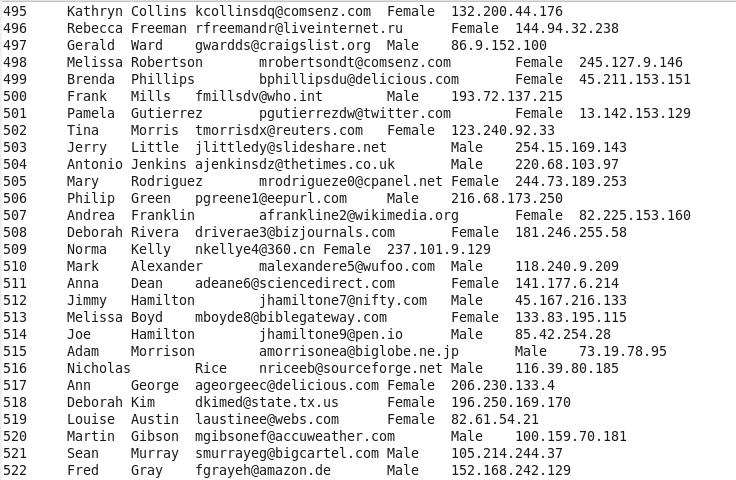
**ASSIGNMENT 27.6**

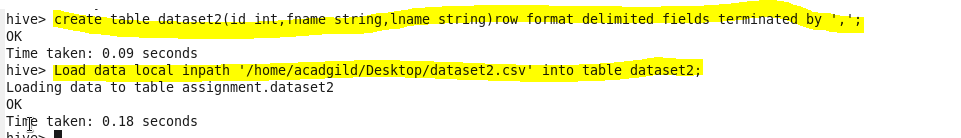
* **Using the below given two datasets you need to give a demo on the below joins in hive.**
* **Bucket Map join**
* **Sort-Merge Bucket join**
* **Sort-Merge Bucket Map join**
* **Left semi join**

**First we create the two tables:**

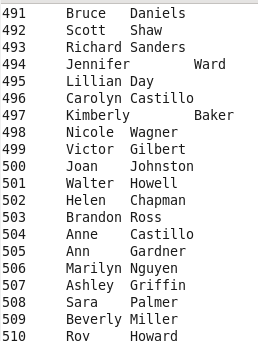


**Dataset1**





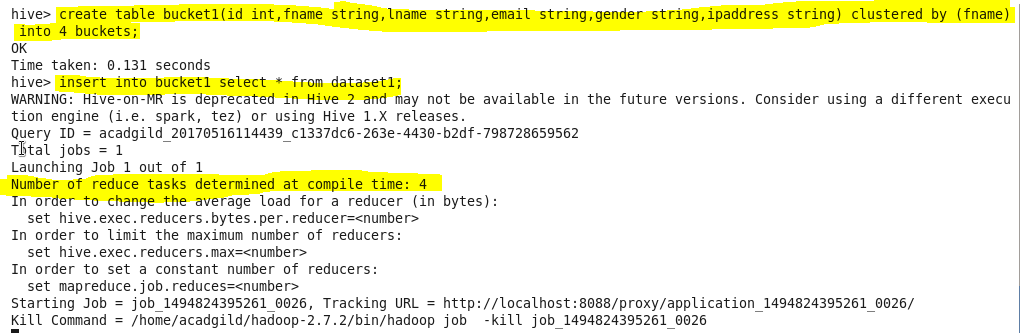
**Dataset2**

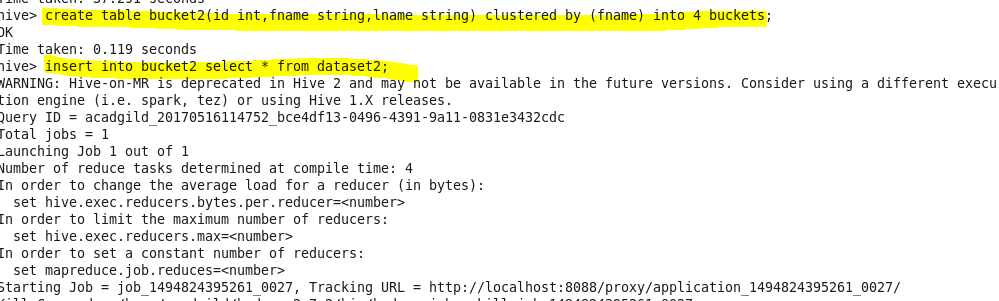


* **Bucket map join**

A bucket map join is used when the tables are large and all the tables used in the join are bucketed on the join columns. In this type of join, one table should have buckets in multiples of the number of buckets in another table.

We create the bucketed tables to hold the data.



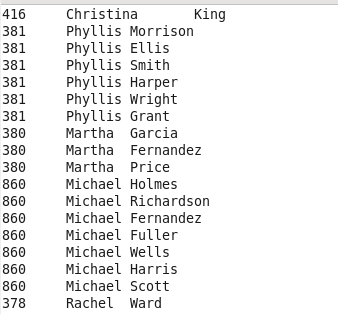


To perform bucket map join we need to set configuration i.e ,

* Set hive.optimize.bucketmapjoin=true;



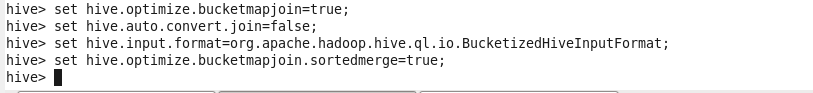
Output:

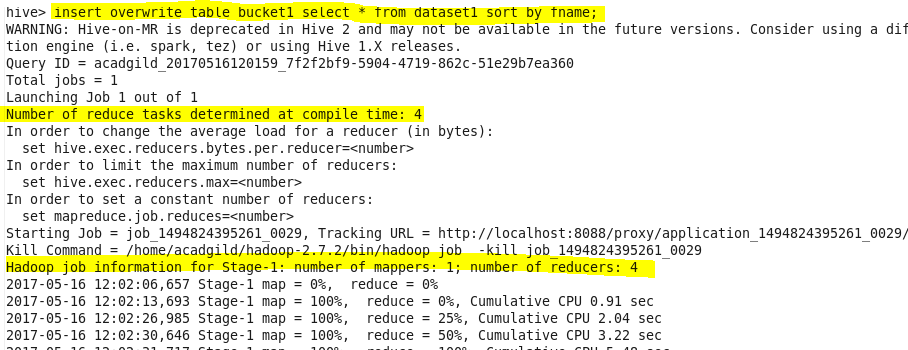


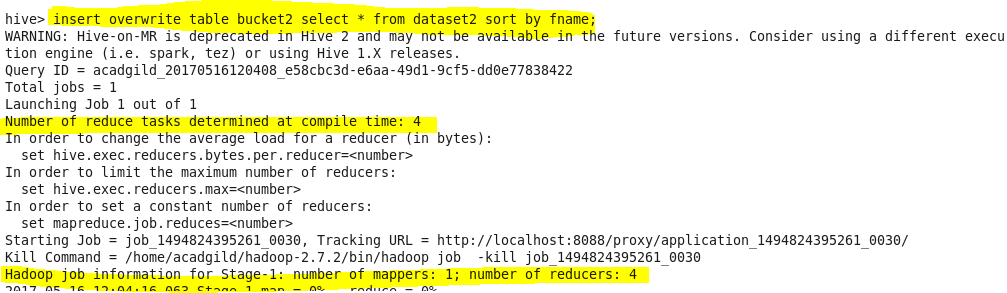
**Sort-Merge Bucket join :**

In SMB join in hive,each mapper needs a bucket from the first table and the corresponding bucket from the second table and then a merge sort join is performed. It is mainly used as there is no limit on file or partition or table join. It can best be used when the tables are large. In this join columns are bucketed and sorted using the join columns. All tables should have the same number of buckets in SMB join.

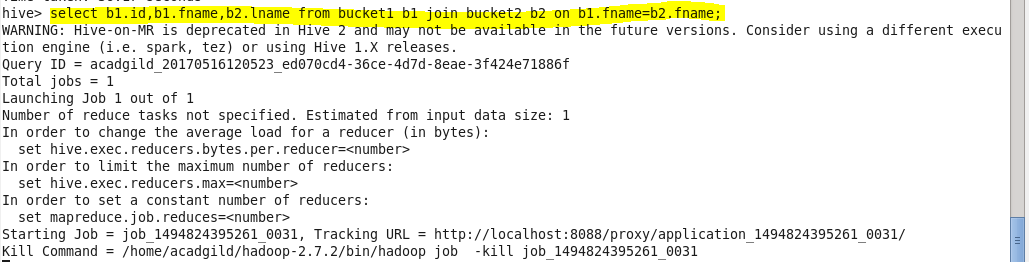
To perform SMB join we need to set following configuration showing in the attached screenshot then inserting the data in created bucketed tables.







Query:



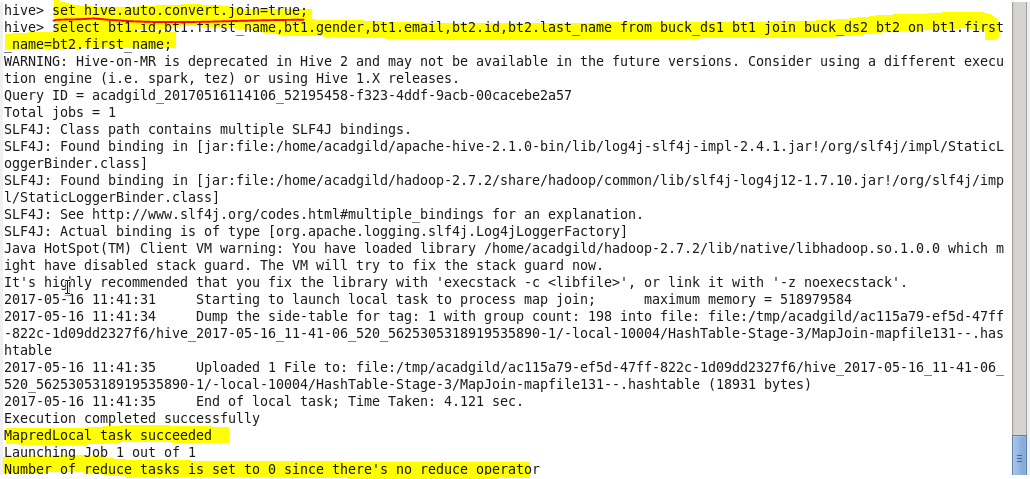
Output:

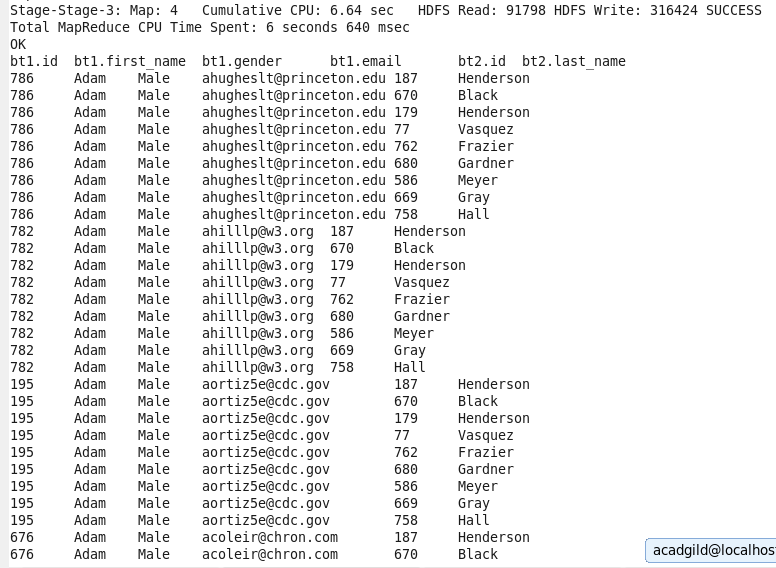


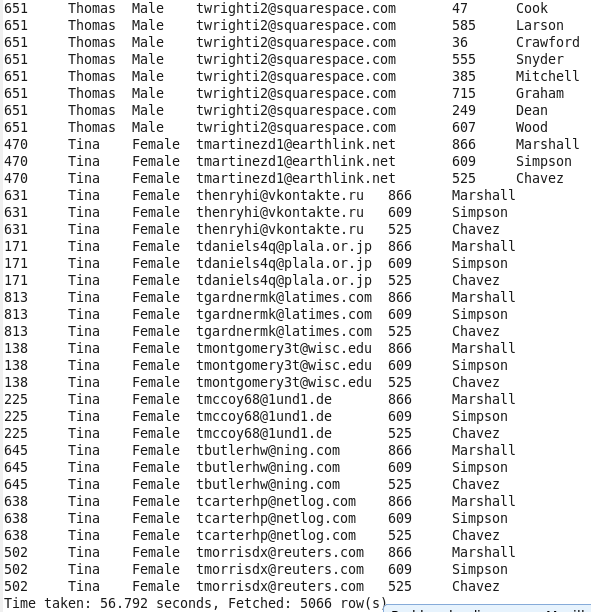
**SORT-MERGE BUCKET MAP JOIN:**

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.

After setting configuration performing SMBM join on created bucketed tables attached with the output.







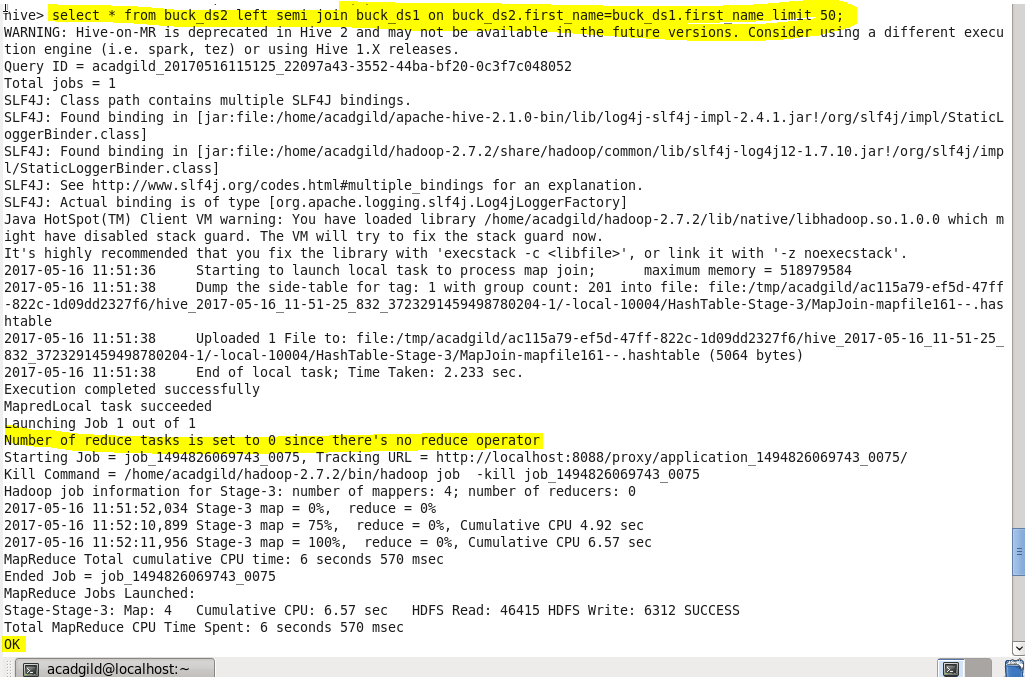
**LEFT SEMI JOIN:**

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.

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

It will display all the details of the Table on left side of Join command If a matching value for the join column exists in right table

**For example** In this case all the first 50 details of Dataset2 is displayed for every matchingId in right column



**Output :**

