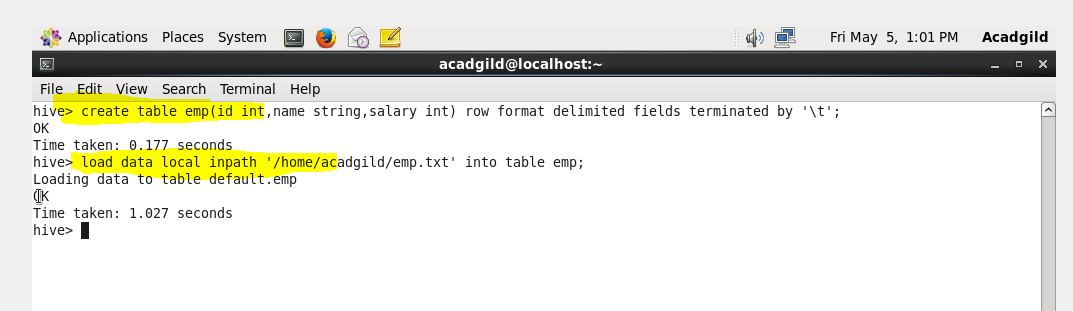
**Assignment 27.5**

**Perform join optimizations in hive**

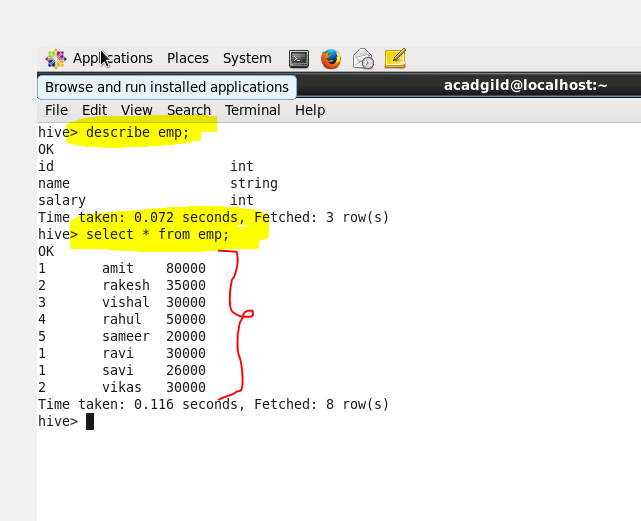
[Hive](https://acadgild.com/blog/hive-beginners-guide/), like other SQL databases, allows users to join various tables. However, Joins can be computationally expensive, especially on big tables. Hive on top of [Hadoop](https://acadgild.com/big-data/big-data-development-training-certification) makes data processing so straightforward and scalable that we can easily forget to optimize our [Hive queries](https://acadgild.com/blog/run-hive-queries-using-spark-sql/).

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

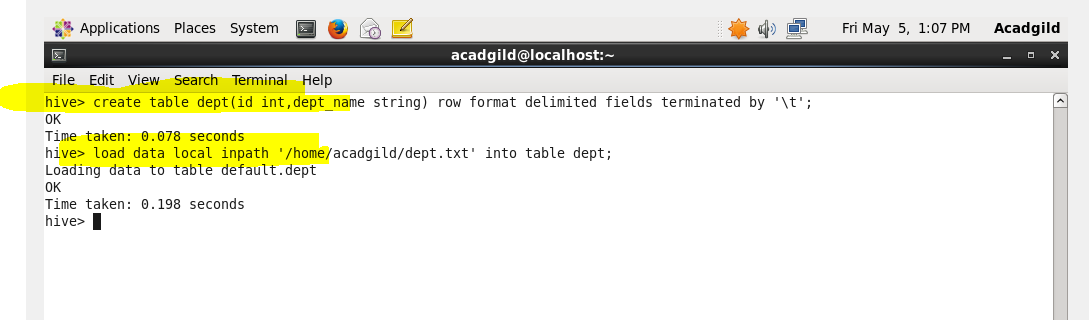
**Creating & loading data in table emp**

****

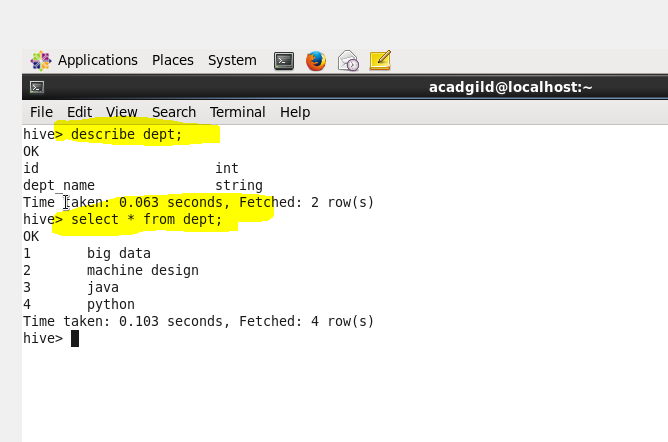
**Displaying table emp**

****

**Creating & loading data in table dept**

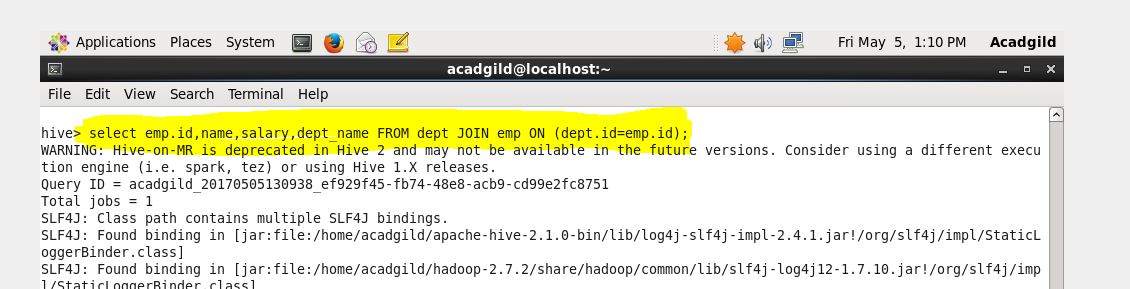
****

**Displaying table dept**

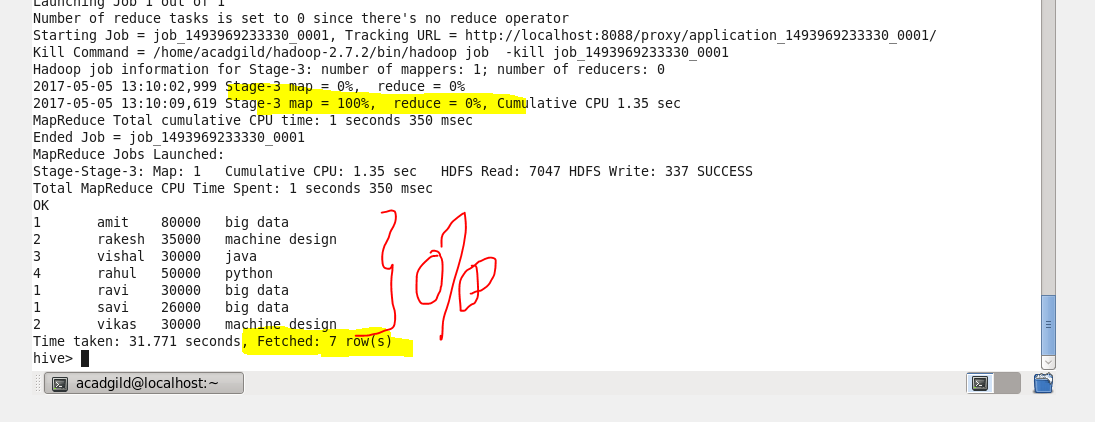
****

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

The ‘emp’ table consists of department id, employee name, and employee salary.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.

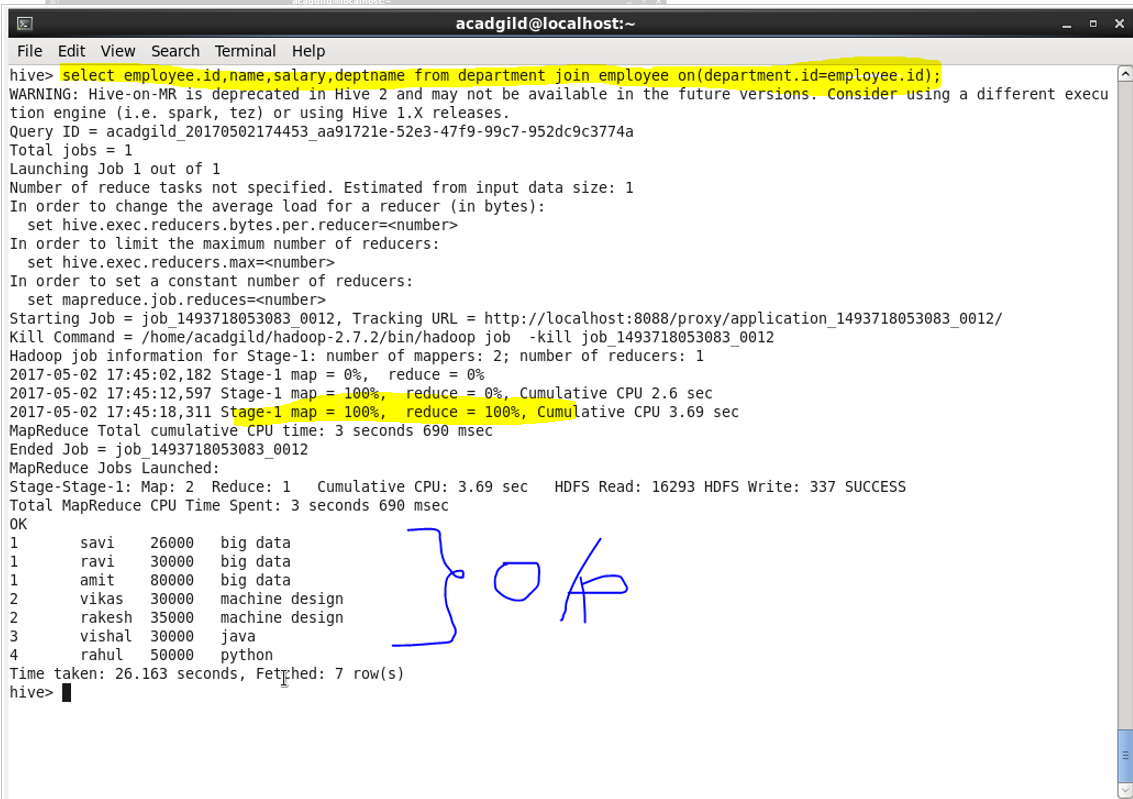


**Output**



**Map Side Join:**

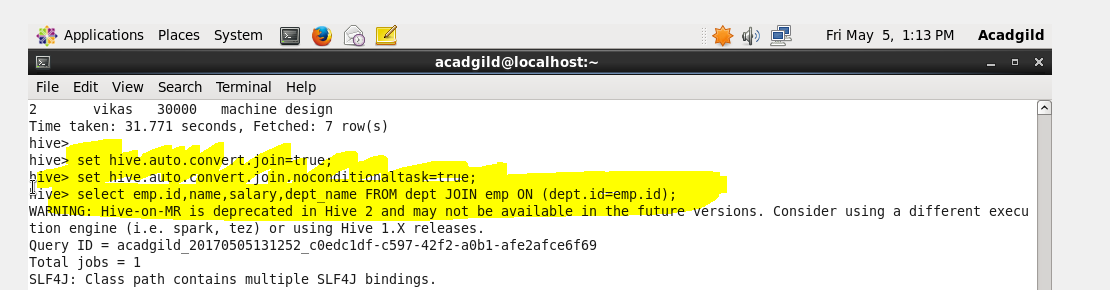
A map-side join is a special type of join where a smaller table is loaded in memory and join is performed in map phase of MapReduce job. Since there is no reducer involved in the map-side join, it is much faster when compared to regular join.

****

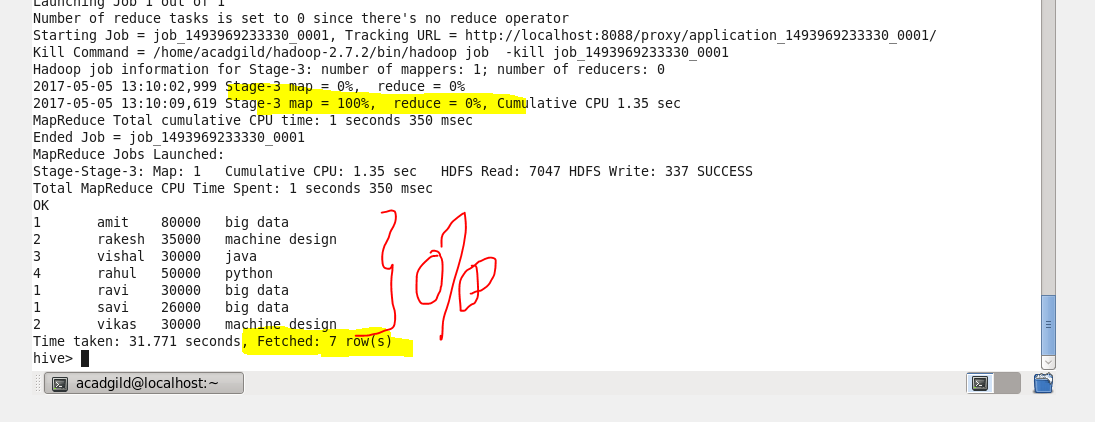
Now, to perform map-side join, setting few configurations

**hive> set hive.auto.convert.join=true;**

**hive> set hive.auto.convert.join.noconditionaltask=true;**



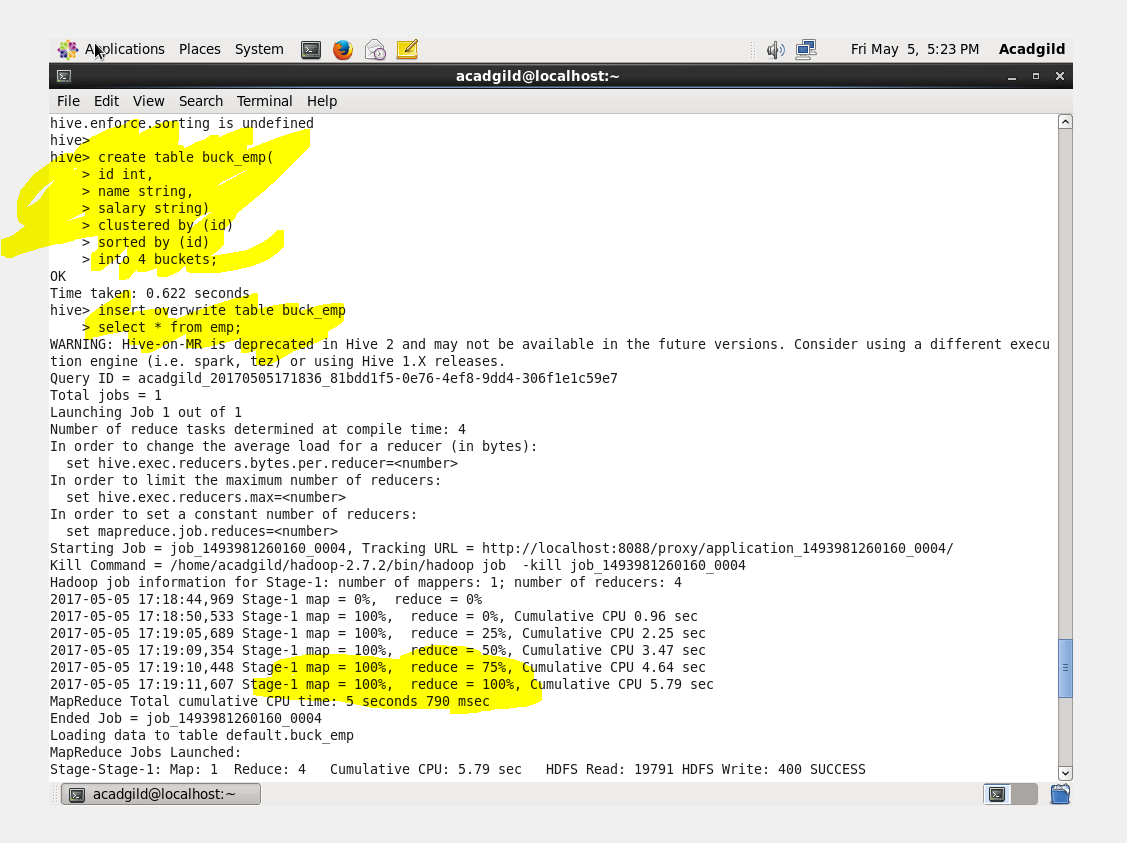
**Output**



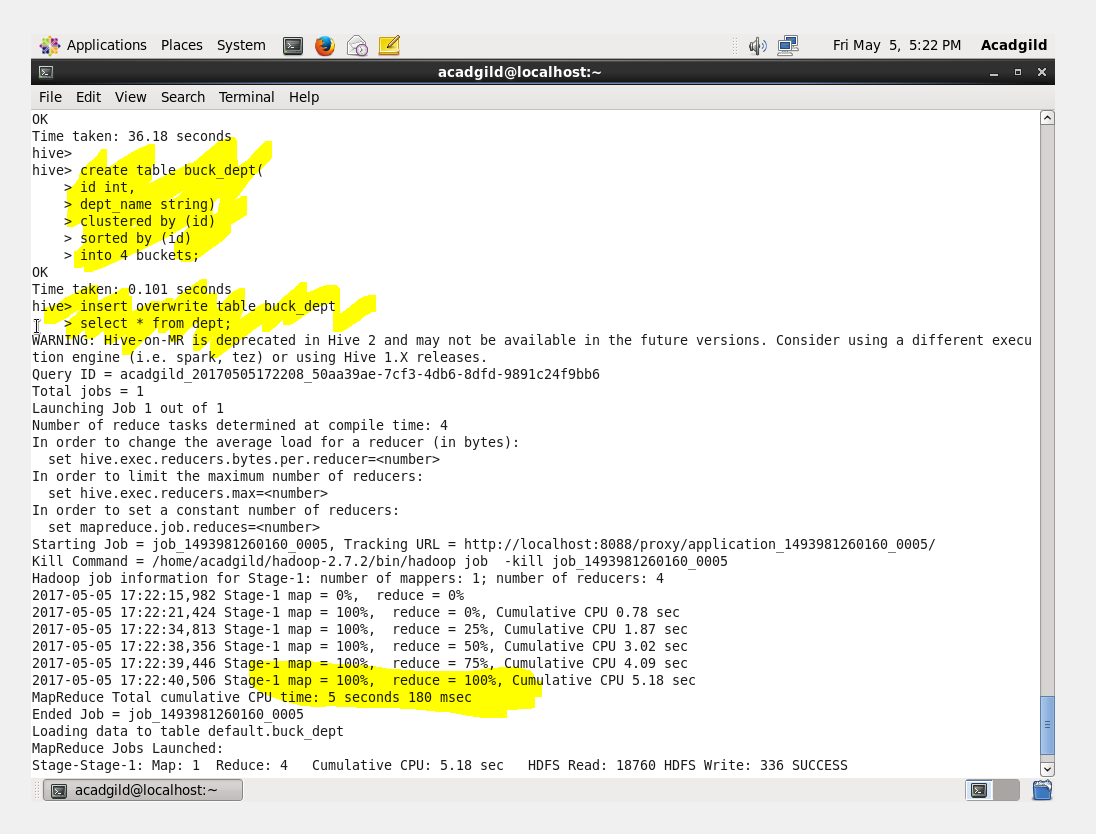
**Sort-Merge-Bucket Join**

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 a bucket table emp**

****

**Creating a bucket table dept**

****

**Output**

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.

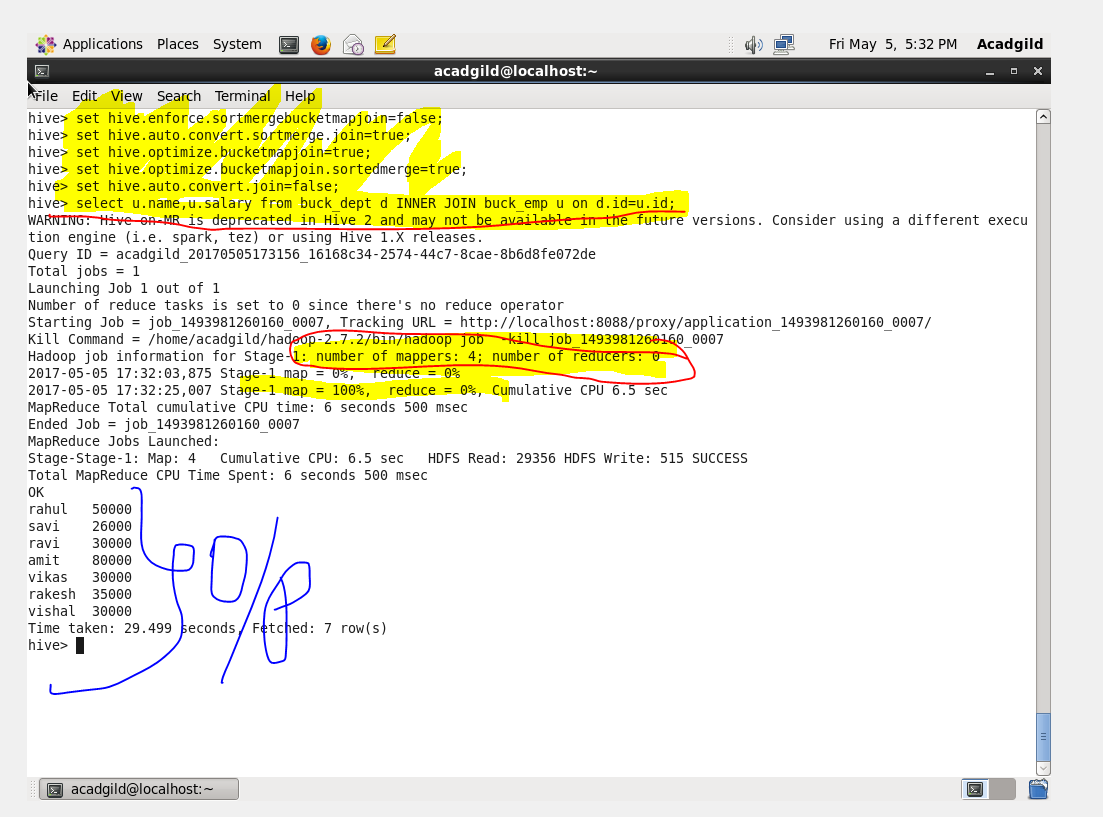
**hive>set hive.enforce.sortmergebucketmapjoin=false;**

**hive>set hive.auto.convert.sortmerge.join=true;**

**hive>set hive.optimize.bucketmapjoin = true;**

**hive>set hive.optimize.bucketmapjoin.sortedmerge = true;**

**hive>set hive.auto.convert.join=false**;

****