**PARALLEL PROCESSING HIVE**

Hive querries are converted into job and these will be executed on stages which will be executed one at a time (ie) they are done sequentially say there are 2 jobs first job 1 run and its execution will complete in stages that also will be done sequentially .

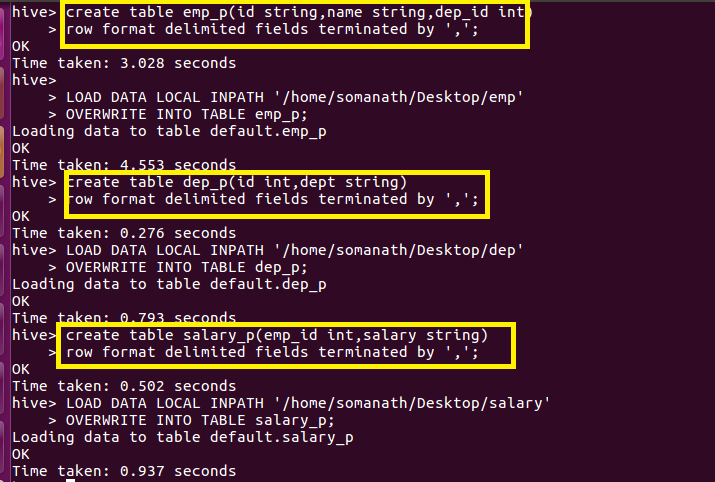
These Stages may be  a Map stage,[Reduce stage](https://acadgild.com/big-data/big-data-development-training-certification), a sampling stage, a merge stage, a limit stage, or other possible tasks Hive needs to do.

Thus what happens is that if we are executing a complex queries it will be done in a sequence which results in increased time for the completion of the job

But if we have stages that are not dependent on each other we can make it execute in a parallel manager so that it results in faster completion of job

To understand this we will take a scenario as below,

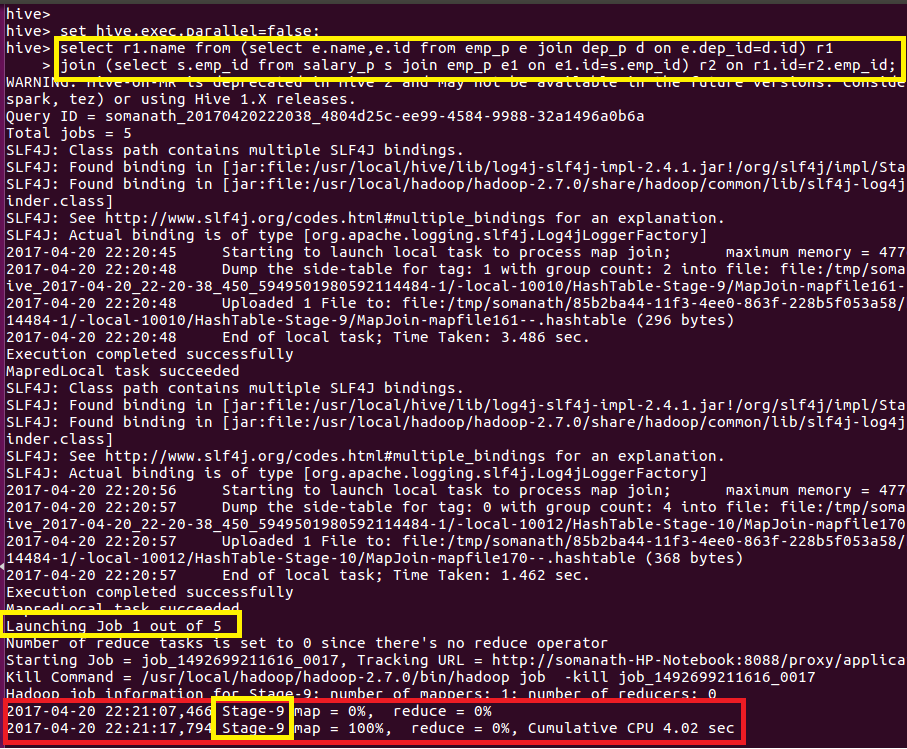
**We are creating 3 tables employee,salary,department**



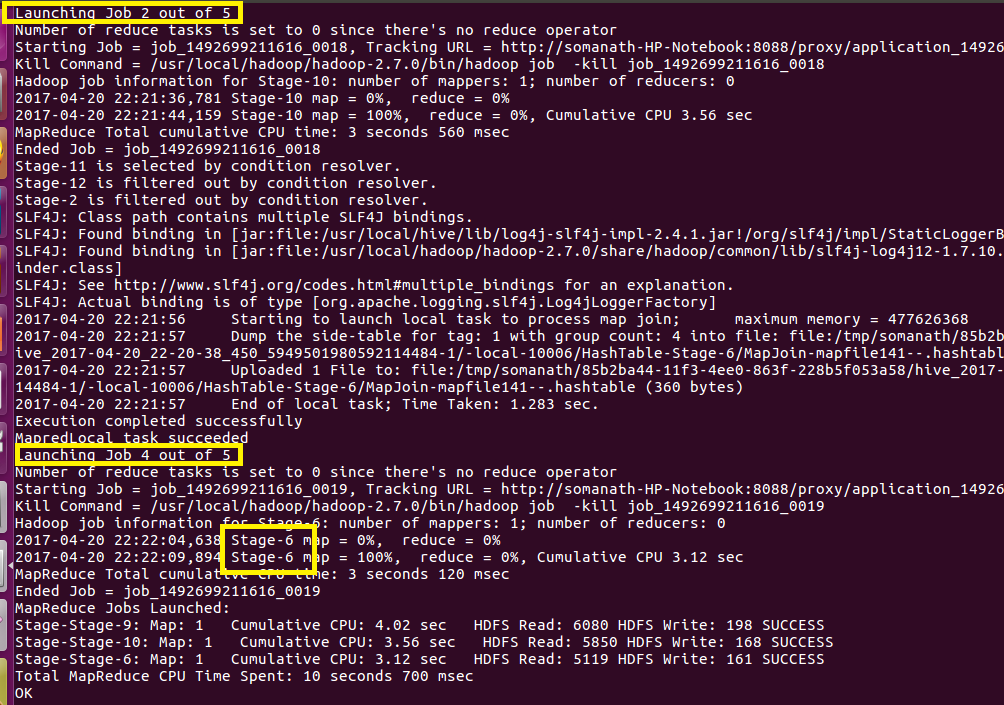
Now we want join these 3 table and display the employee name using the below querry

I am using a nested querry where I am Joining employee and Department table separately as ri and I am Joining Employee and salary table separately as r2 and atlast joining both r1 and r2

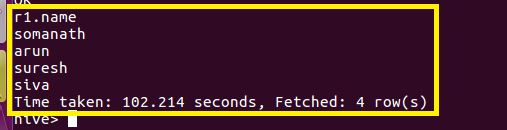
The Thing to note is that the above two joins are independent of each other so that they can be performed in parallel but here they are performed sequentially as **job 1 executed first and stage 9 is alone executed as shown**



**And after job 1 is completed Job2 runs and so on**



Output



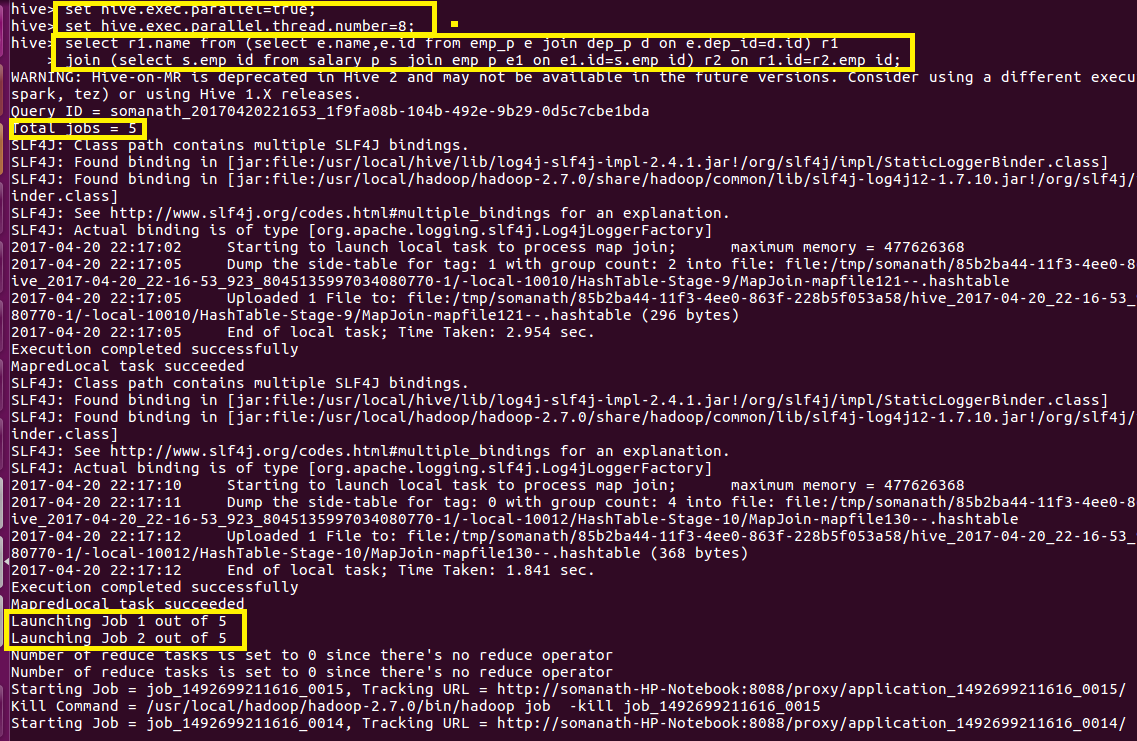
**Now we can execute the same querry in parallel**

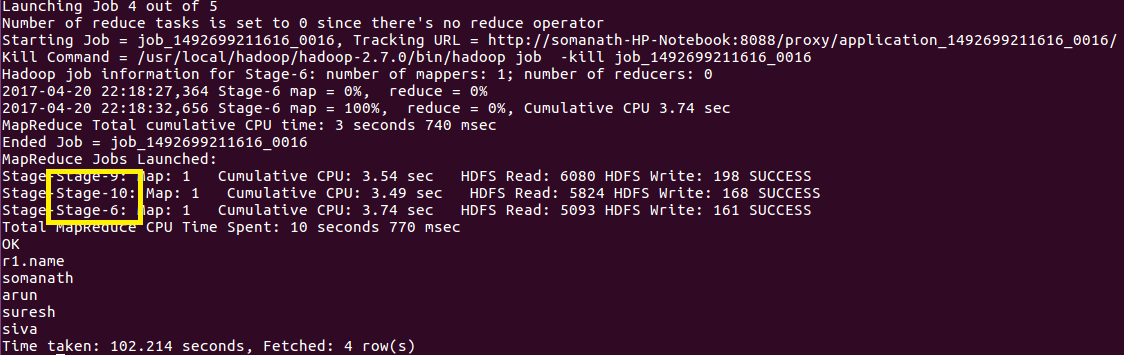
In order to run the job in parallel

**Set hive.exec.parallel=true**

**Set hive.exec.parallel.thread.number=true in hive terminal as shown and running the same join**

**We can see that job are executed in parallel and staging is also done in parallel as shown**

****

****