**Hive Mini Project 1**

**1. Create a schema based on the given dataset**

**hive> create table Agent\_login\_report**

**> (**

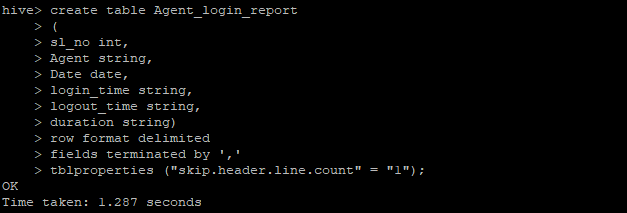
**> sl\_no int, Agent string, Date date, login\_time string, logout\_time string, duration string**

**> )**

**> row format delimited**

**> fields terminated by ','**

**> tblproperties ("skip.header.line.count" = "1");**

****

**hive> create table Agent\_Performance**

**> (**

**> sl\_no int, Date date, Agent\_name string, total\_chats int, average\_response\_time string,**

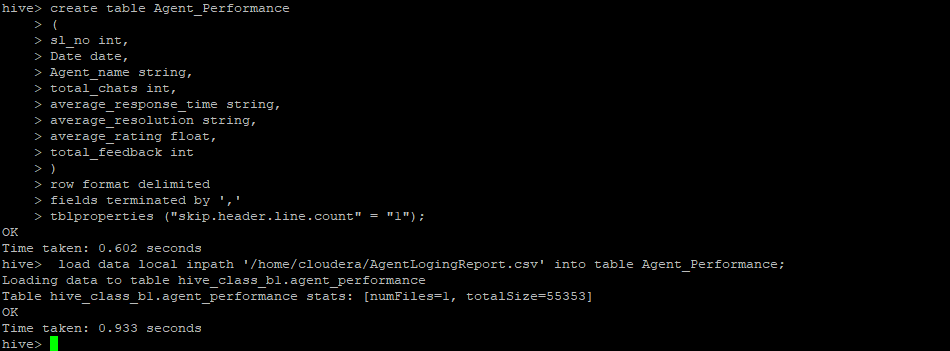
**> average\_resolution string, average\_rating float, total\_feedback int**

**> )**

**> row format delimited**

**> fields terminated by ','**

**> tblproperties ("skip.header.line.count" = "1");**

****

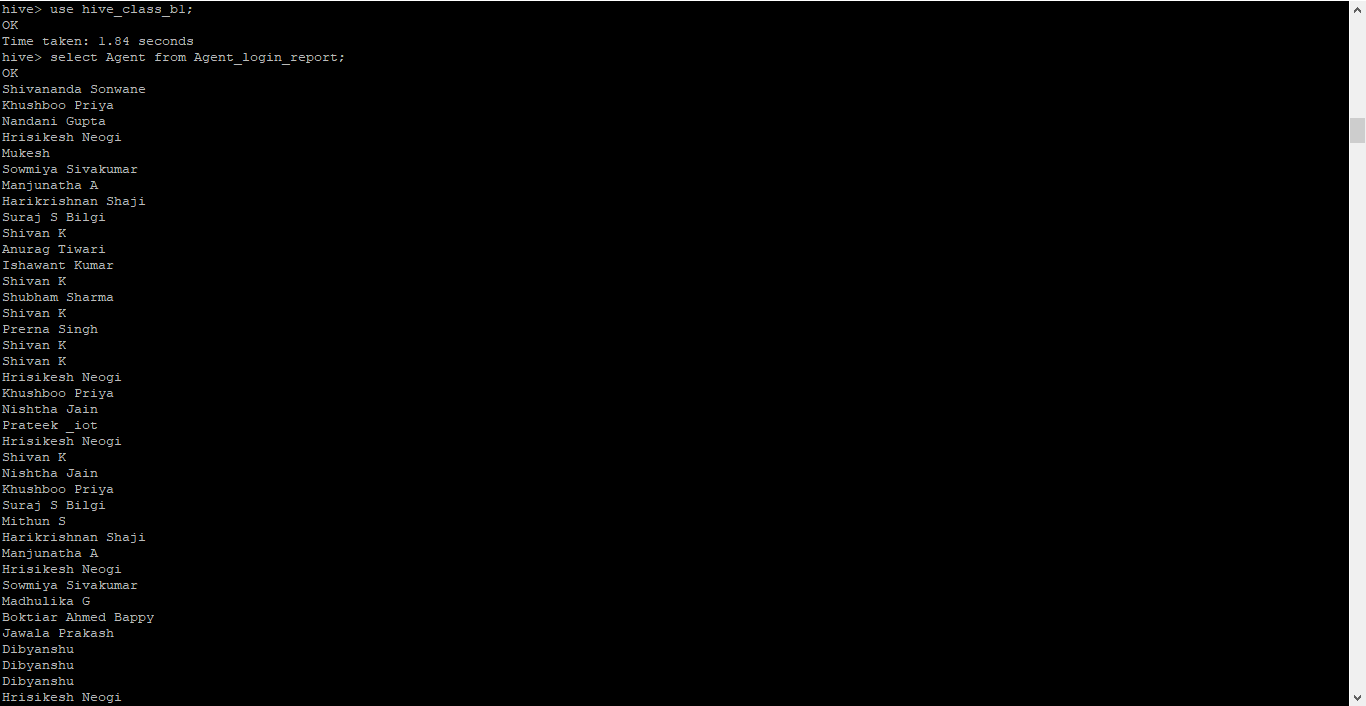
**2. Dump the data inside the hdfs in the given schema location.**

hive> load data local inpath '/home/cloudera/AgentLogingReport.csv' into table Agent\_login\_report;

hive> load data local inpath '/home/cloudera/AgentLogingReport.csv' into table Agent\_Performance;

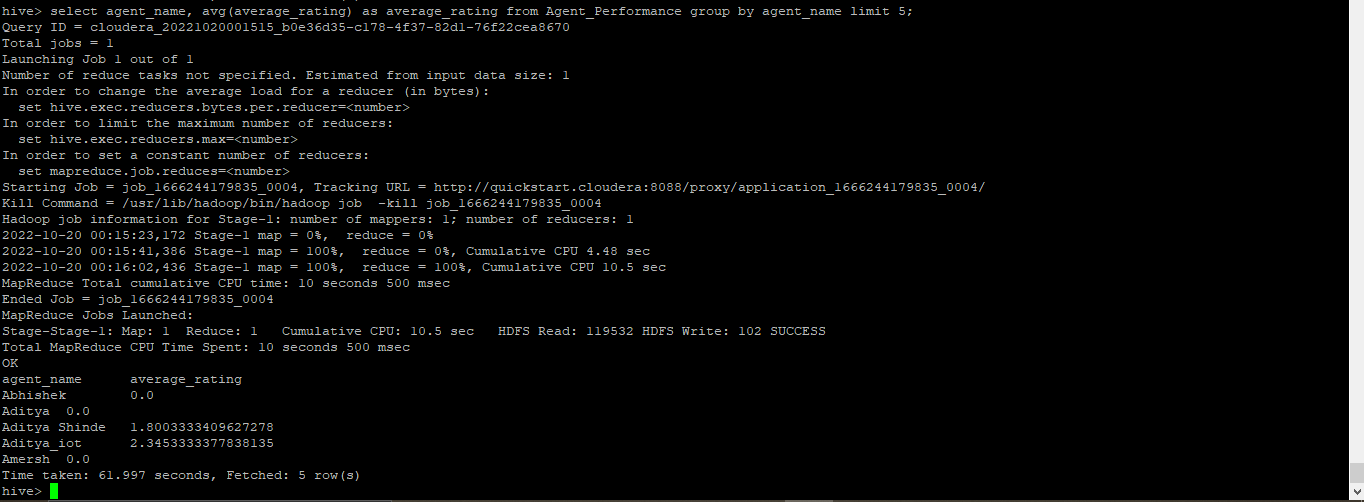
**3. List of all agents' names.**

Hive> select Agent from Agent\_login\_report;

****

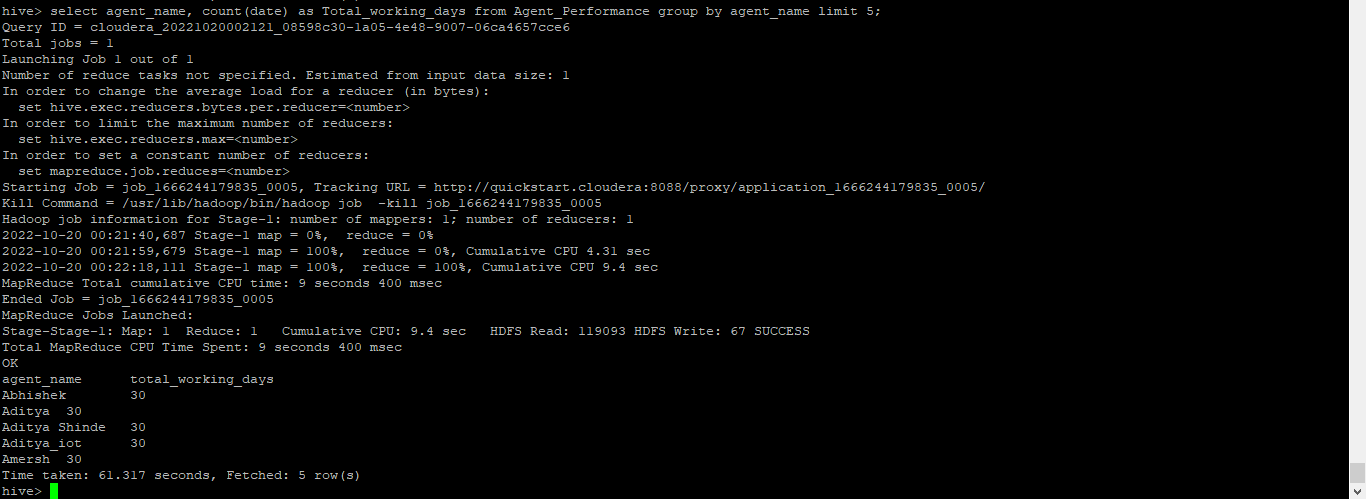
**4. Find out agent average rating.**

hive> select agent\_name, avg(average\_rating) as average\_rating from Agent\_Performance group by agent\_name limit 5;

****

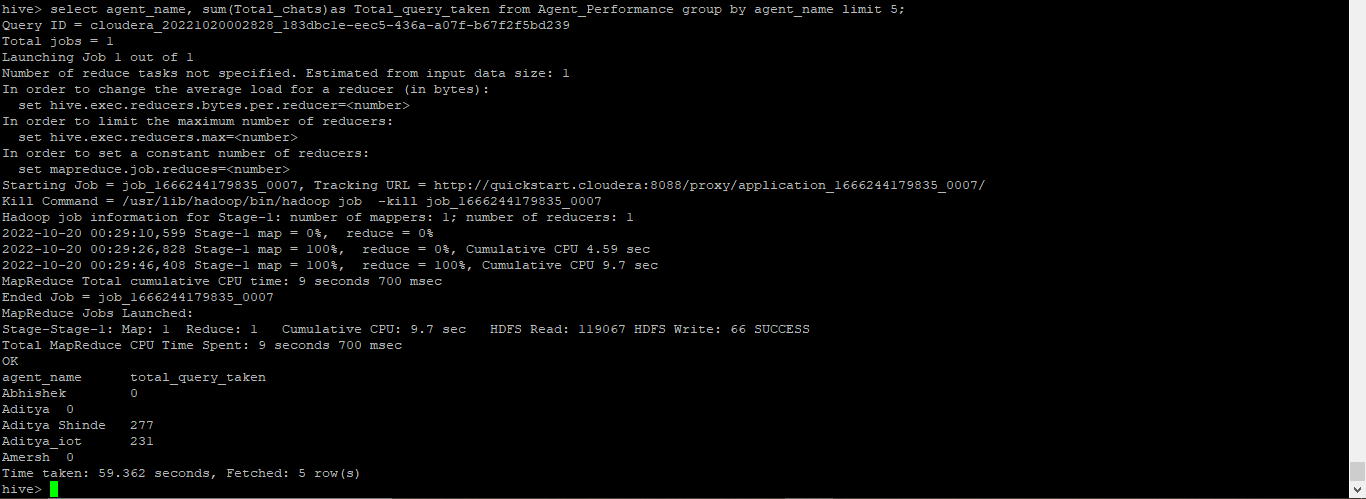
**5. Total working days for each agents**

hive> select agent\_name, count(date) as Total\_working\_days from Agent\_Performance group by agent\_name limit 5;

****

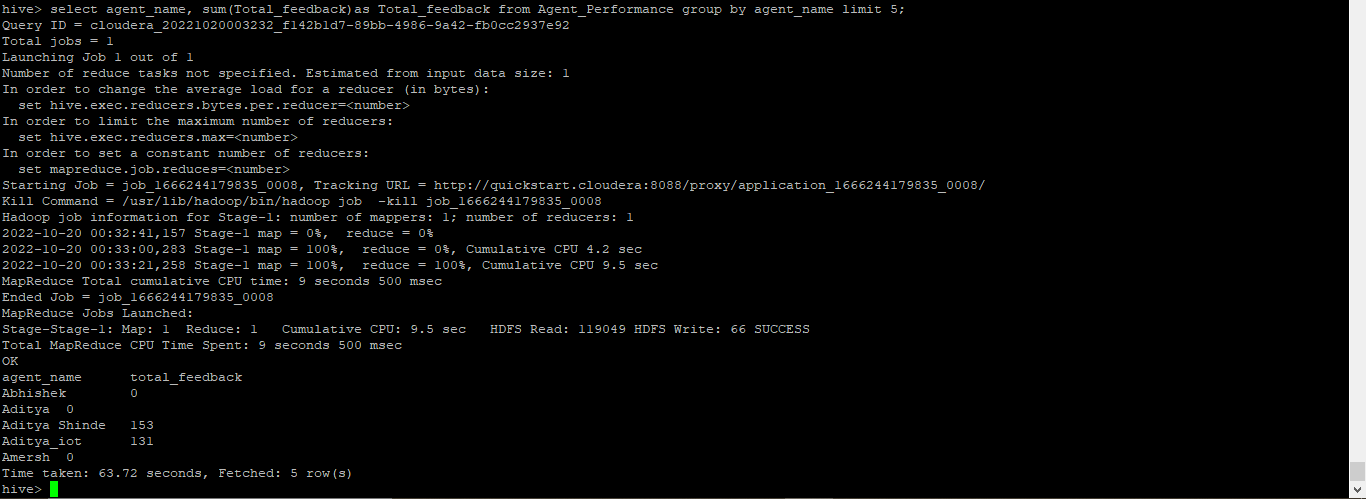
**6. Total query that each agent have taken**

hive> select agent\_name, sum(Total\_chats)as Total\_query\_taken from Agent\_Performance group by agent\_name limit 5;

****

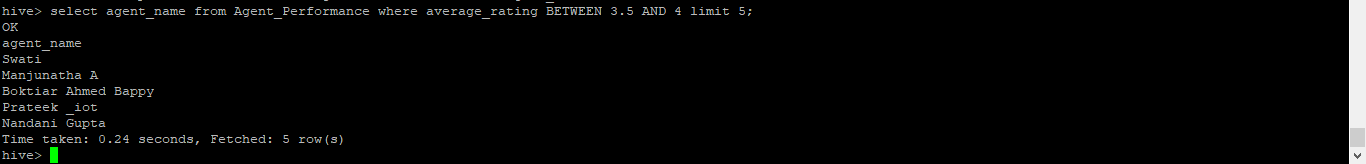
**7. Total Feedback that each agent have received**

hive> select agent\_name, sum(Total\_feedback)as Total\_feedback from Agent\_Performance group by agent\_name limit 5;

****

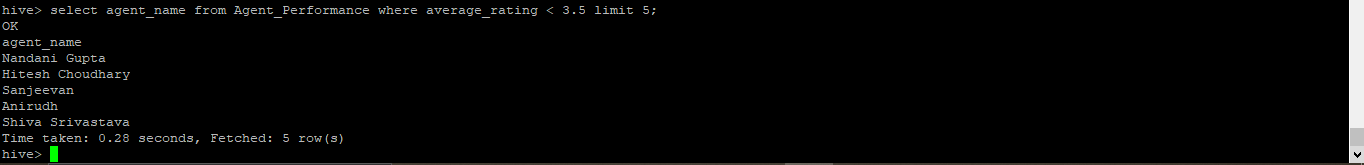
**8. Agent name who have average rating between 3.5 to 4**

hive> select agent\_name from Agent\_Performance where average\_rating BETWEEN 3.5 AND 4 limit 5;

****

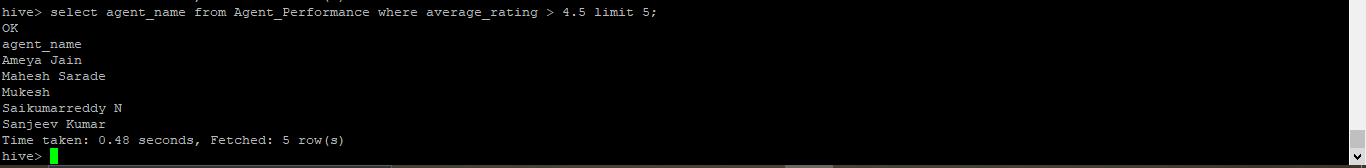
**9. Agent name who have rating less than 3.5**

hive> select agent\_name from Agent\_Performance whereaverage\_rating < 3.5 limit 5;

****

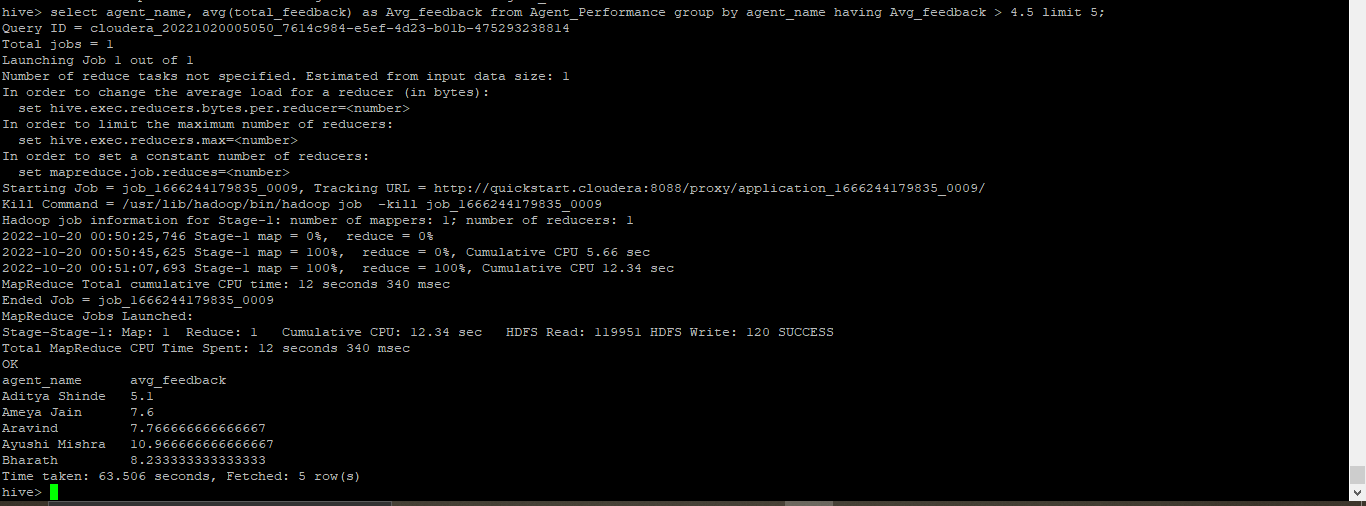
**10. Agent name who have rating more than 4.5**

hive> select agent\_name from Agent\_Performance where average\_rating > 4.5 limit 5;

****

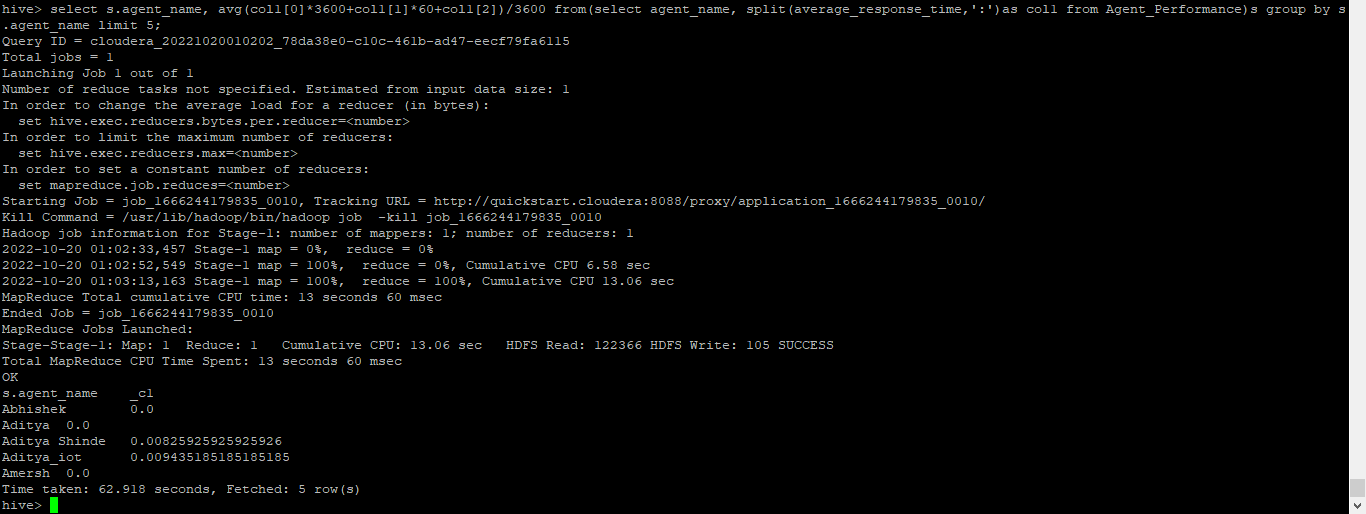
**11. How many feedback agents have received more than 4.5 average**

hive> select agent\_name, avg(total\_feedback) as Avg\_feedback from Agent\_Performance group by agent\_name having Avg\_feedback > 4.5 limit 5;

****

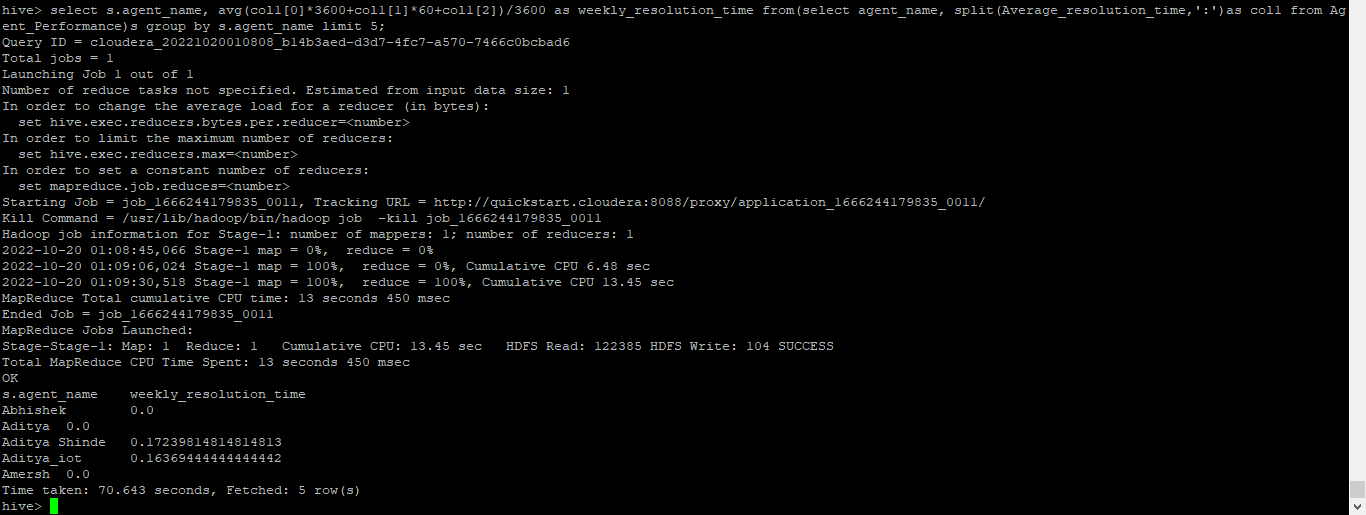
**12. average weekly response time for each agent**

hive> select s.agent\_name, avg(col1[0]\*3600+col1[1]\*60+col1[2])/3600 from(select agent\_name, split(average\_response\_time,':')as col1 from Agent\_Performance)s group by s.agent\_name limit 5;

****

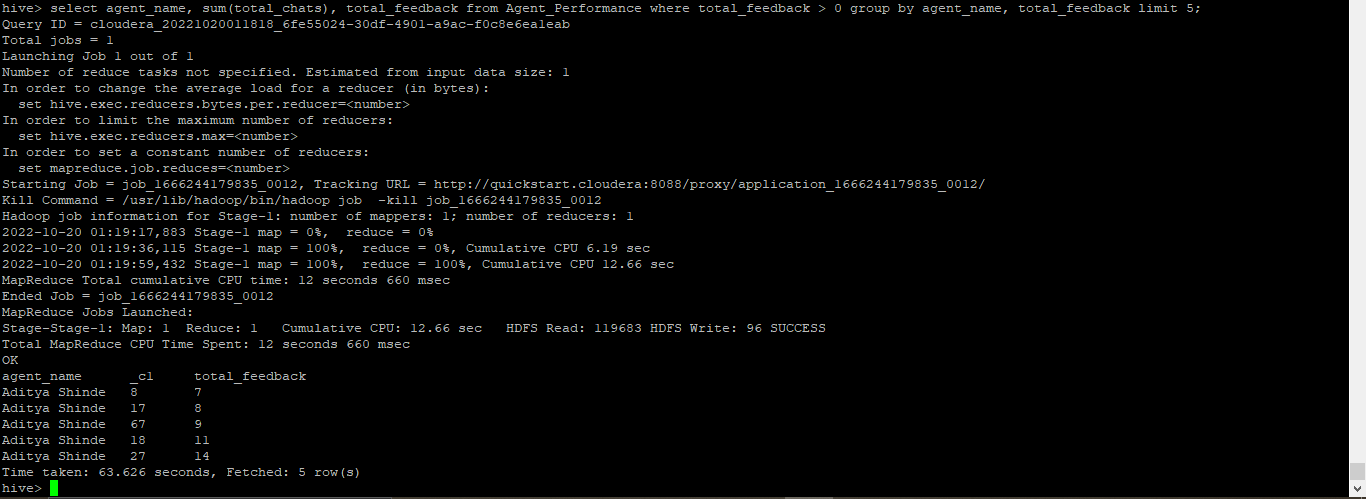
**13. average weekly resolution time for each agents**

hive> select s.agent\_name, avg(col1[0]\*3600+col1[1]\*60+col1[2])/3600 as weekly\_resolution\_time from(select agent\_name, split(Average\_resolution\_time,':')as col1 from Agent\_Performance)s group by s.agent\_name limit 5;

****

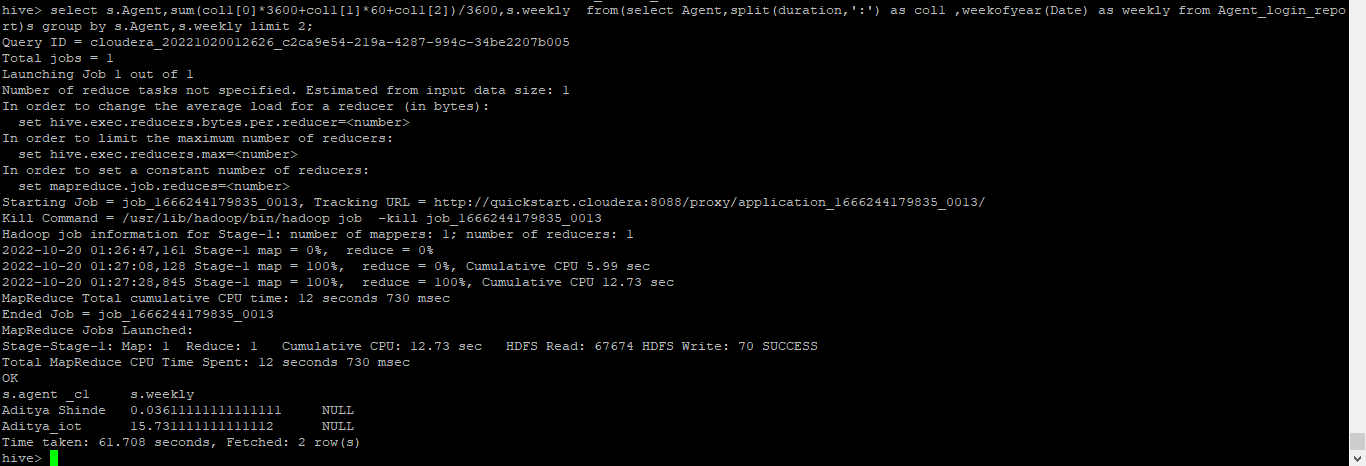
**14. Find the number of chat on which they have received a feedback**

hive> select agent\_name, sum(total\_chats), total\_feedback from Agent\_Performance where total\_feedback > 0 group by agent\_name, total\_feedback limit 5;

****

**15. Total contribution hour for each and every agents weekly basis**

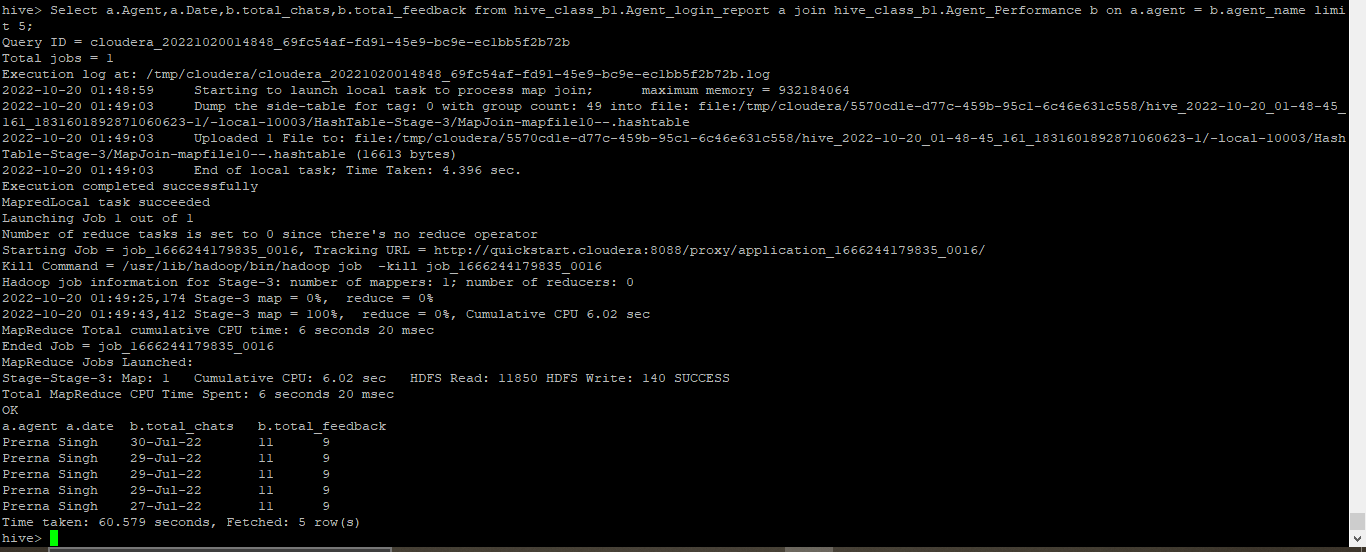
hive> select s.Agent,sum(col1[0]\*3600+col1[1]\*60+col1[2])/3600,s.weekly from(select Agent,split(duration,':') as col1 ,weekofyear(Date) as weekly from Agent\_login\_report)s group by s.Agent,s.weekly limit 2;

****

**16. Perform inner join, left join and right join based on the agent column and after joining the table export that data into your local system.**

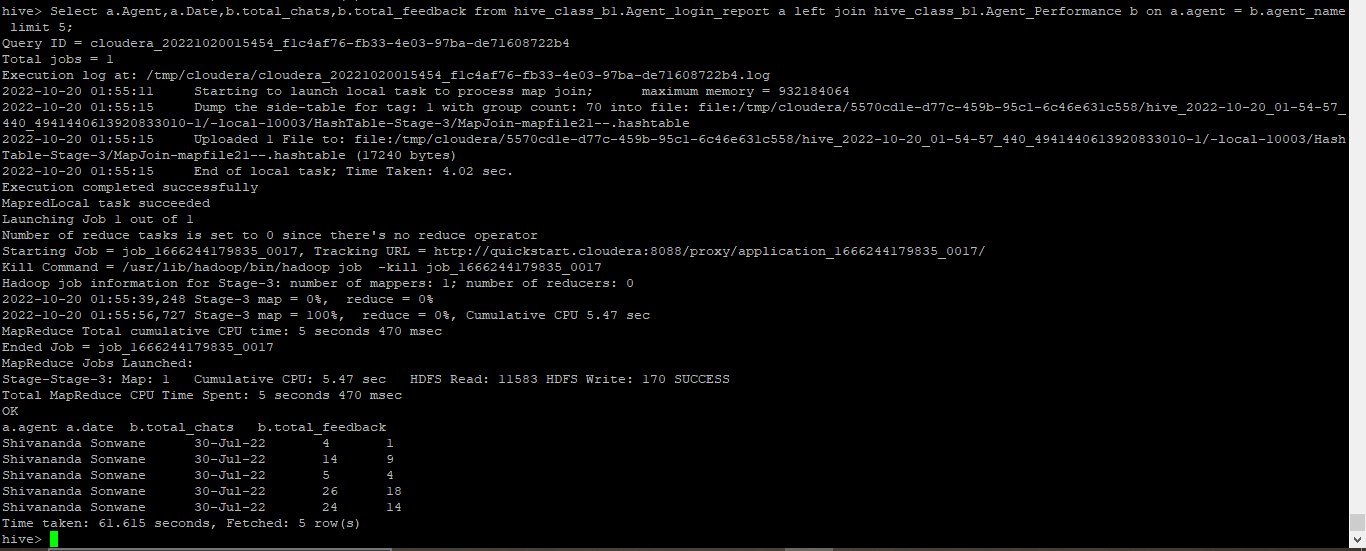
**INNER JOIN**

hive> Select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5;

****

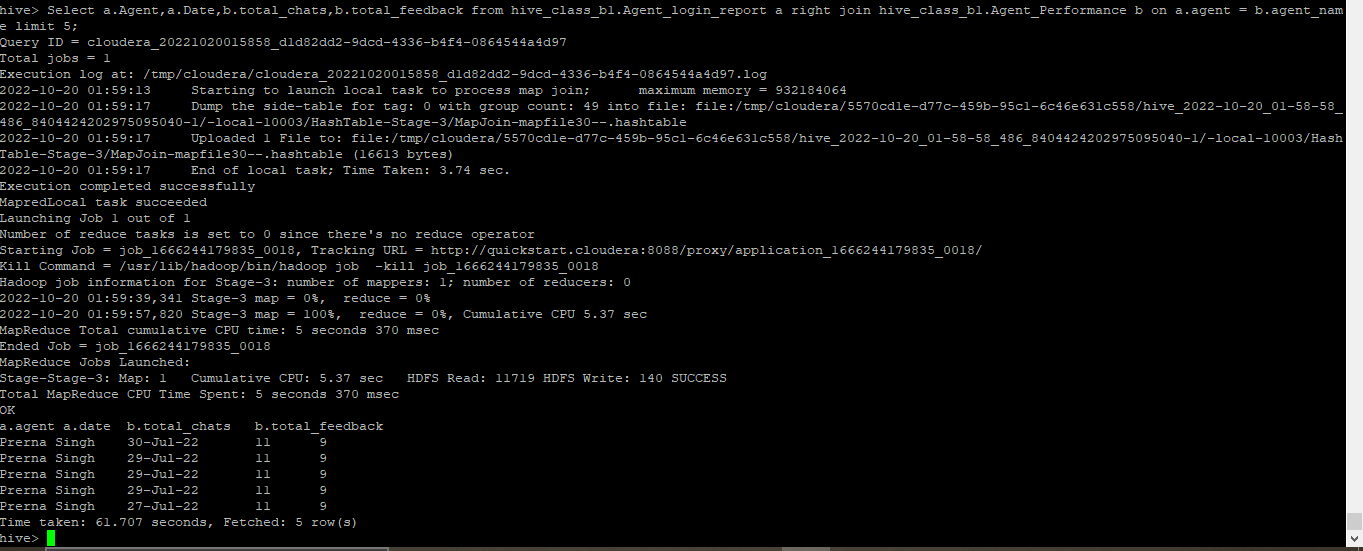
**LEFT JOIN**

hive> Select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a left join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5;

****

**RIGHT JOIN**

hive> Select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a right join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5;

****

**Saving the output into csv file by using following command in cli**

[cloudera@quickstart ~]$ hive -e 'select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5' /home/cloudera/hive\_mini\_project/inner\_join.csv;

[cloudera@quickstart ~]$ hive -e 'select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a left join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5' /home/cloudera/hive\_mini\_project/left\_join.csv;

[cloudera@quickstart ~]$ hive -e 'select a.Agent,a.Date,b.total\_chats,b.total\_feedback from hive\_class\_b1.Agent\_login\_report a right join hive\_class\_b1.Agent\_Performance b on a.agent = b.agent\_name limit 5' /home/cloudera/hive\_mini\_project/right\_join.csv;

**17. Perform partitioning on top of the agent column and then on top of that perform bucketing for**

**each partitioning.**

hive> create table Alr\_part\_buck

> (sl\_no int,Date string,login\_time string,logout\_time string,duration string)

> partitioned by (Agent string)

> clustered by (Date) sorted by (Date) into 4 buckets

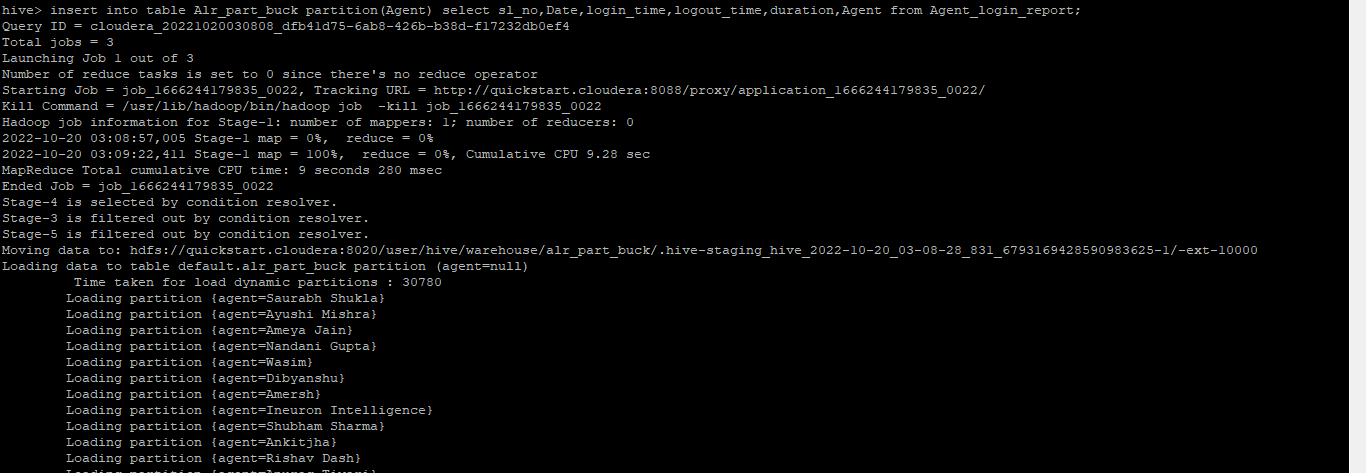
> row format delimited

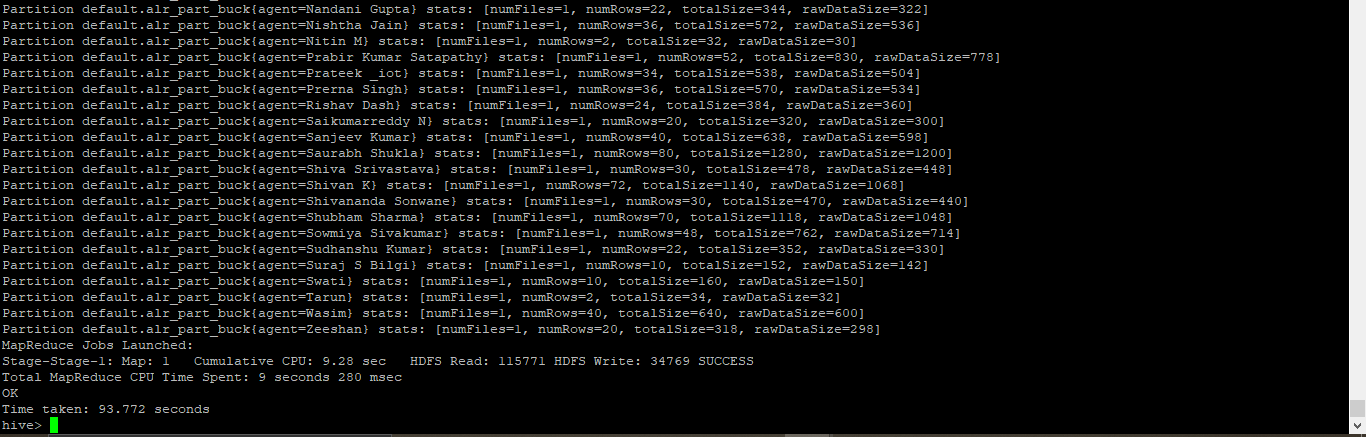
> fields terminated by ',';

hive> set hive.exec.dynamic.partition=true;

hive> set hive.exec.dynamic.partition.mode=nonstrict;

hive> insert into table Alr\_part\_buck partition(Agent) select sl\_no,Date,login\_time,logout\_time,duration,Agent from Agent\_login\_report;





hive> Create table AP\_partition\_Bucket

> (

> sl\_no int,date string,total\_chats int,average\_response\_time string,average\_resolution\_time string,average\_rating float,total\_feedback int

> )

> partitioned by (agent\_name string)

> CLUSTERED BY (Date) sorted by (Date) INTO 8 BUCKETS

> ROW FORMAT DELIMITED

> FIELDS TERMINATED BY ',';

hive> insert into table AP\_partition\_Bucket partition(agent\_name) select sl\_no,date,total\_chats, average\_response\_time, average\_resolution\_time,average\_rating,total\_feedback,agent\_name from Agent\_Performance;



