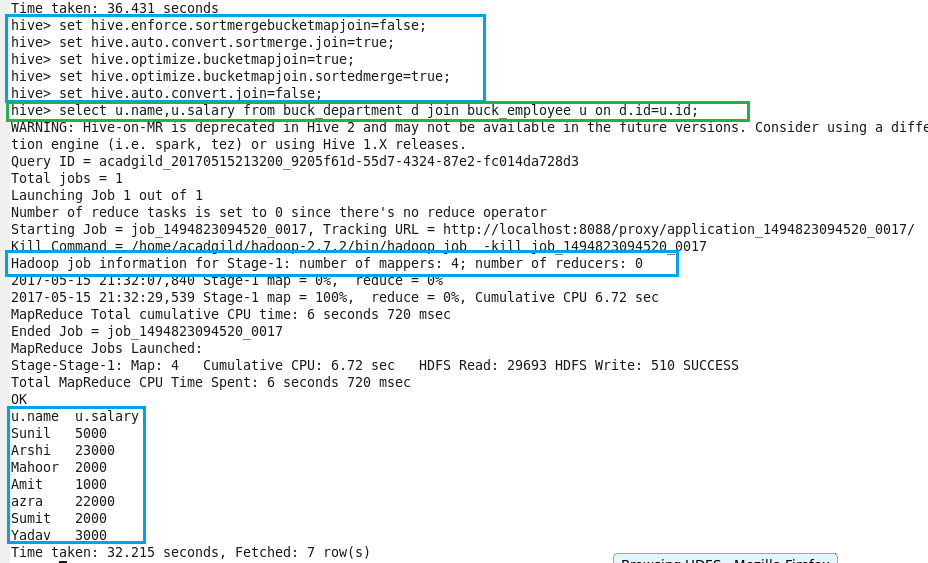
Creating bucketed table as buck\_employee and inserting to it.



Creating bucketed table as buck\_department and inserting to it.



Now the stage is set to perform SMB Map Join to optimize Hive joining. Again, make some changes in properties to perform SMB Map join.



See highlighted part in blue color. You will find that 4 mapper tasks are running (as we had 4 buckets). This helps in performing faster join operation when compared to regular.