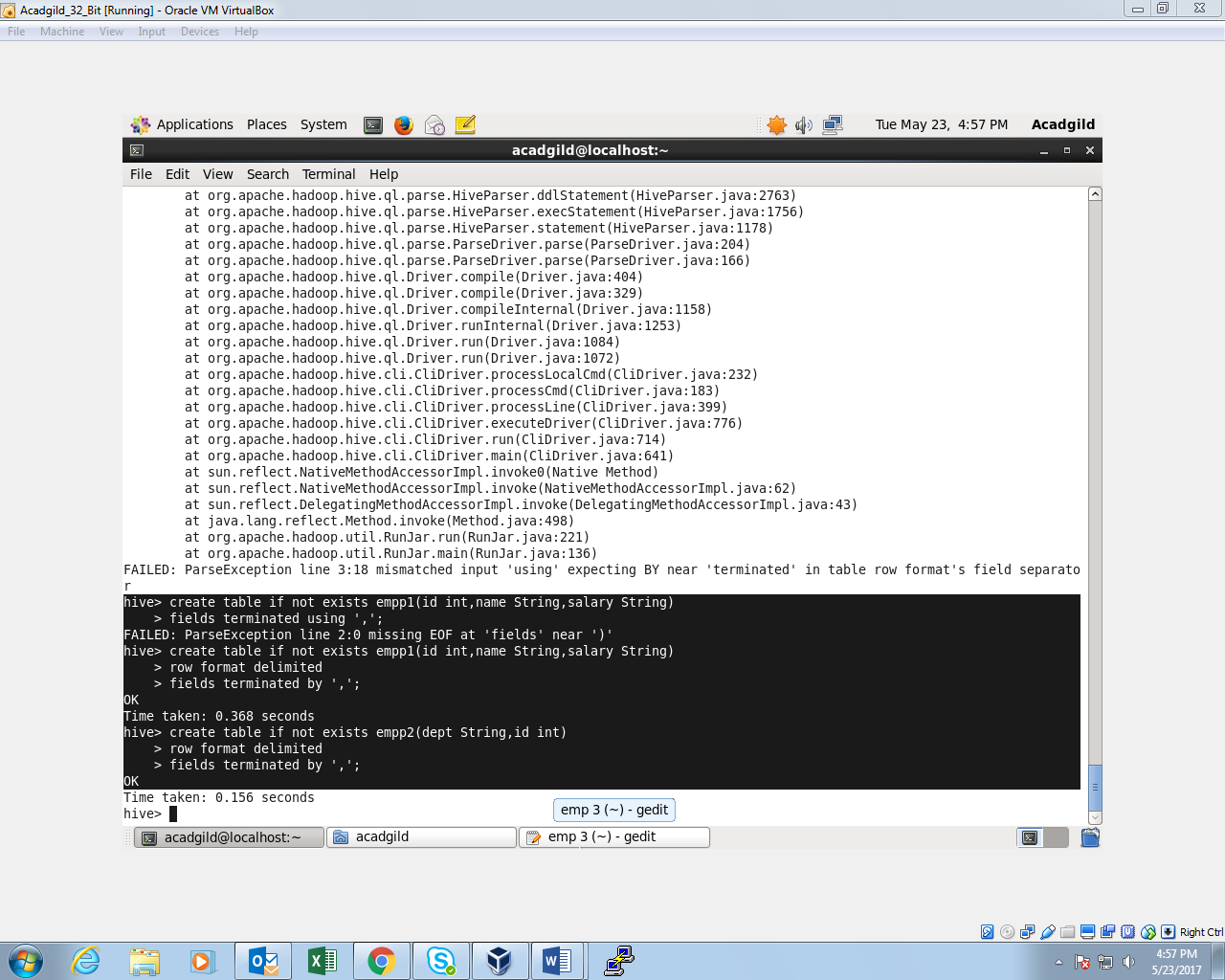
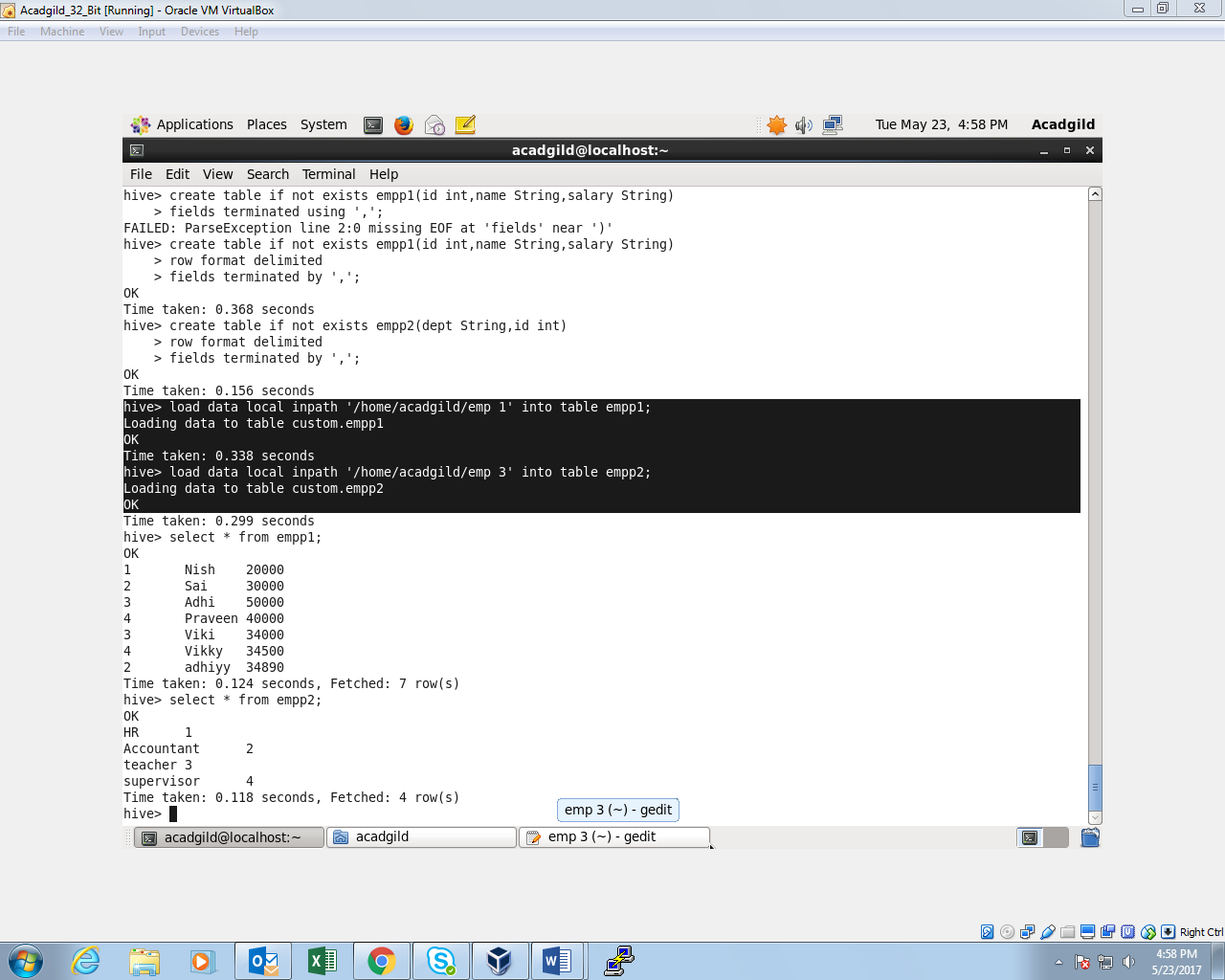
**Perform join optimizations in hive by following the steps in the below blog link**

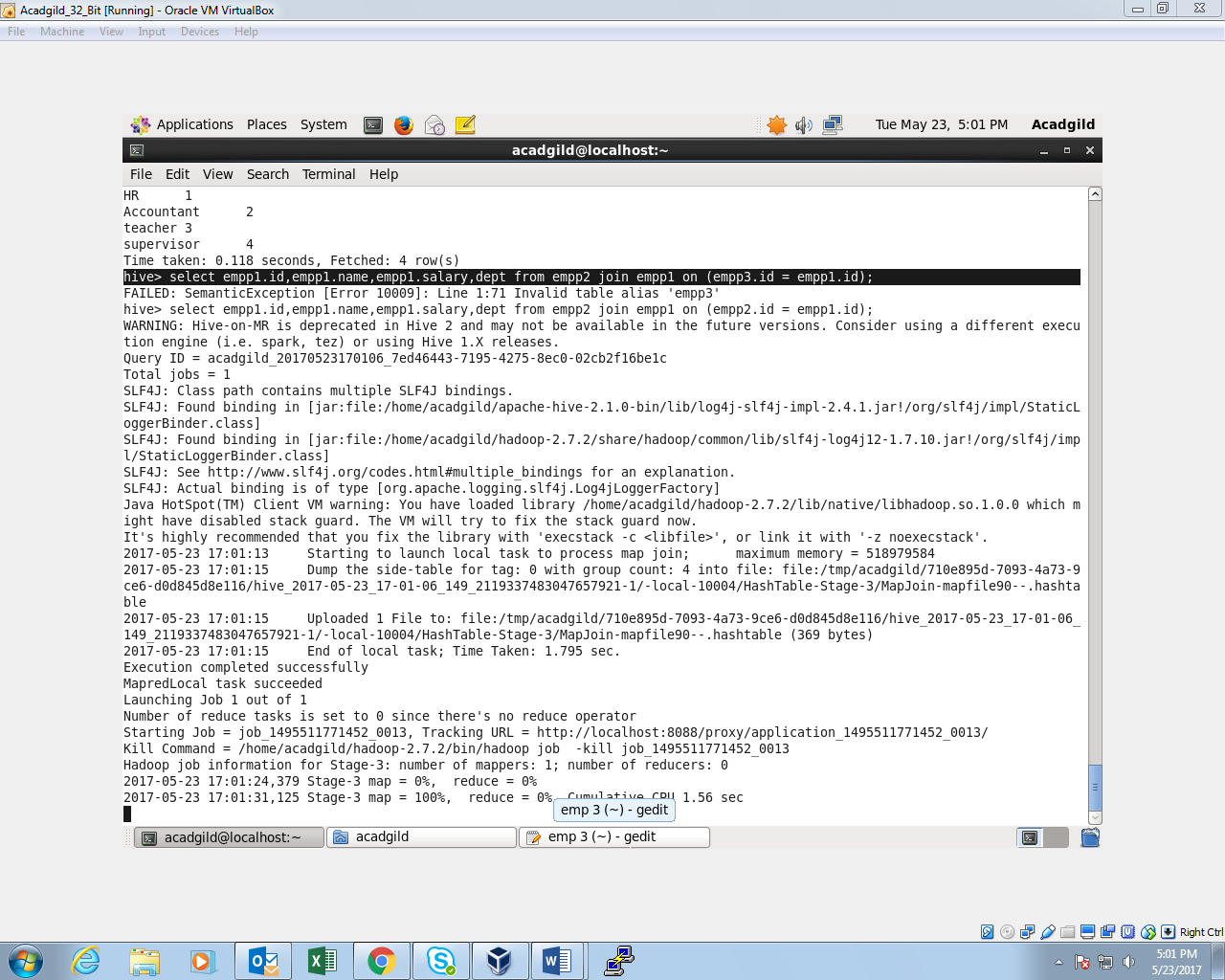
<https://acadgild.com/blog/join-optimization-in-apache-hive/>

Two tables are created and data is feeded

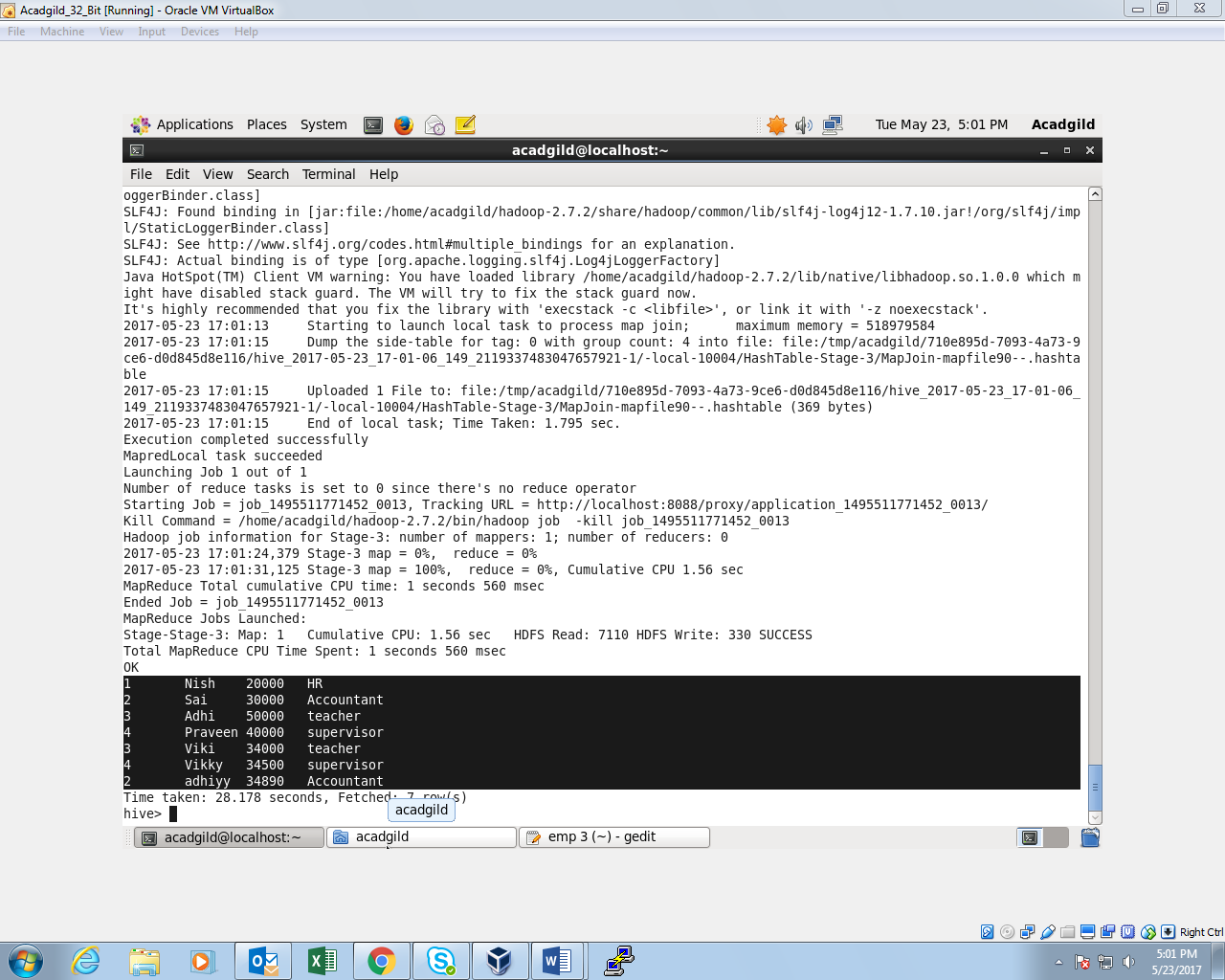




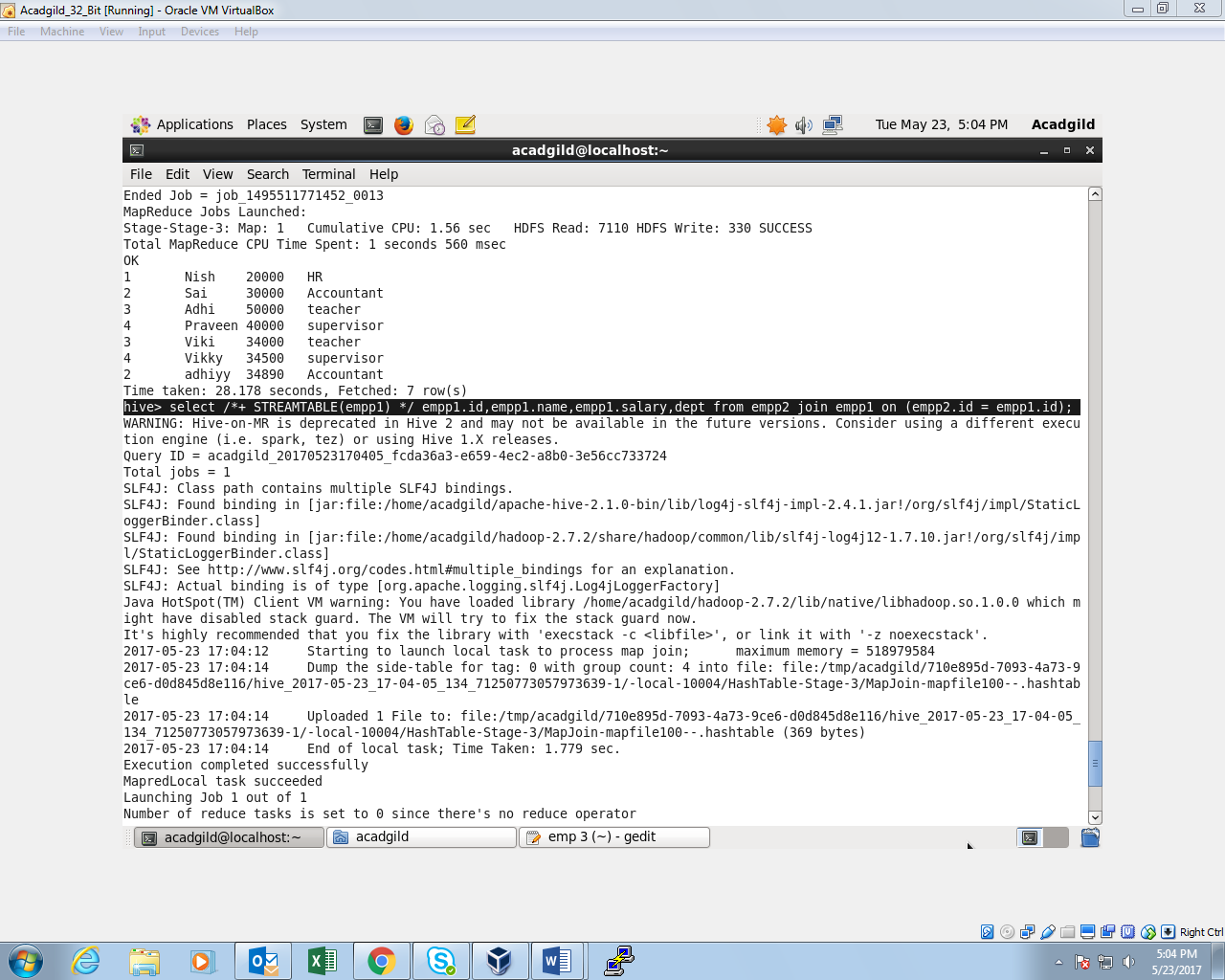
**When Hive executes a join, it needs to select which table is streamed and which table is cached. Hive takes the last table in the JOIN statement for streaming, so we need to ensure that this streaming table is largest among the two.**



**OUTPUT:**



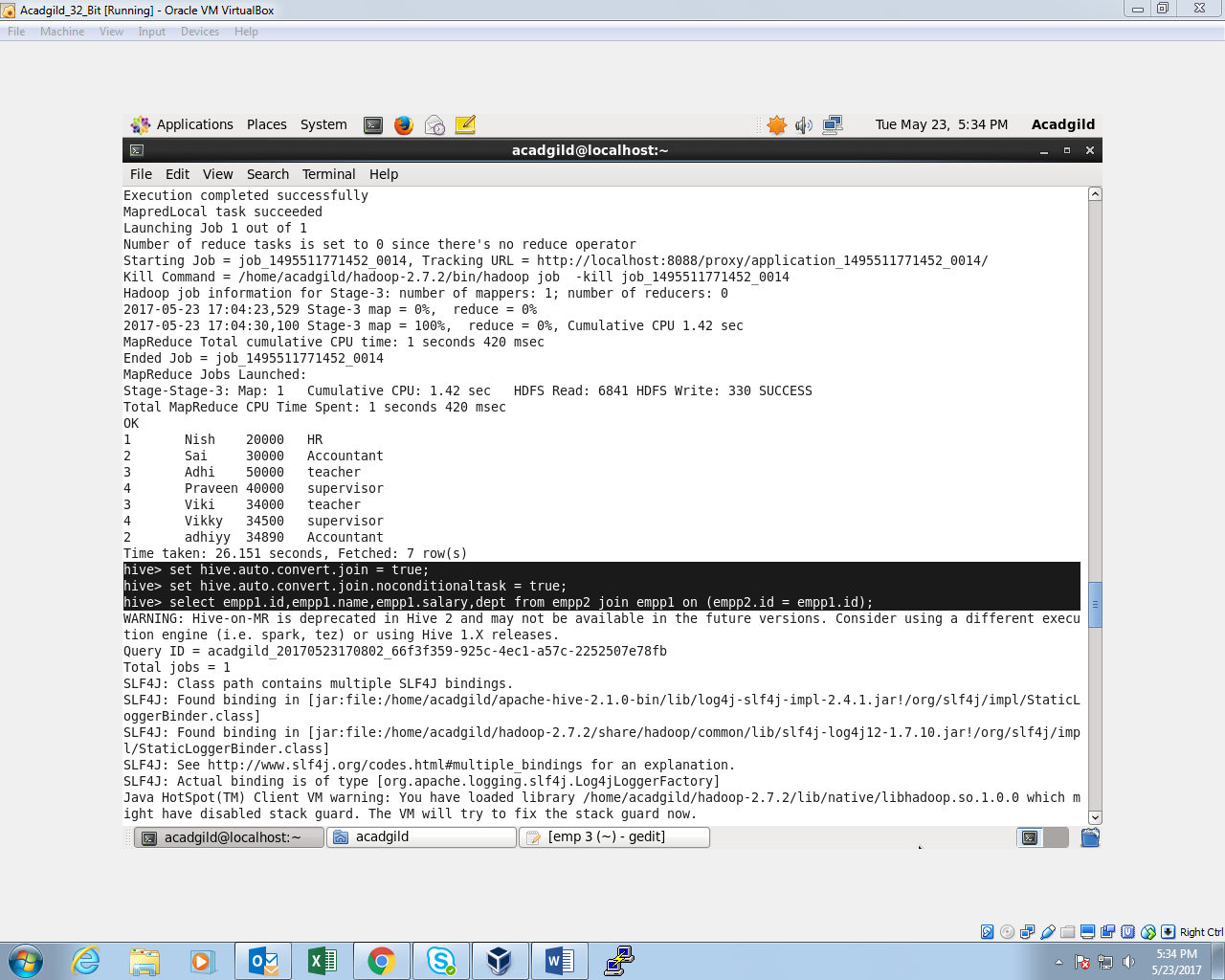
**You can also explicitly tell Hive which table it should stream.**



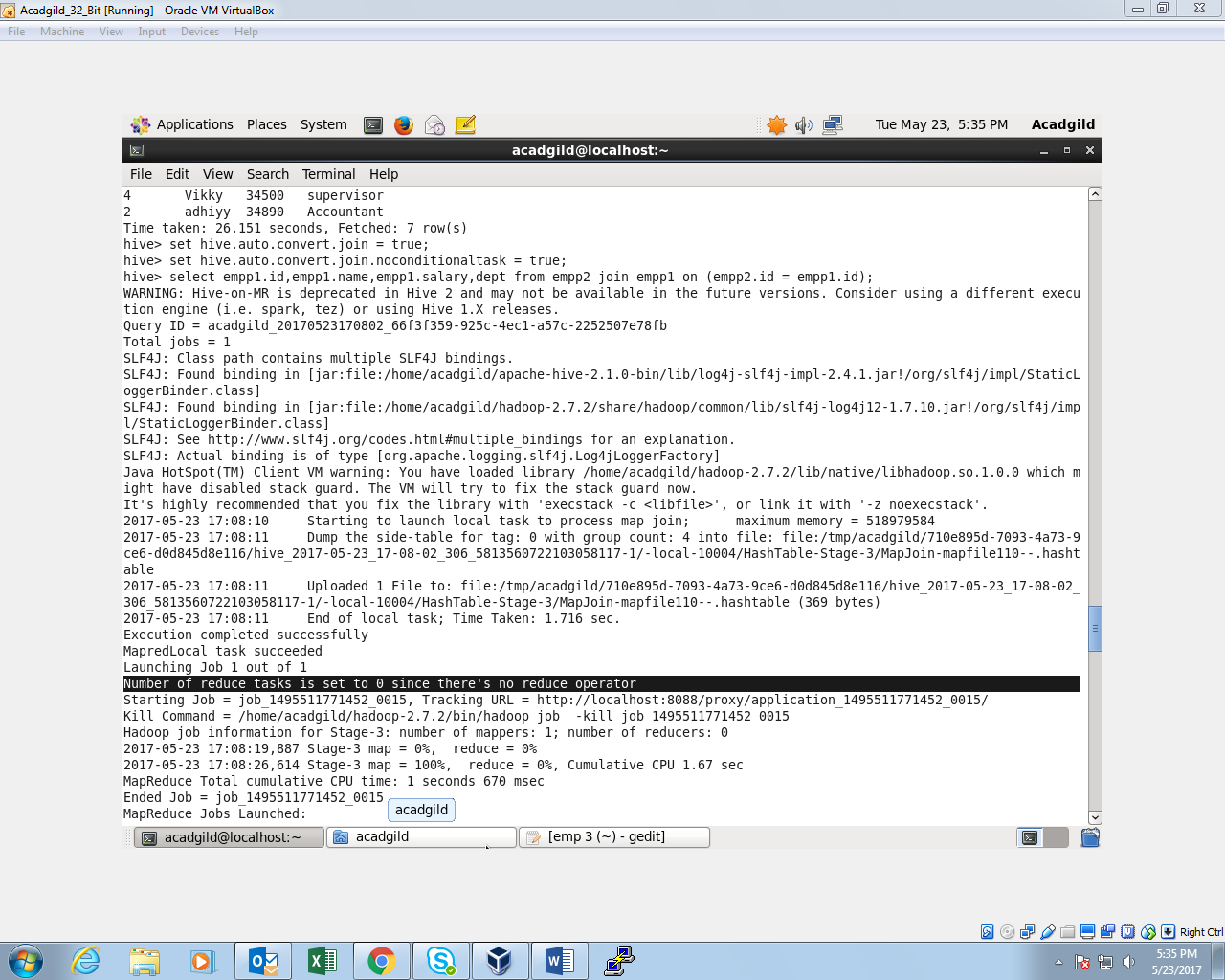
**Map Side Join:**

**Optimizing the hive operation by setting the reducer operation to ‘0’ so only map could run.**

**Setting up the configuration and the command:**



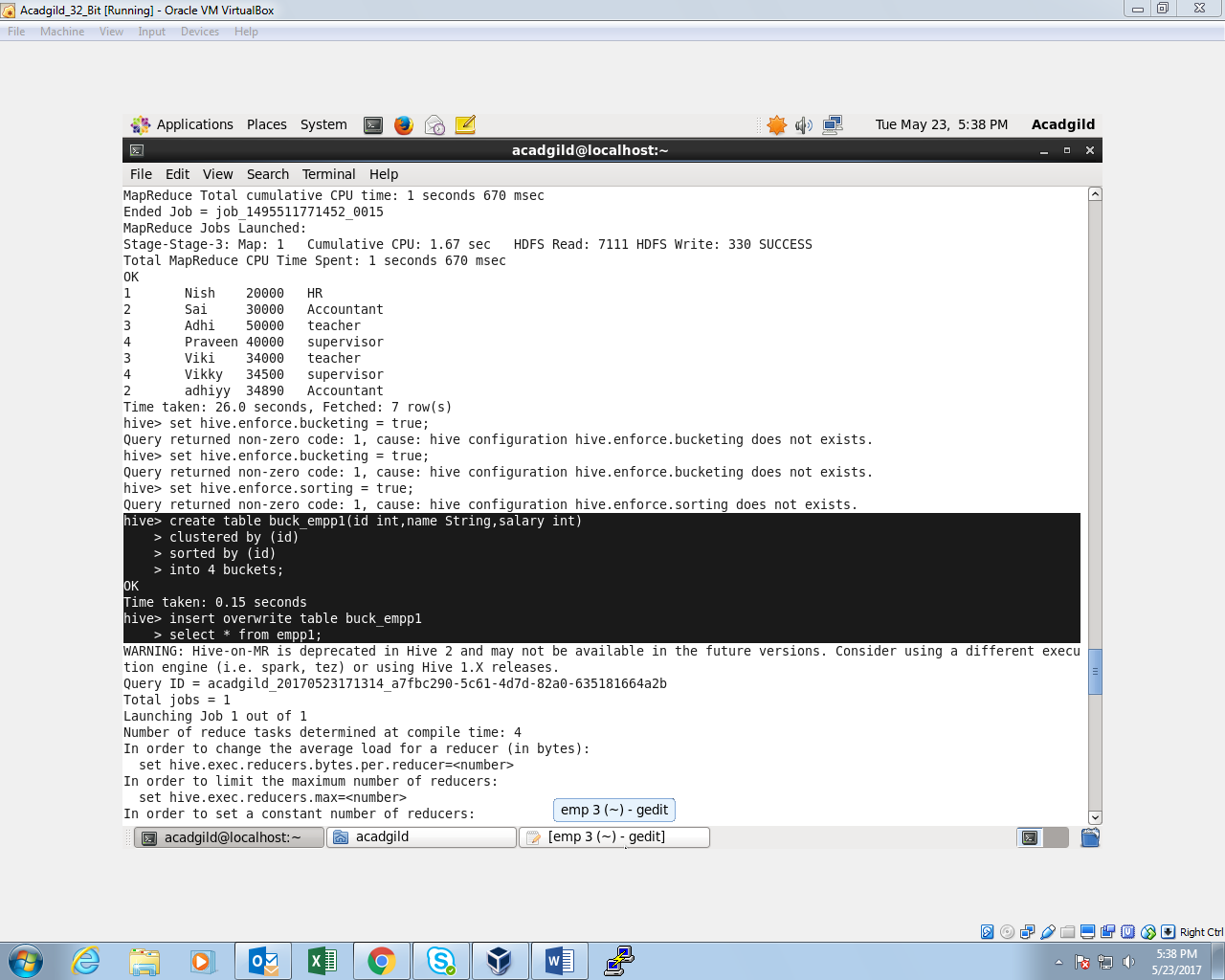
**There are no reducer tasks**

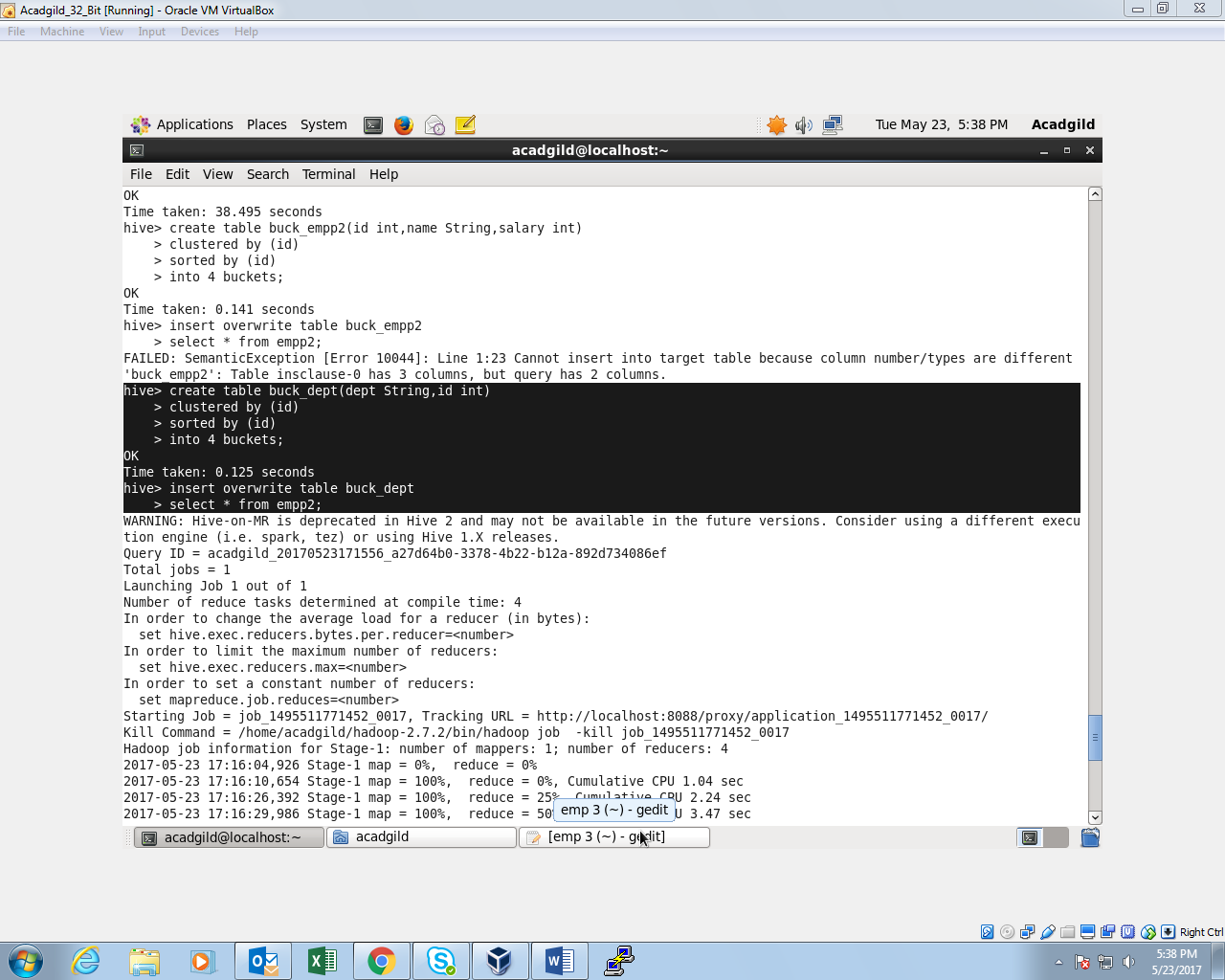


**Sort-Merge-Bucket (SMB) Map Join:**

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.

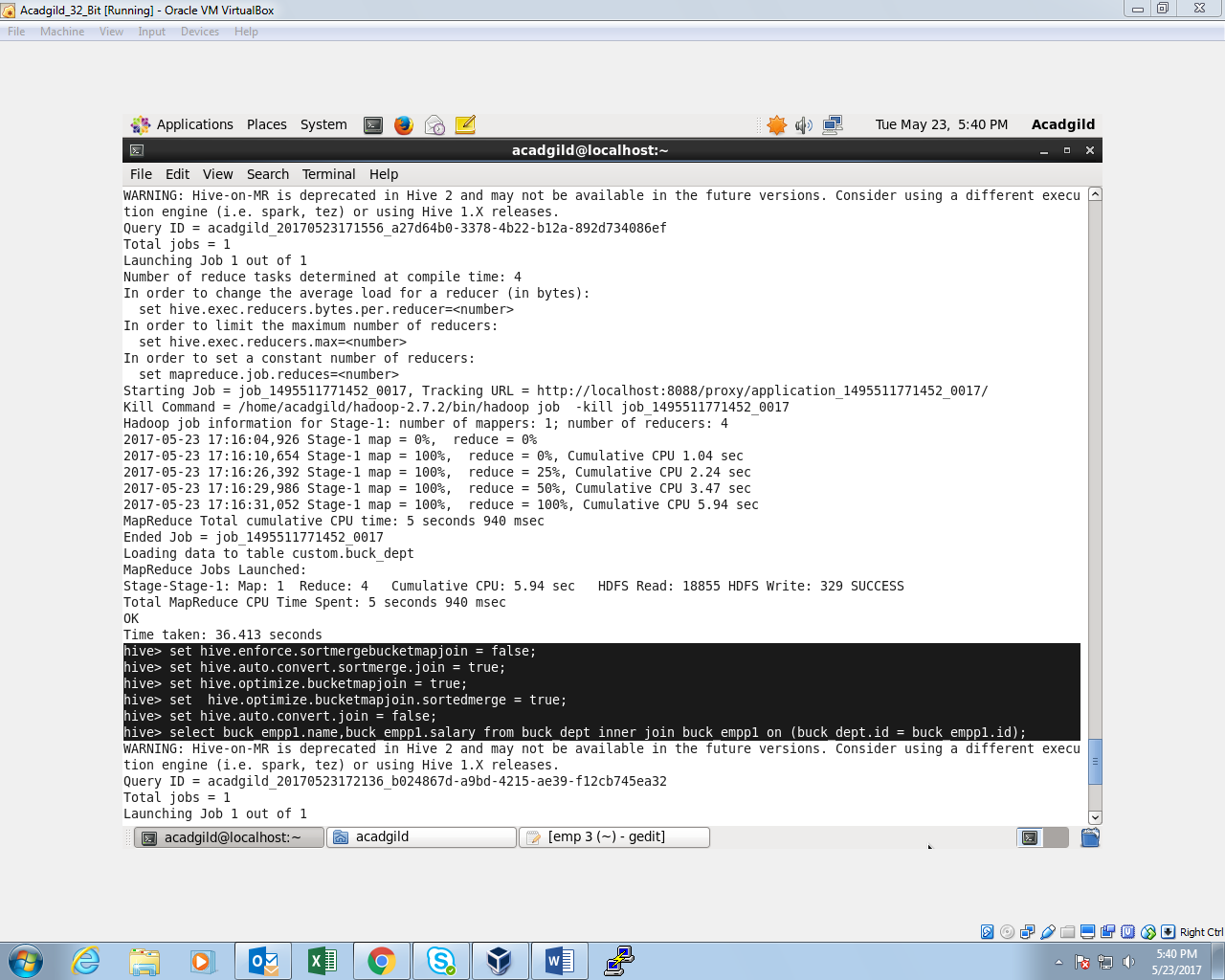
**Once the configuration is setup, two bucketed tables are created**





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.

**Configuration setup and command:**



**Mapper tasks are running (as we had 4 buckets). This helps in performing faster join operation when compared to regular and the output is shown.**

