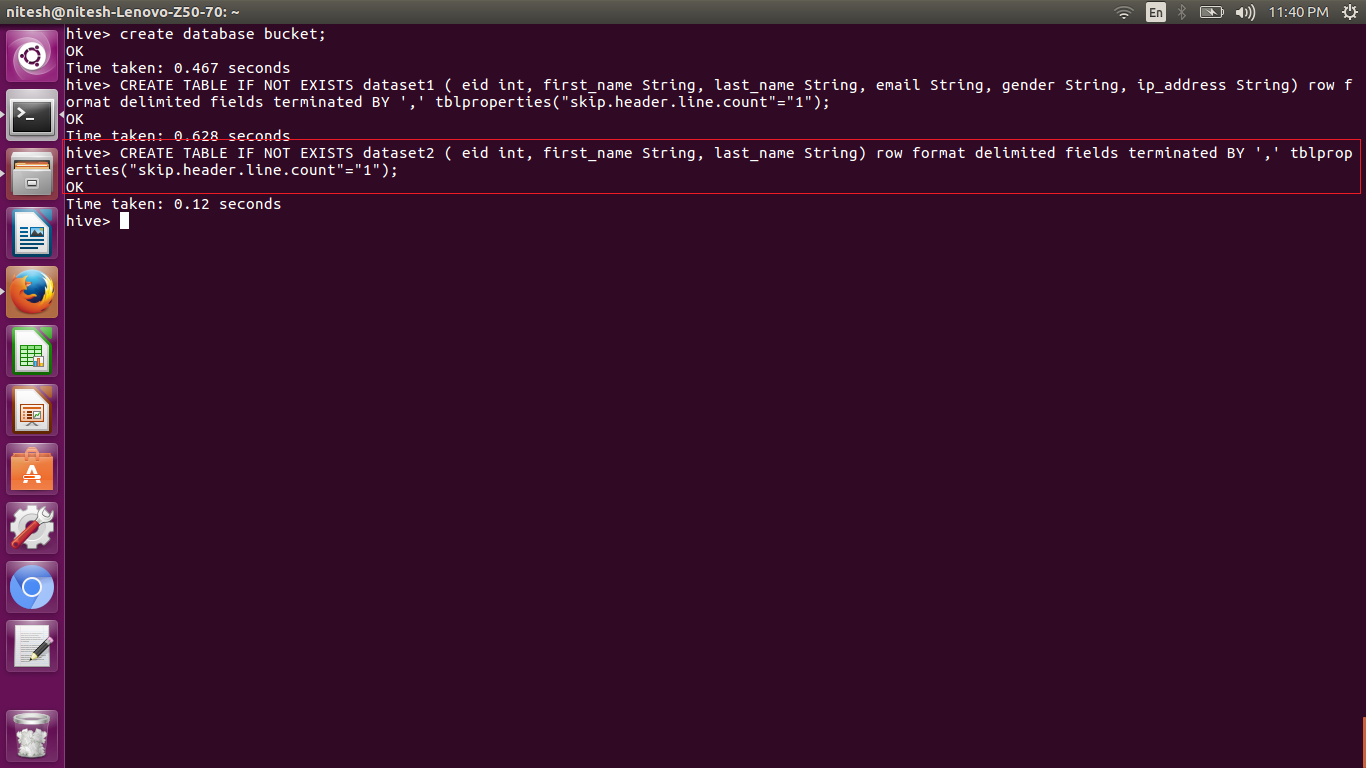
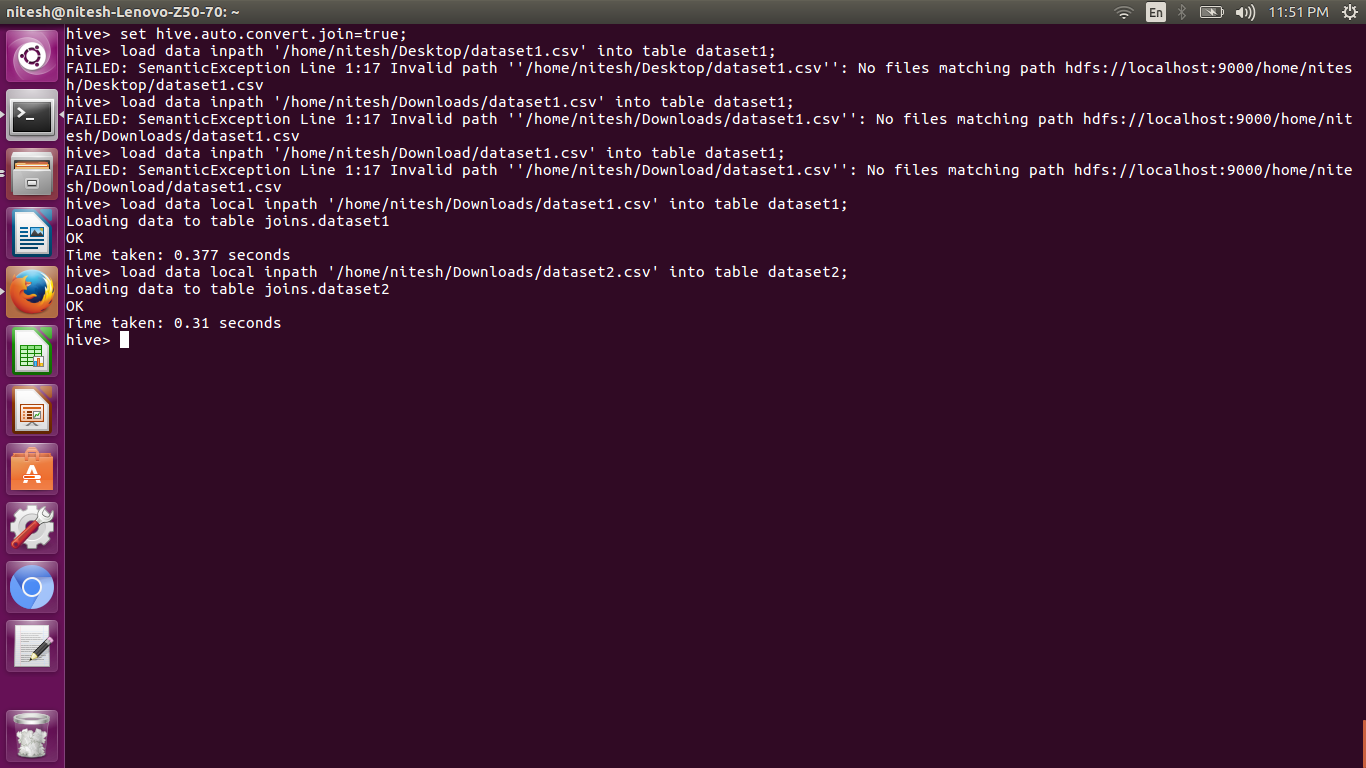
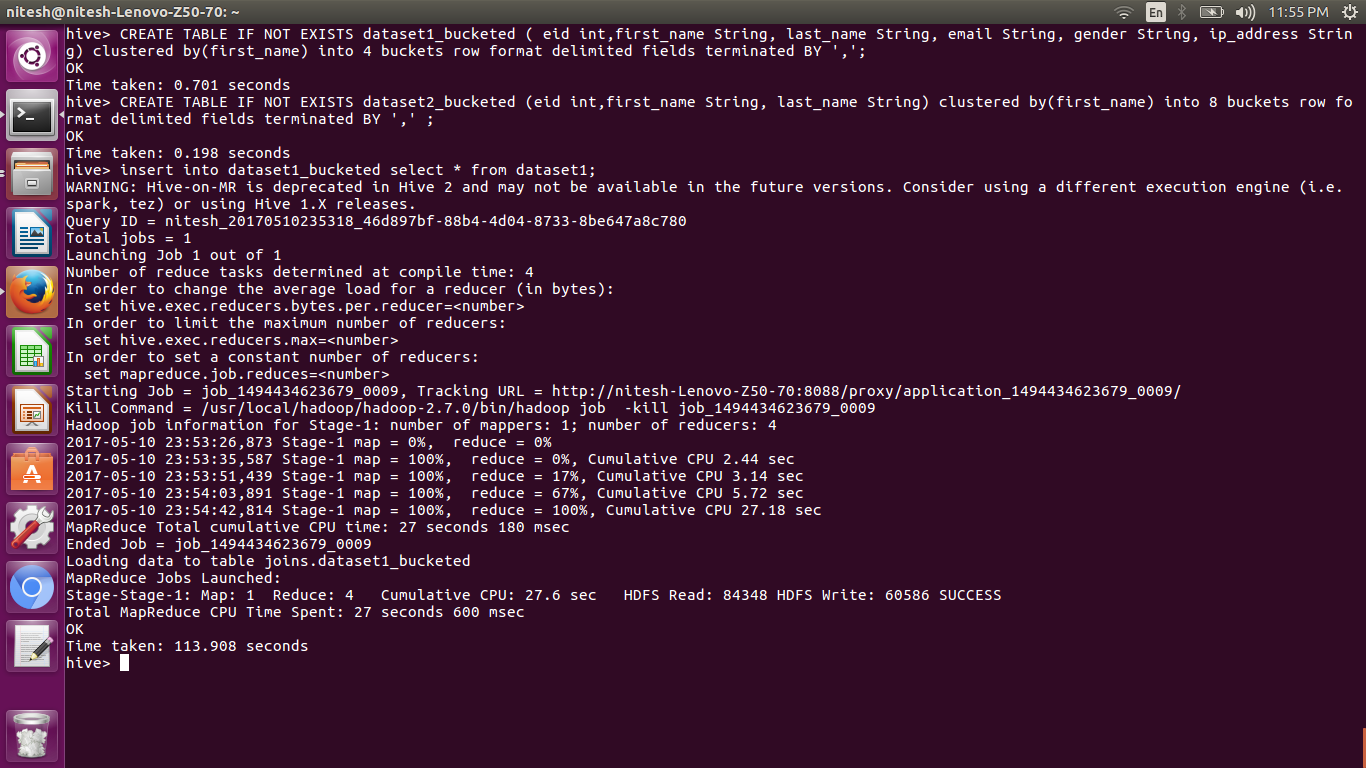
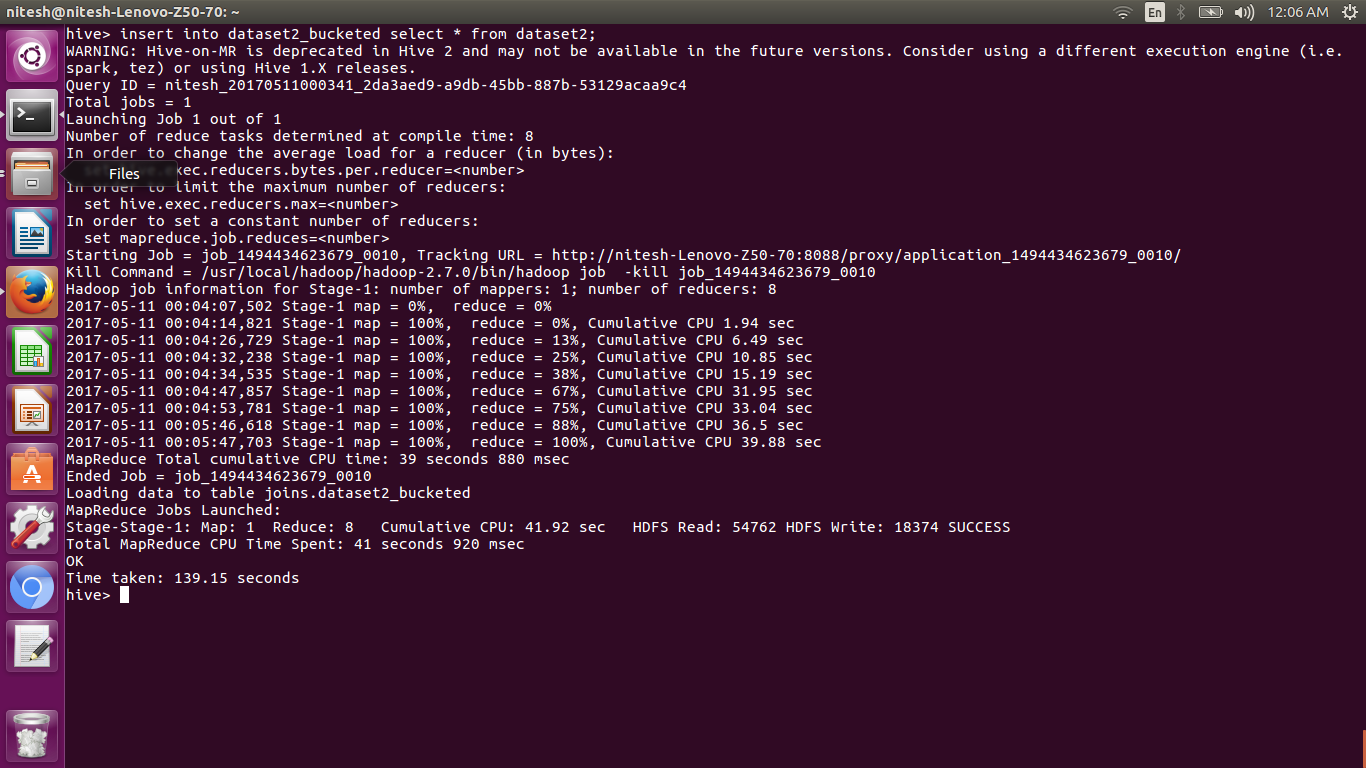
Using the below given two datasets you need to give a demo on the below joins in hive.

* Bucket Map join
* The constraint for performing Bucket-Map join is:
* If tables being joined are bucketed on the join columns, and the number of buckets in one table is a multiple of the number of buckets in the other table, the buckets can be joined with each other.





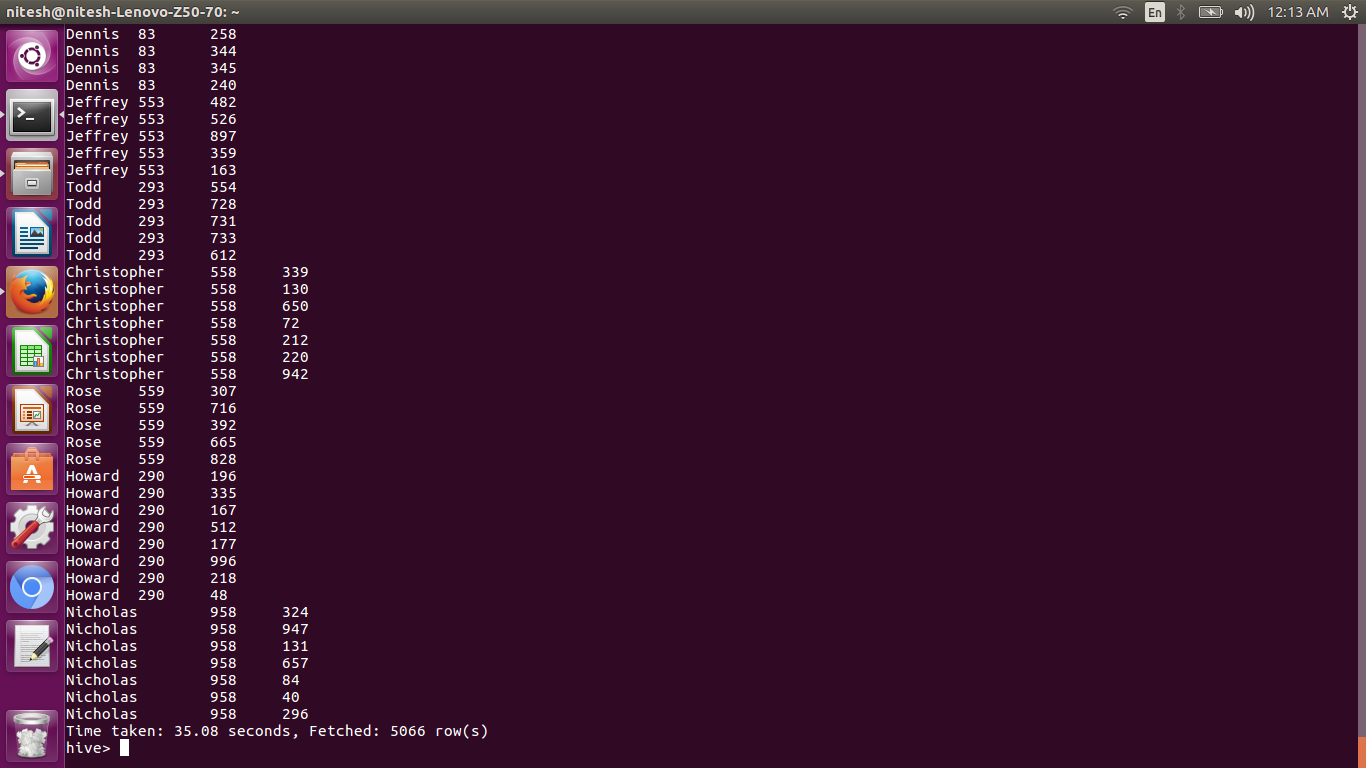








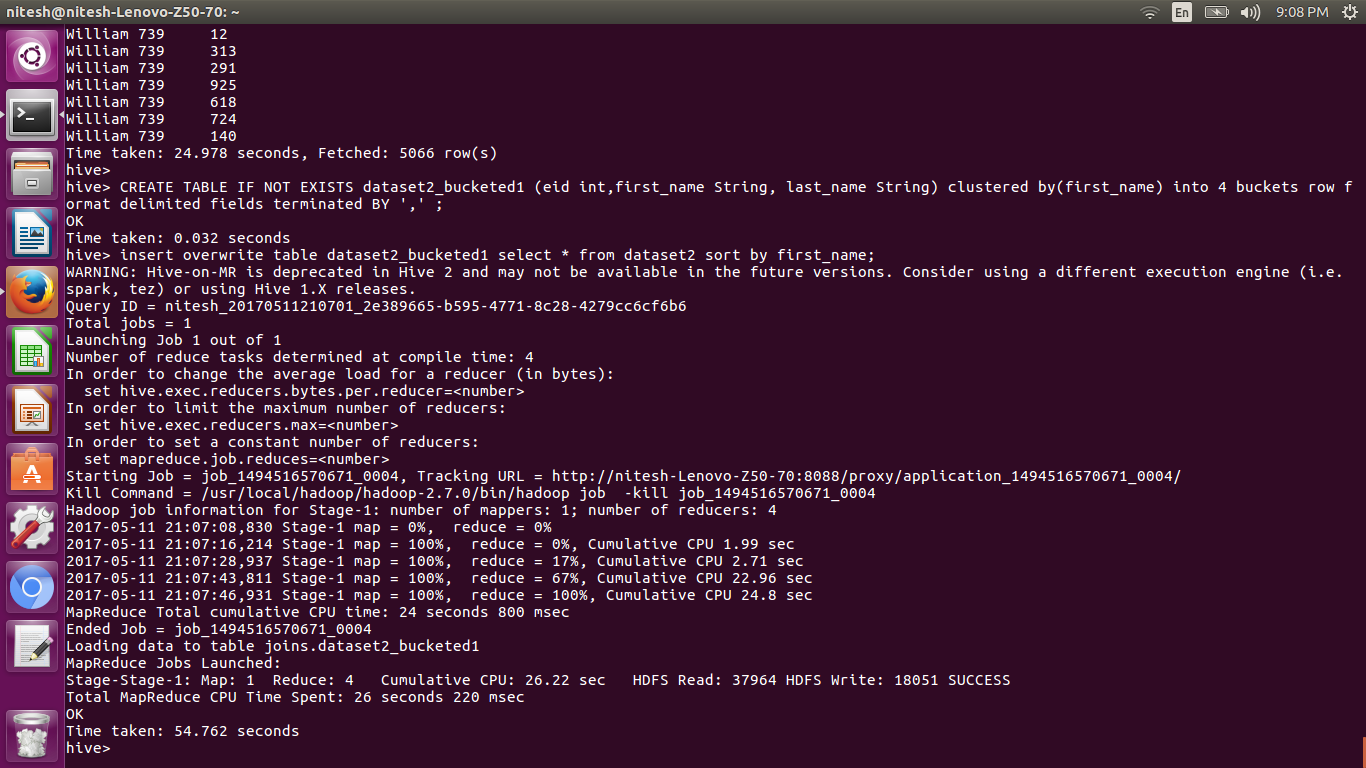


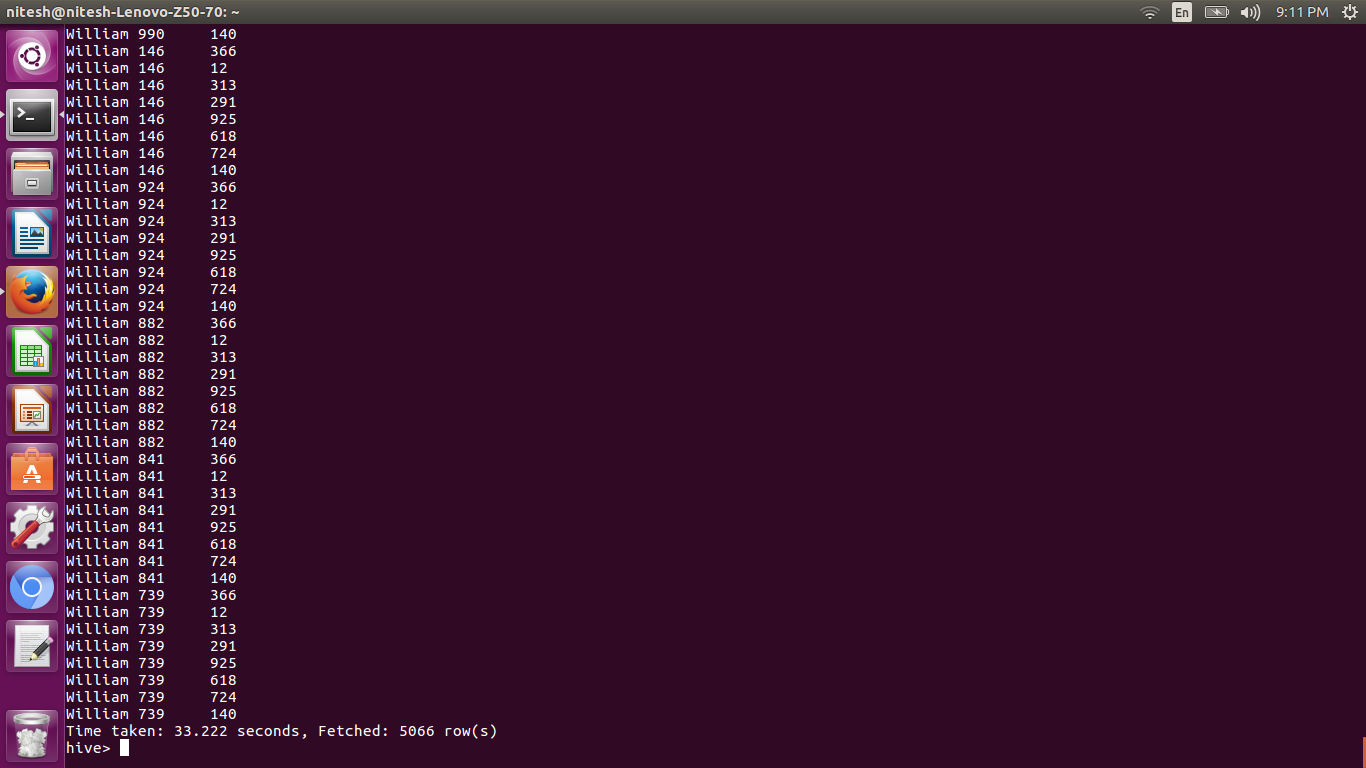


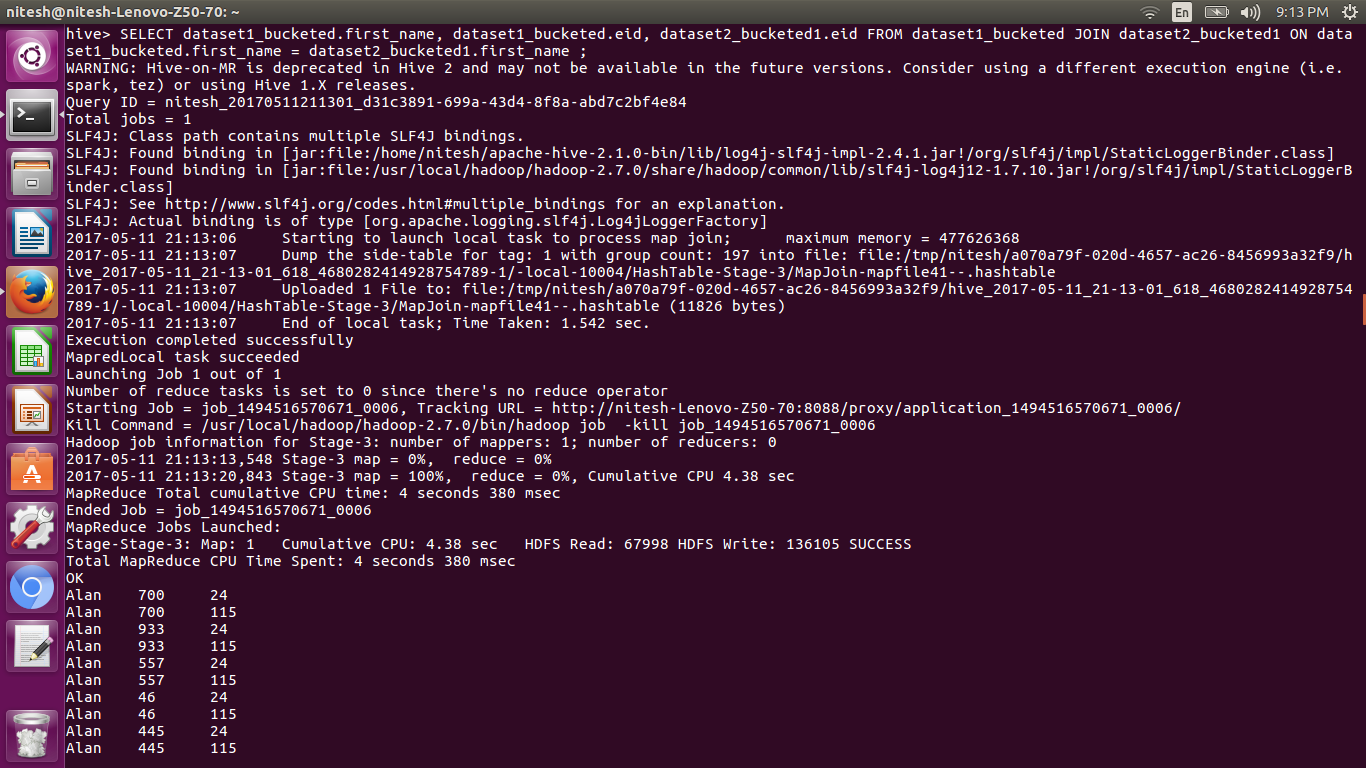
* Sort-Merge Bucket join

hive.auto.convert.join=false;

this command is to be set before the smb join is performed. When we perform smb join mapper come into picture.







* Sort-Merge Bucket Map join

If the tables being joined are sorted and bucketized on the join columns and have the same number of buckets, a sort-merge join can be performed. The corresponding buckets are joined with each other at the mapper.

Here we have 4 buckets for dataset1 and 8 buckets for dataset2. Now, we will create another table with 4 buckets for dataset2.

For performing the SMB-Map join, we need to set the following properties:

Set hive.input.format=org.apache.hadoop.hive.ql.io.BucketizedHiveInputFormat;

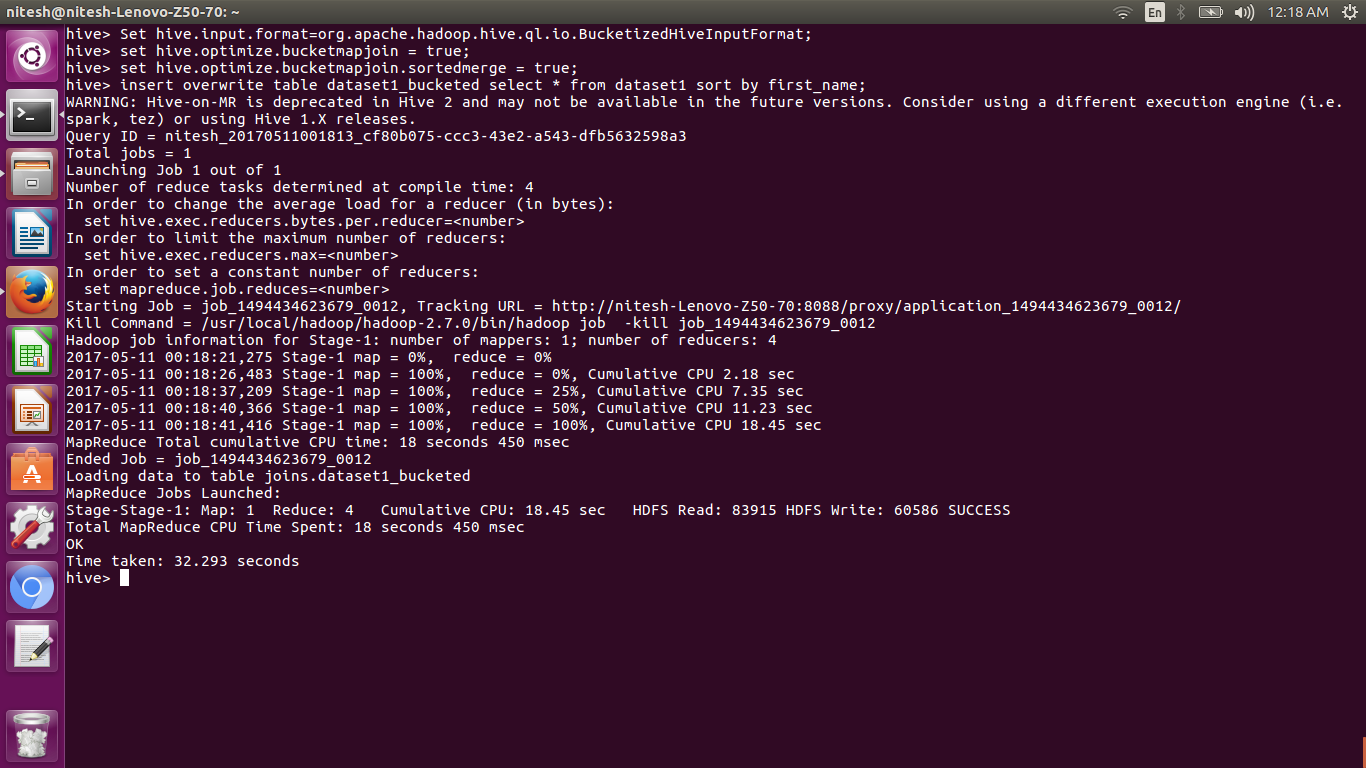
set hive.optimize.bucketmapjoin = true;

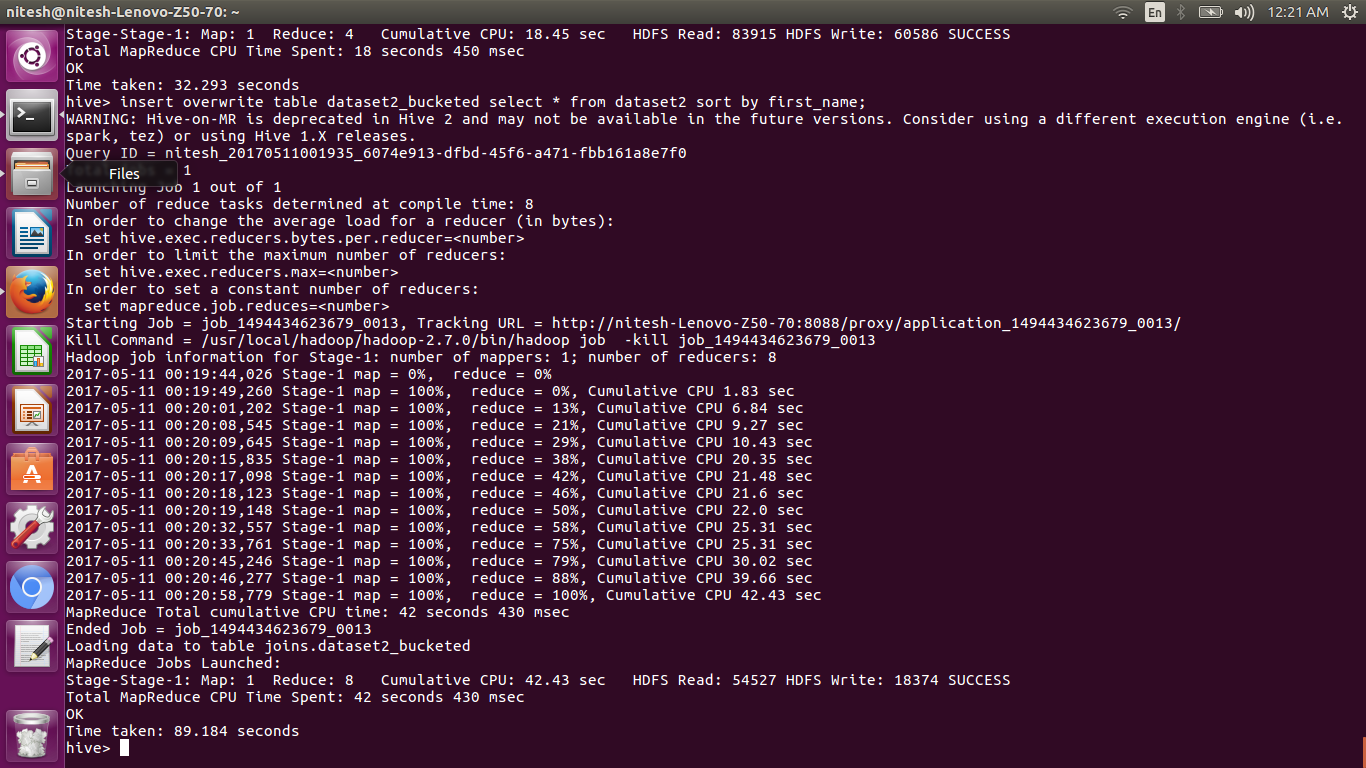
set hive.optimize.bucketmapjoin.sortedmerge = true;

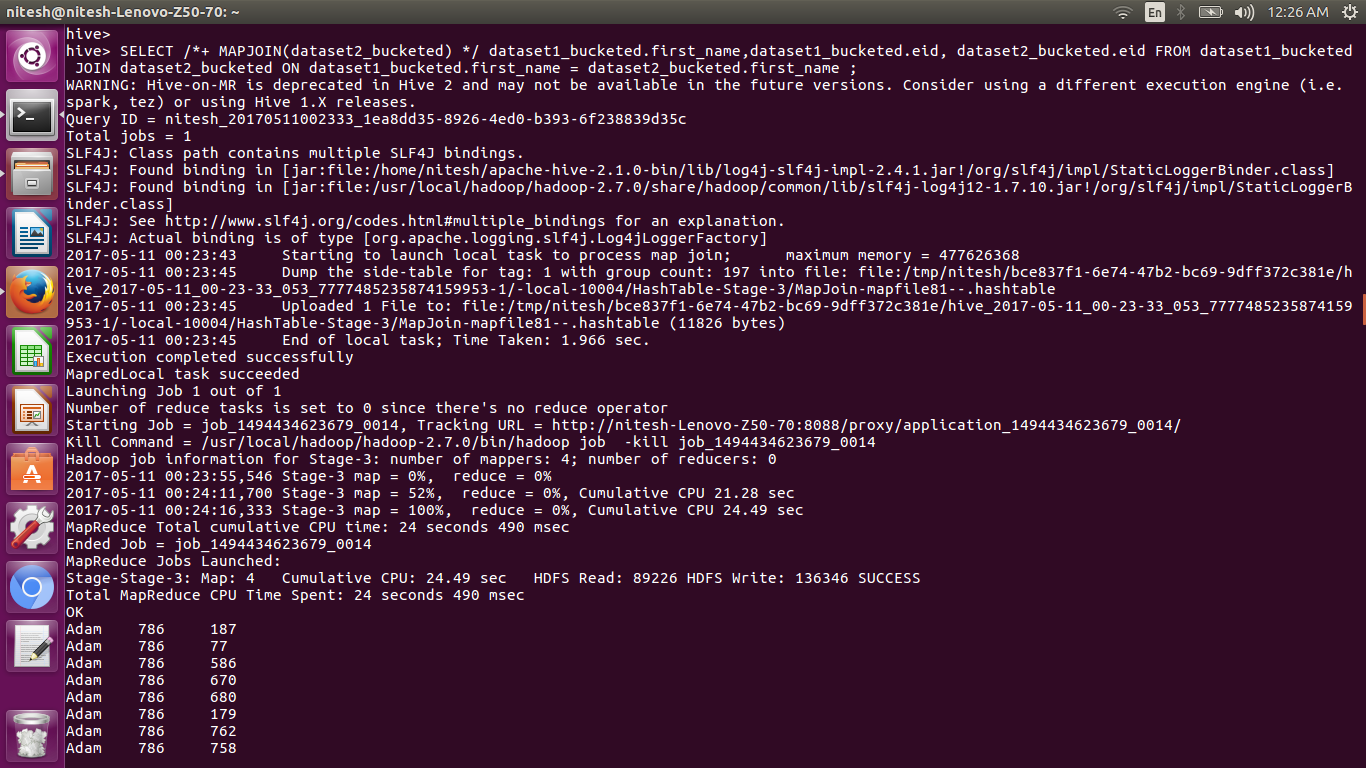
To perform this join, we need to have the data in the bucketed tables sorted by the join column. Now, we will re-insert the data into the bucketed tables by using sorting the records.

insert overwrite table dataset1\_bucketed select \* from dataset1 sort by first\_name;

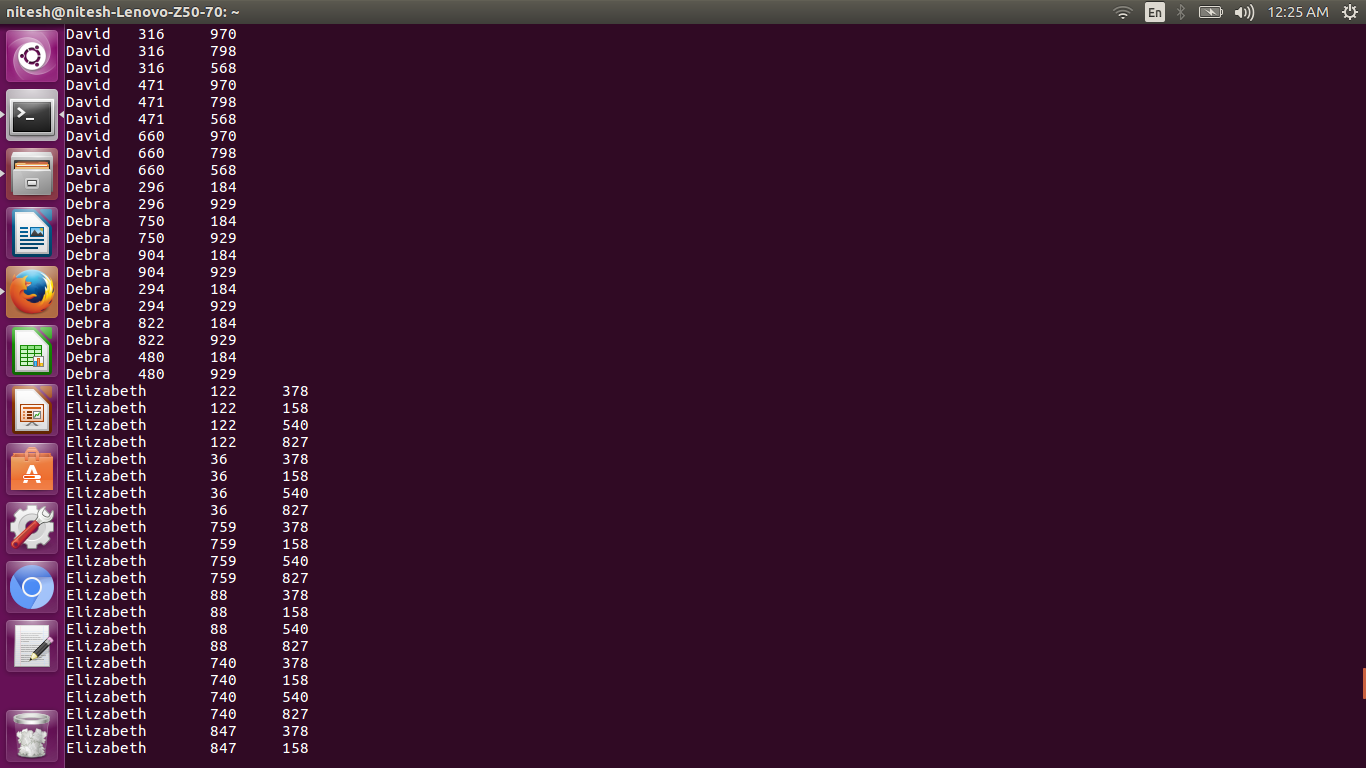
The above command will overwrite the data in the old table and insert the data as per the query. So now the data in the dataset1\_bucketed table is sorted by first\_name. You can see the same in the following screenshot:



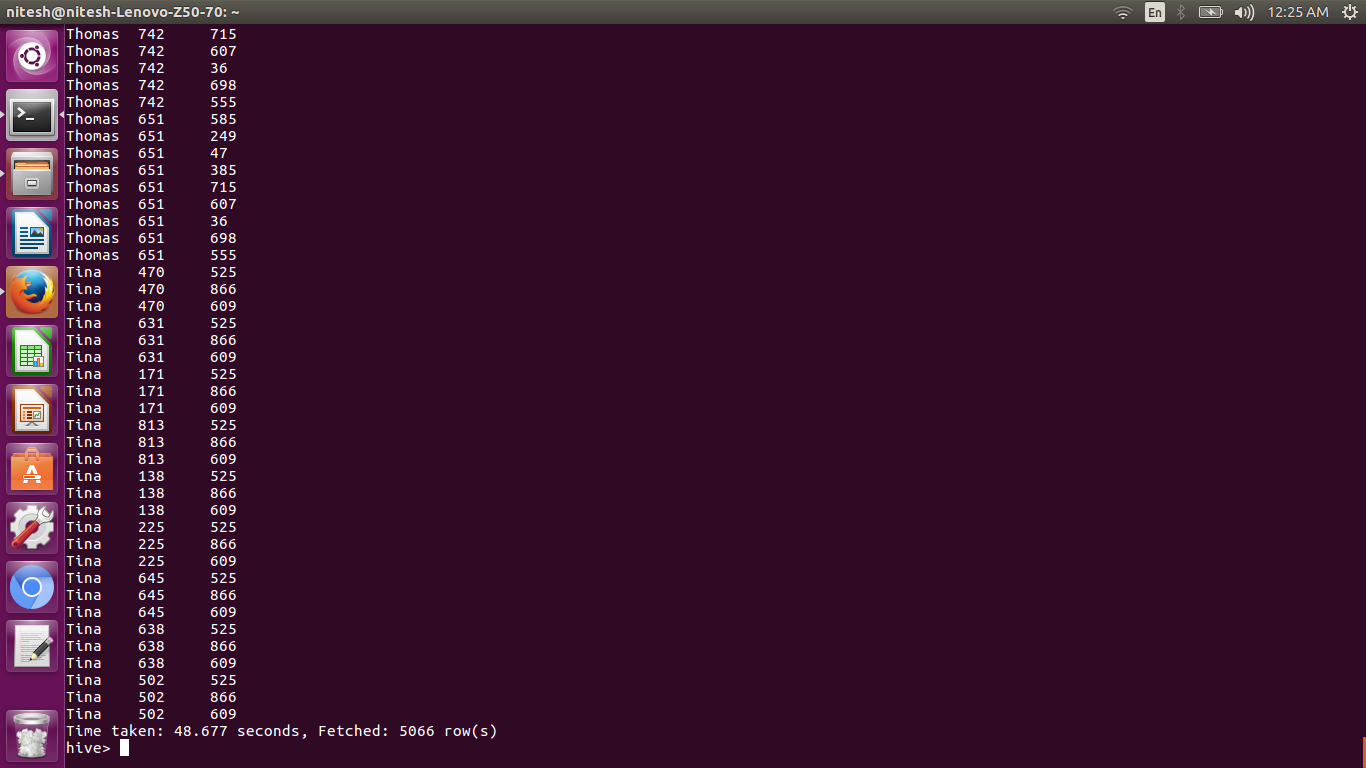


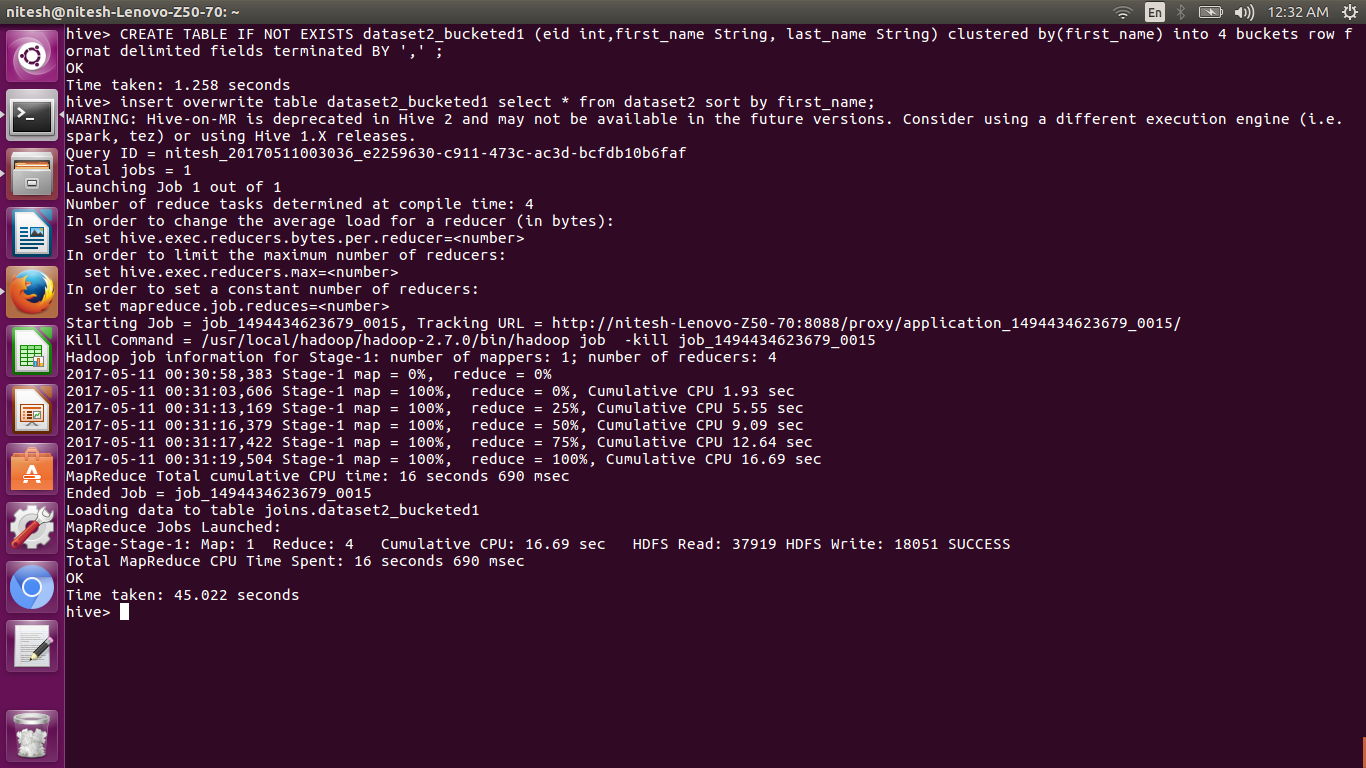


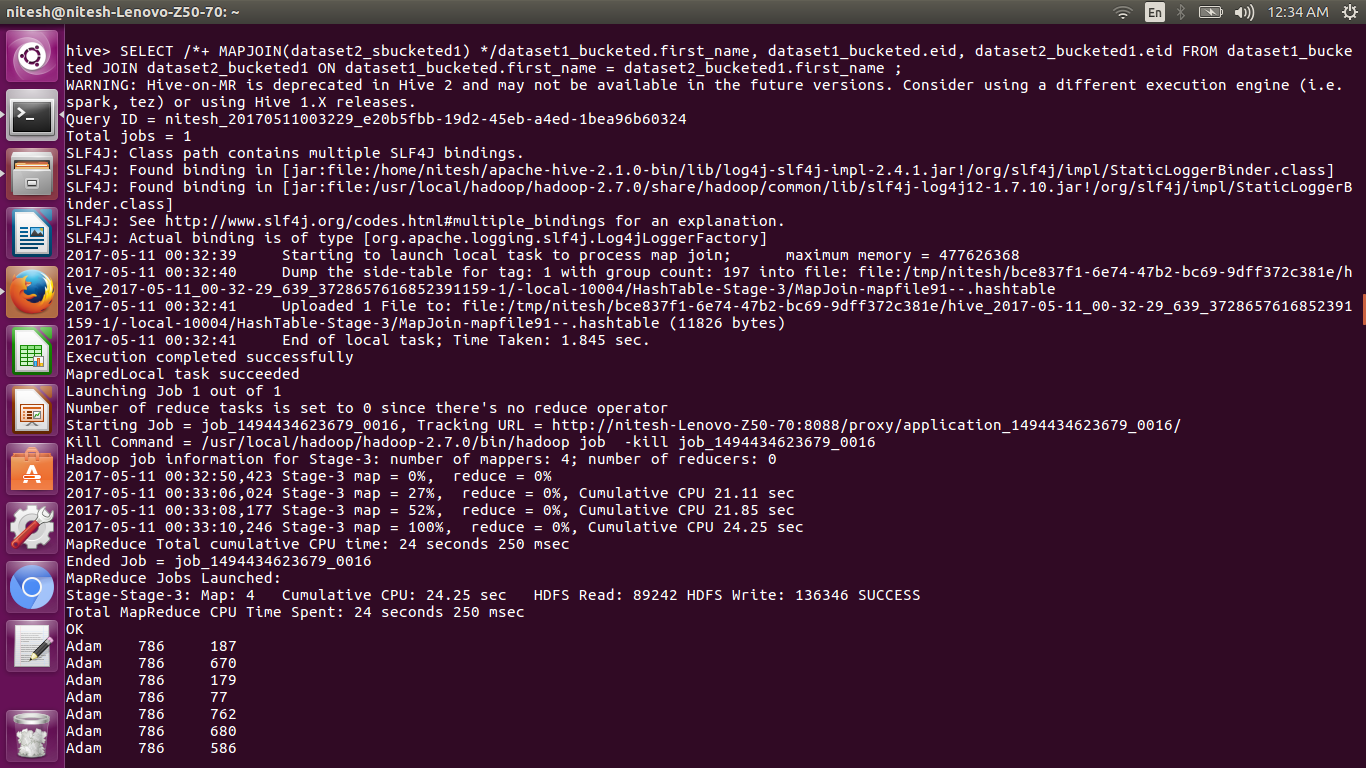






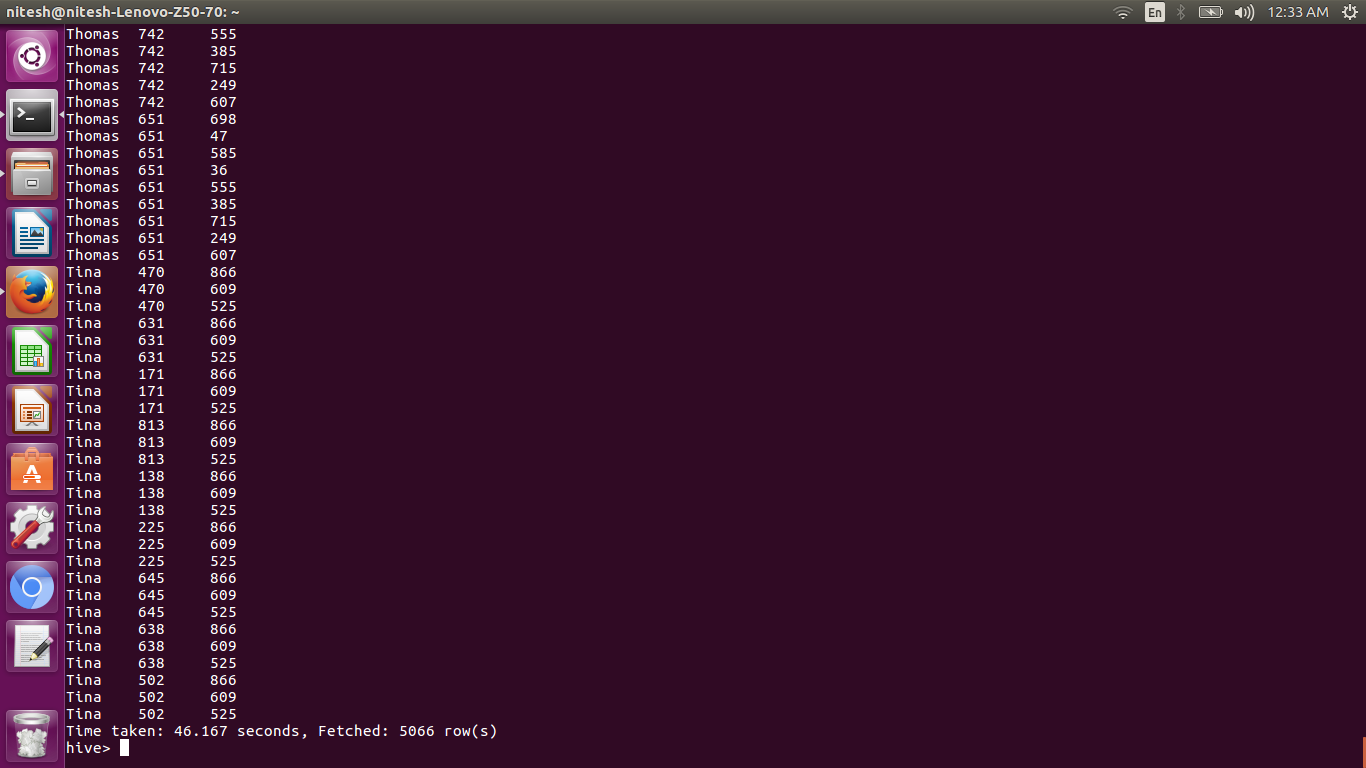












* Left semi join

Left semi join will join only the left table

The restrictions of using LEFT SEMI JOIN are that the right-hand-side table should only be referenced in the join condition (ON-clause), but not in WHERE- or SELECT-clauses etc.

