Problem Statement:

Perform join optimizations in hive by following the steps in the below blog link

<https://acadgild.com/blog/join-optimization-in-apache-hive/>

For join optimization in Hive, we can use repartition joins, replication joins and semi joins.

Table creation:

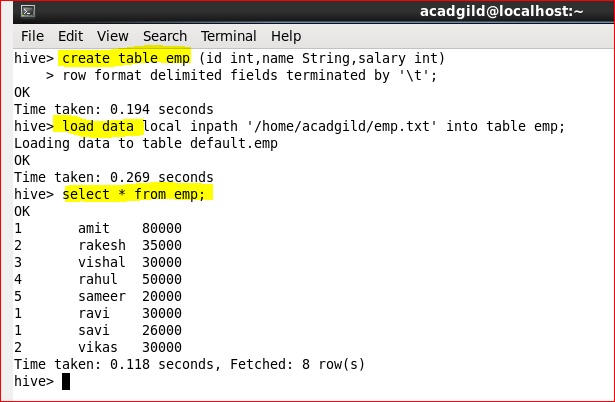
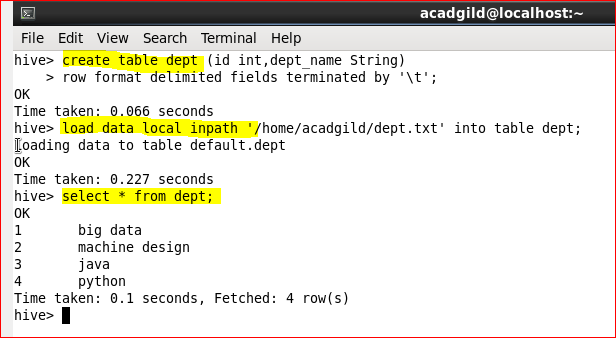
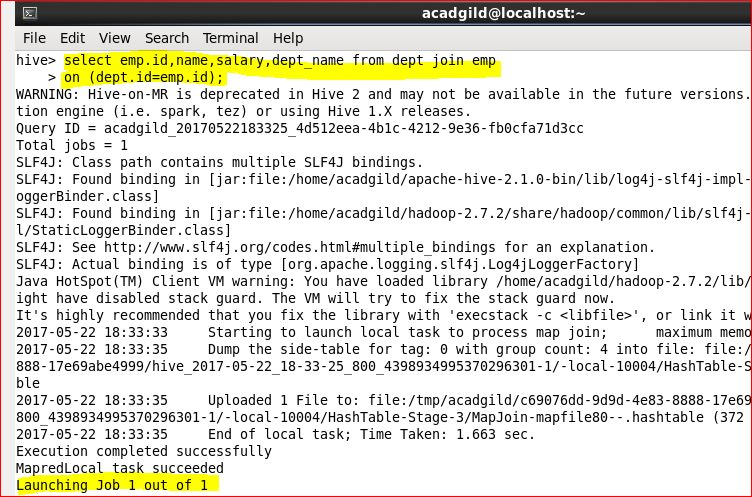


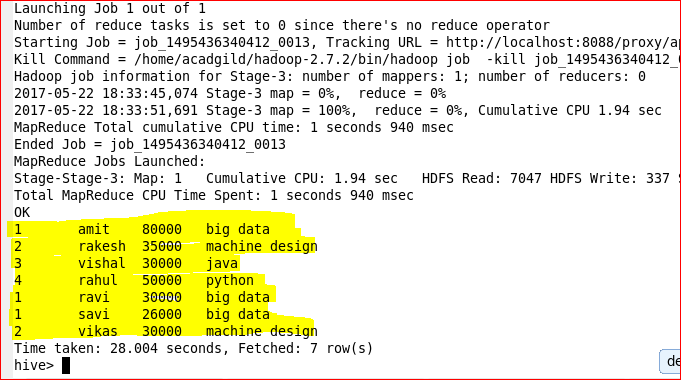
Table creation:



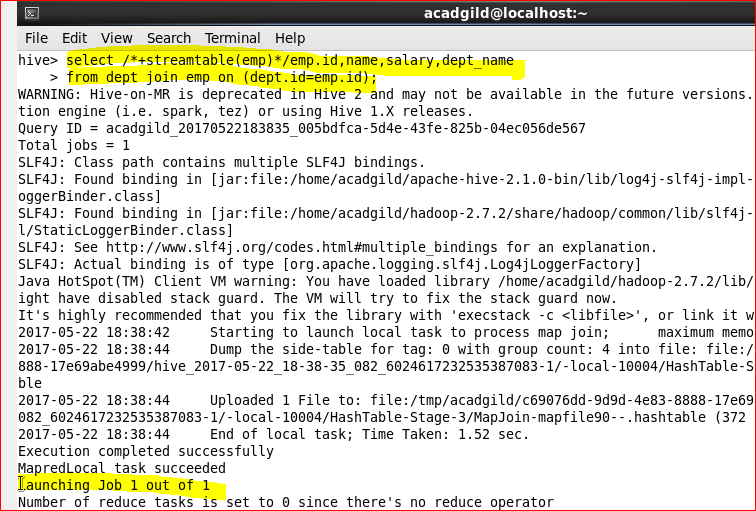
**Join table ordering (Largest table last)**.

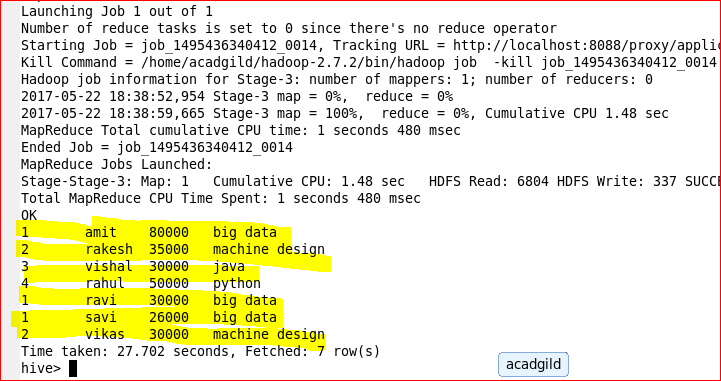
The ‘emp’ table consists of department id, employee name, and employee salary. For any organization, this list can keep growing over the time. But, the ‘dept’ table will be static for most of the time.  Hence, when these two tables are joined it is important that the larger table comes last in the query.





Or we can also explicitly tell Hive which table it should stream.

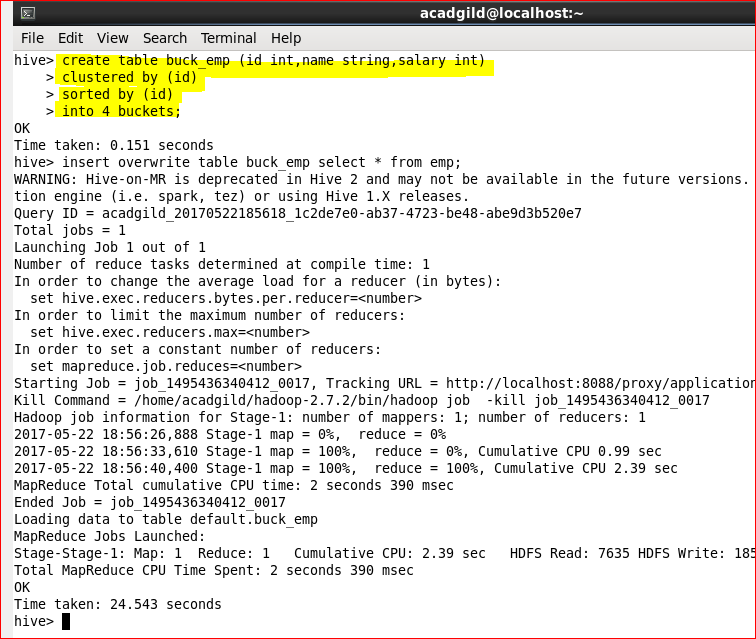


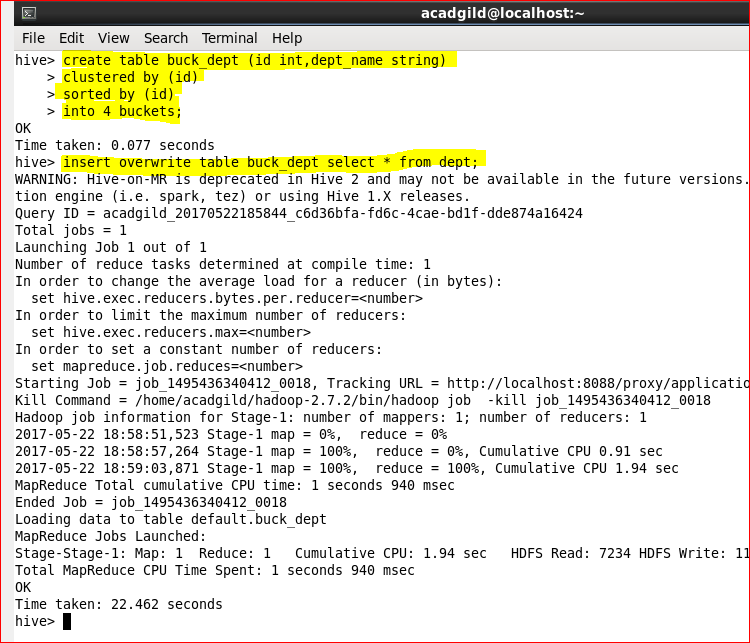


**Sort-Merge-Bucket (SMB) Map Join:**

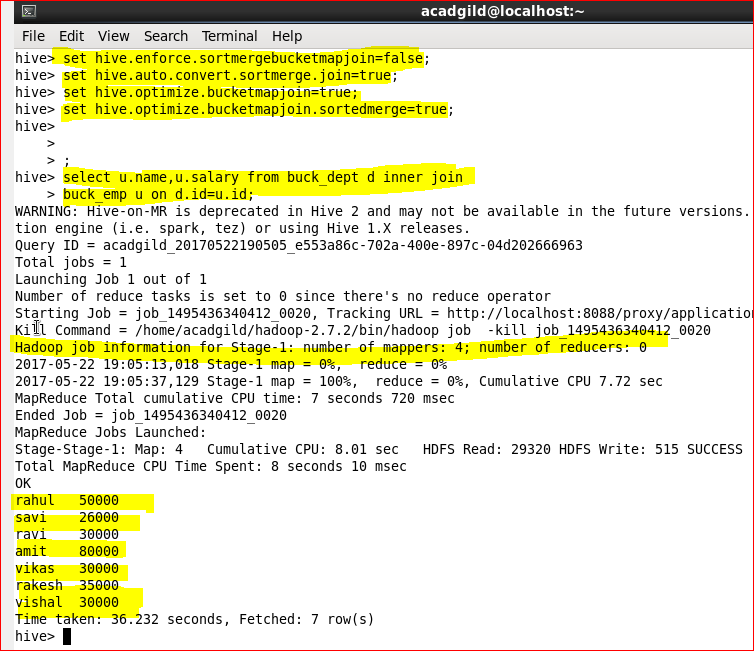
It is another Hive join optimization technique where all the tables need to be bucketed and sorted. In this case joins are very efficient because they require a simple merge of the presorted tables.

Let us create bucketed tables from our existing tables i.e.; emp and dept.





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



4 mapper tasks are running (as we had 4 buckets). This helps in performing faster join operation when compared to regular joins.