This is a real time dataset of the ineuron technical consultant team. You have to perform hive analysis on this given dataset.

Download Dataset 1 - https://drive.google.com/file/d/1WrG-9qv6atP-W3P\_-gYln1hHyFKRKMHP/view

Download Dataset 2 - https://drive.google.com/file/d/1-JIPCZ34dyN6k9CqJa-Y8yxIGq6vTVXU/view

Note: both files are csv files.

1. Create a schema based on the given dataset

**# creating table**

**hive> Create table AgentLogingReport**

**(**

**SLNo int,**

**Agent string,**

**Reportdate date,**

**LoginTime Timestamp,**

**LogoutTime Timestamp,**

**Duration Timestamp**

**)**

**row format delimited**

**fields terminated by ','**

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

**Create table AgentPerformance**

**(**

**SLNo int,**

**Responsedate date,**

**AgentName string,**

**TotalChats int,**

**AverageResponseTime Timestamp,**

**AverageResolutionTime Timestamp,**

**AverageRating float,**

**TotalFeedback int**

**)**

**row format delimited**

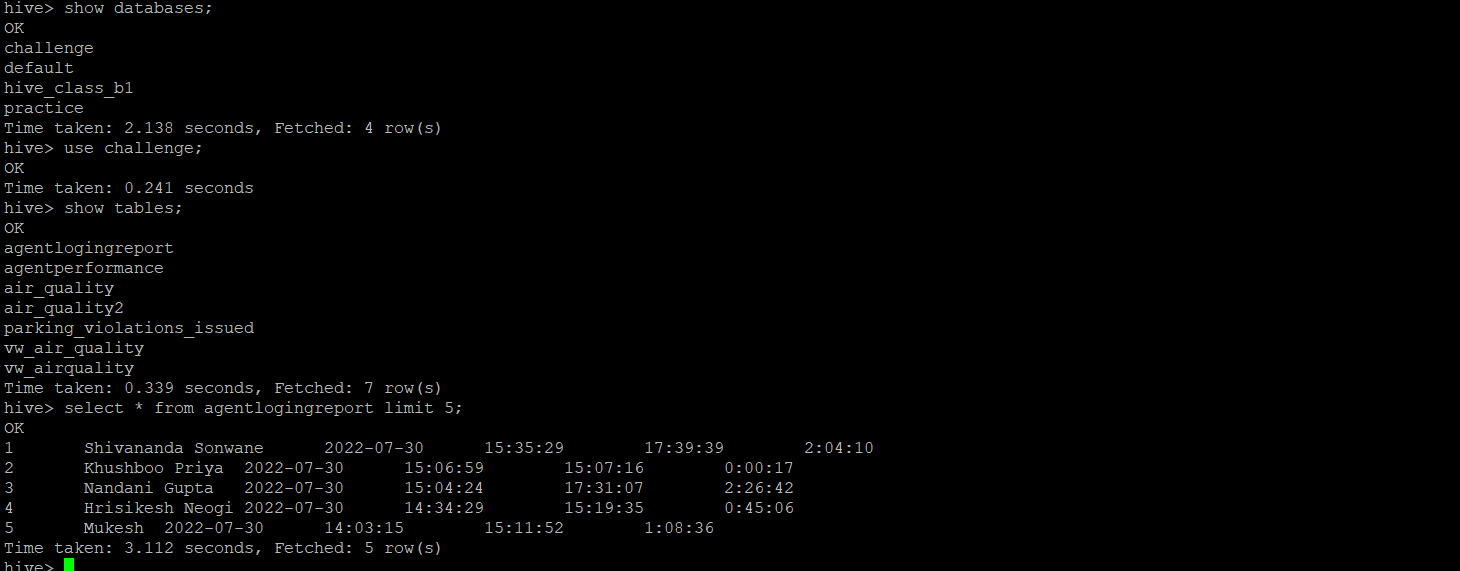
**fields terminated by ','**

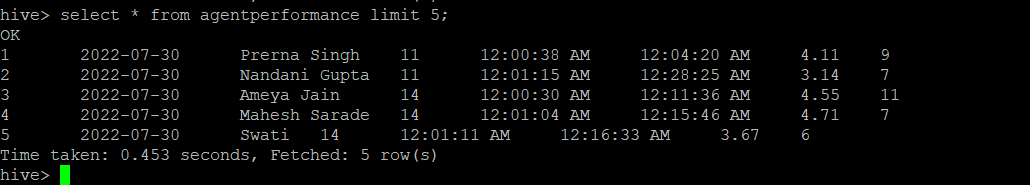
**tblproperties ("skip.header.line.count" = "1");**

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

**load data local inpath 'file:///config/workspace/AgentLogingReport.csv' into table AgentLogingReport;**

load data local inpath 'file:///config/workspace/AgentPerformance.csv' into table AgentPerformance;

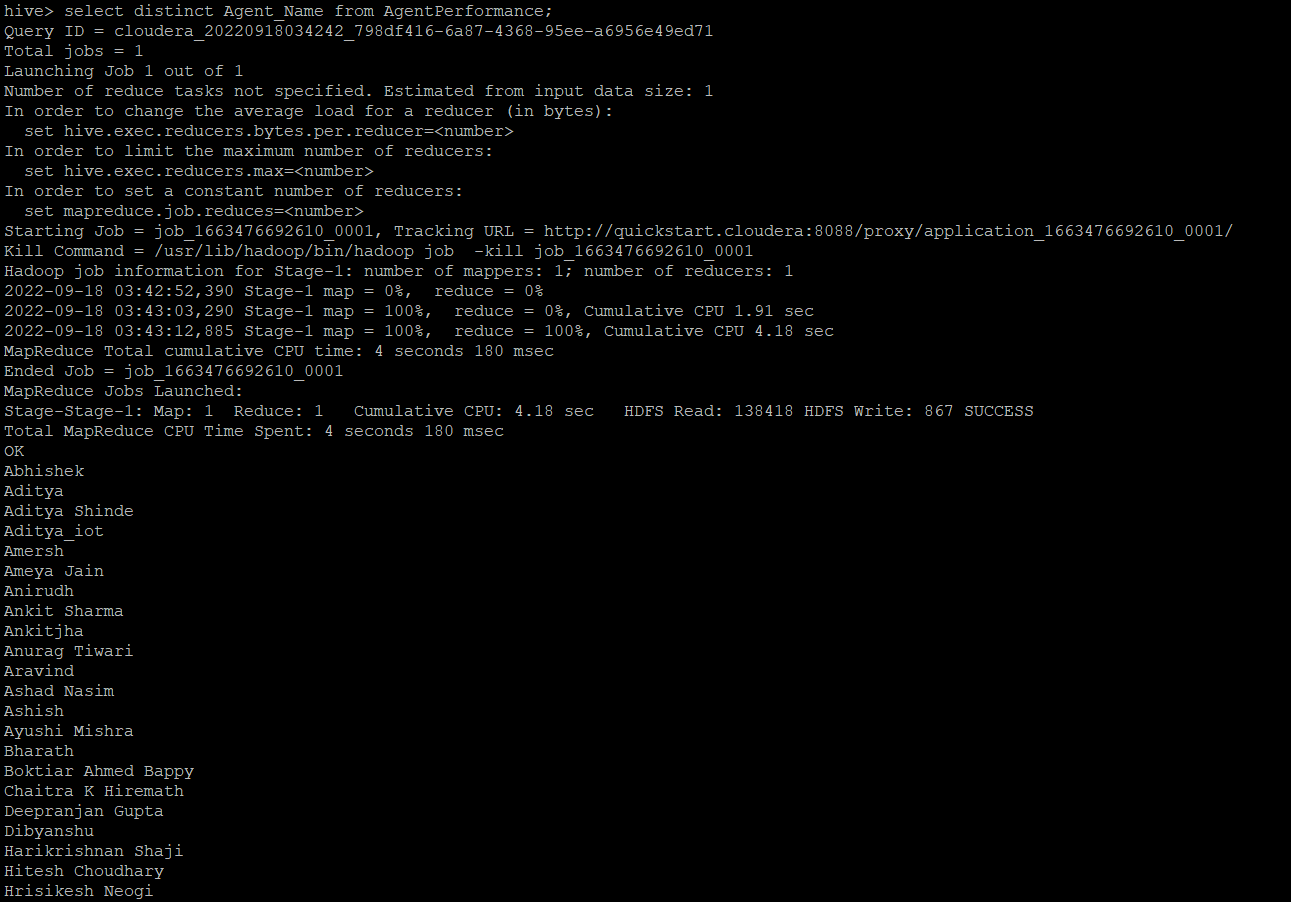


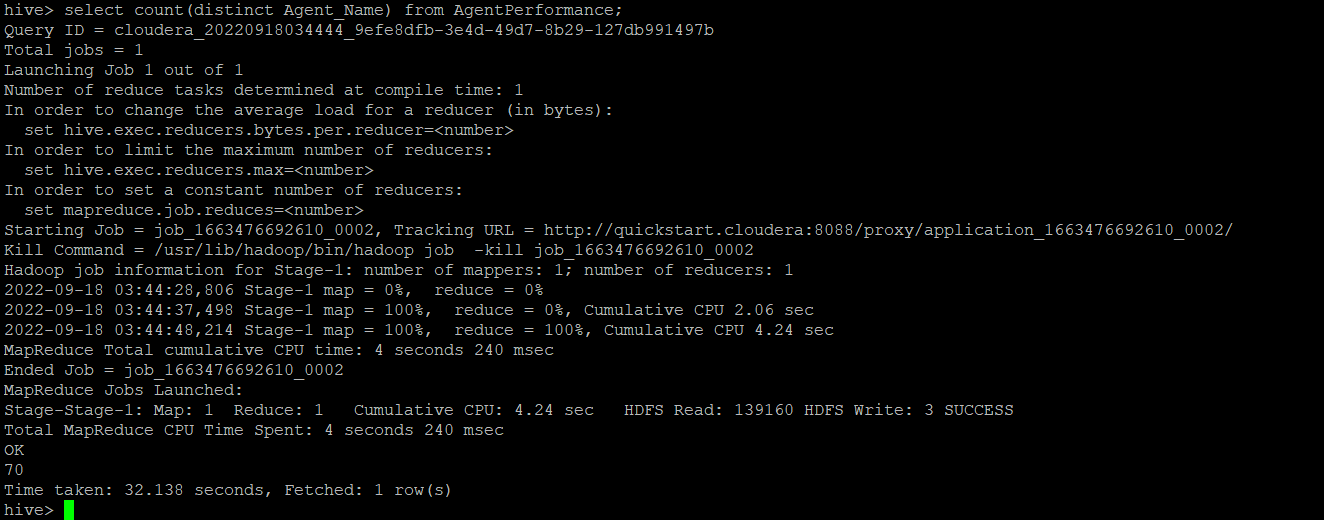


3. List of all agents' names.

**Hive> select distinct AgentName from AgentPerformance;**

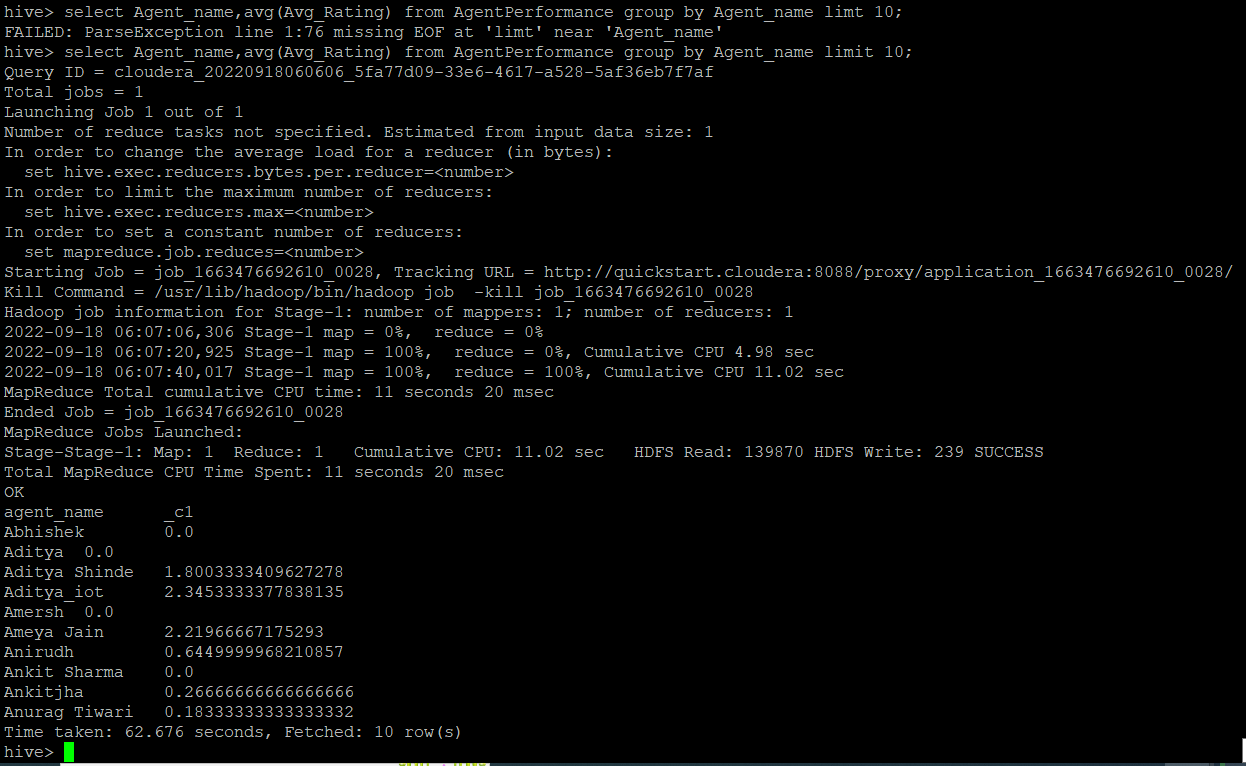
**Hive> select count(distinct AgentName) from AgentPerformance;**





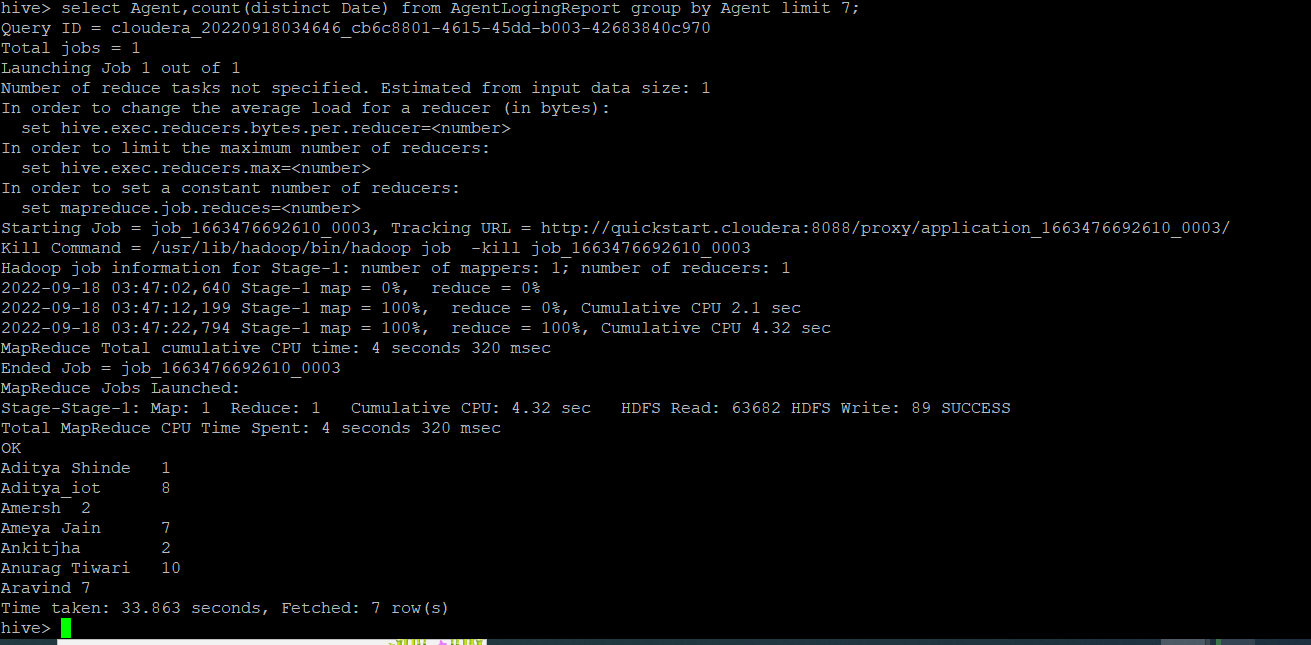
4. Find out agent average rating.

**Hive>select AgentName,avg(AverageRating) from AgentPerformance group by AgentName;**



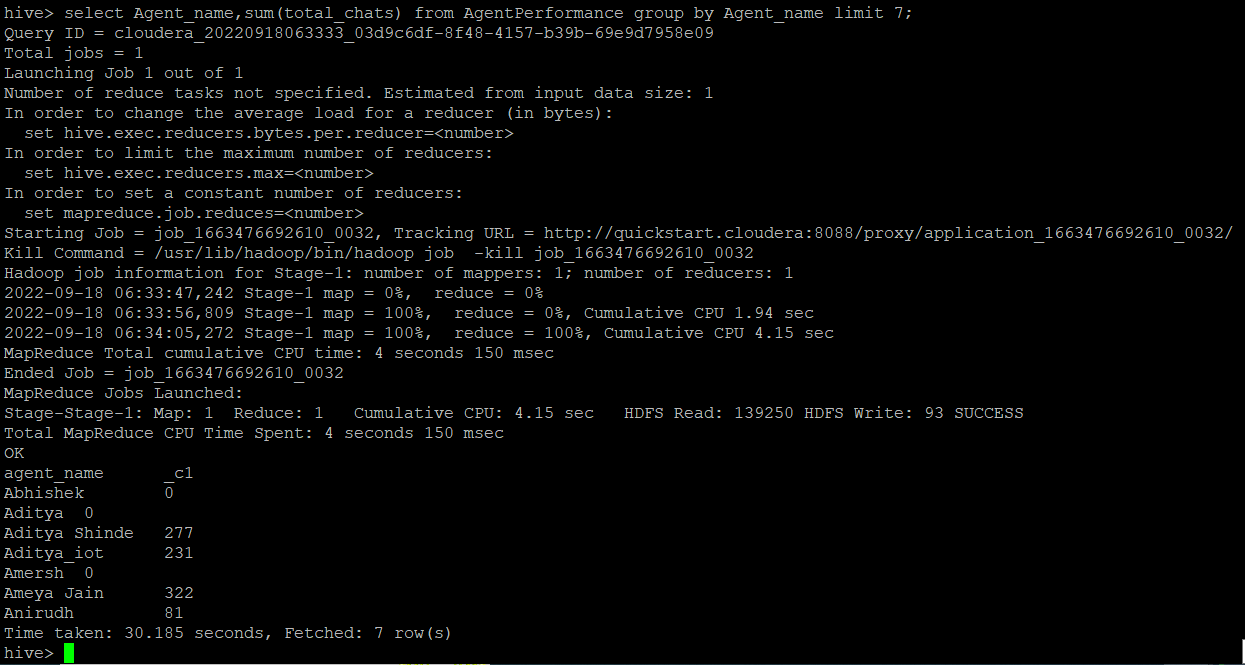
5. Total working days for each agents

**Hive> select Agent,count(distinct Reportdate) from AgentLogingReport group by Agent;**



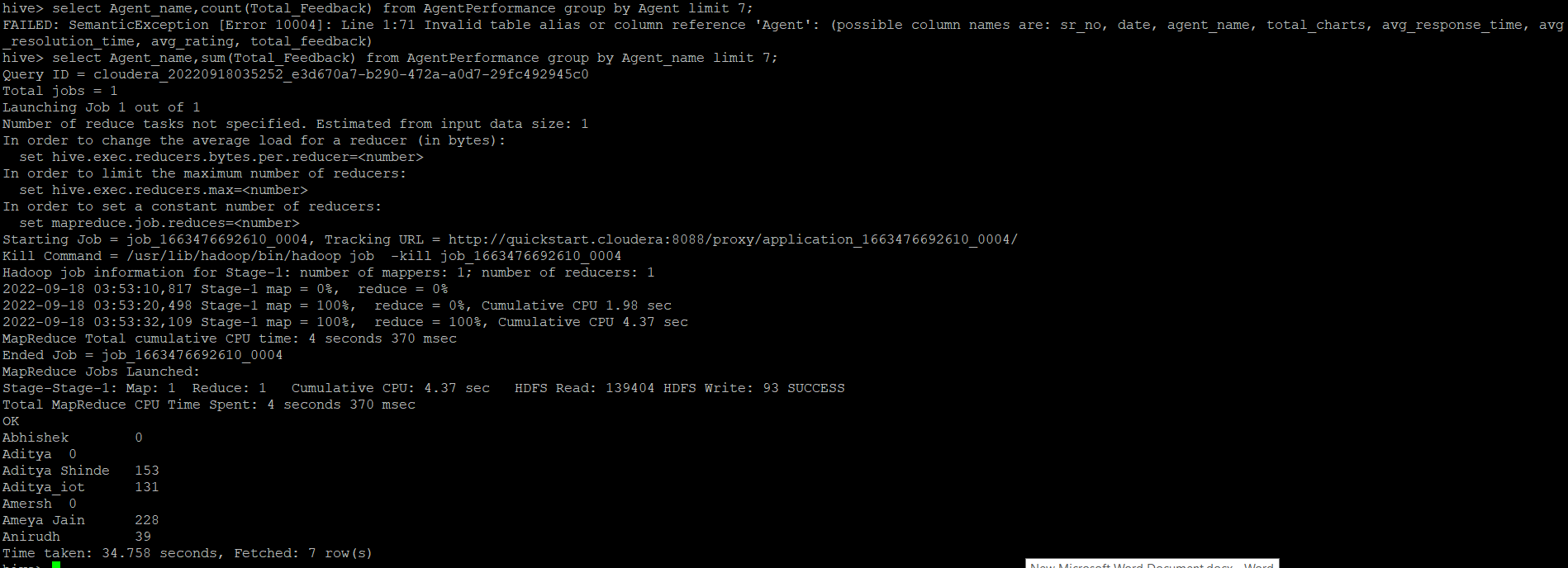
6. Total query that each agent have taken

**Hive>select AgentName,sum(TotalChats) from AgentPerformance group by AgentName;**



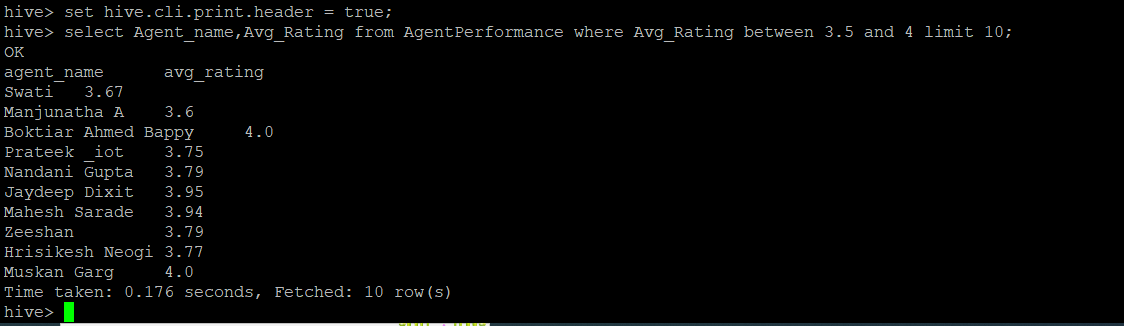
7. Total Feedback that each agent have received

**Hive> select AgenName,sum(TotalFeedback) from AgentPerformance group by AgentName;**



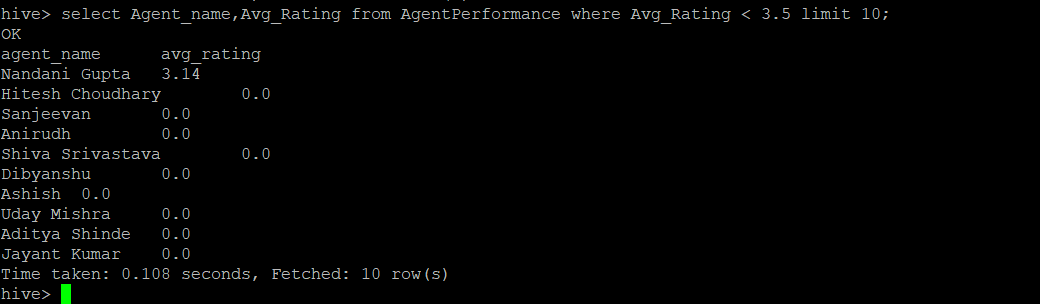
8. Agent name who have average rating between 3.5 to 4

**Hive>select AgentName,AverageRating from AgentPerformance where AverageRating between 3.5 and 4;**



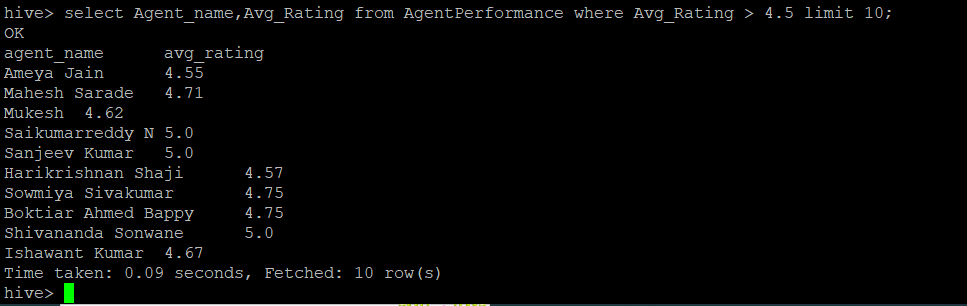
9. Agent name who have rating less than 3.5

**Hive>select AgentName,AverageRating from AgentPerformance where AverageRating< 3.5;**



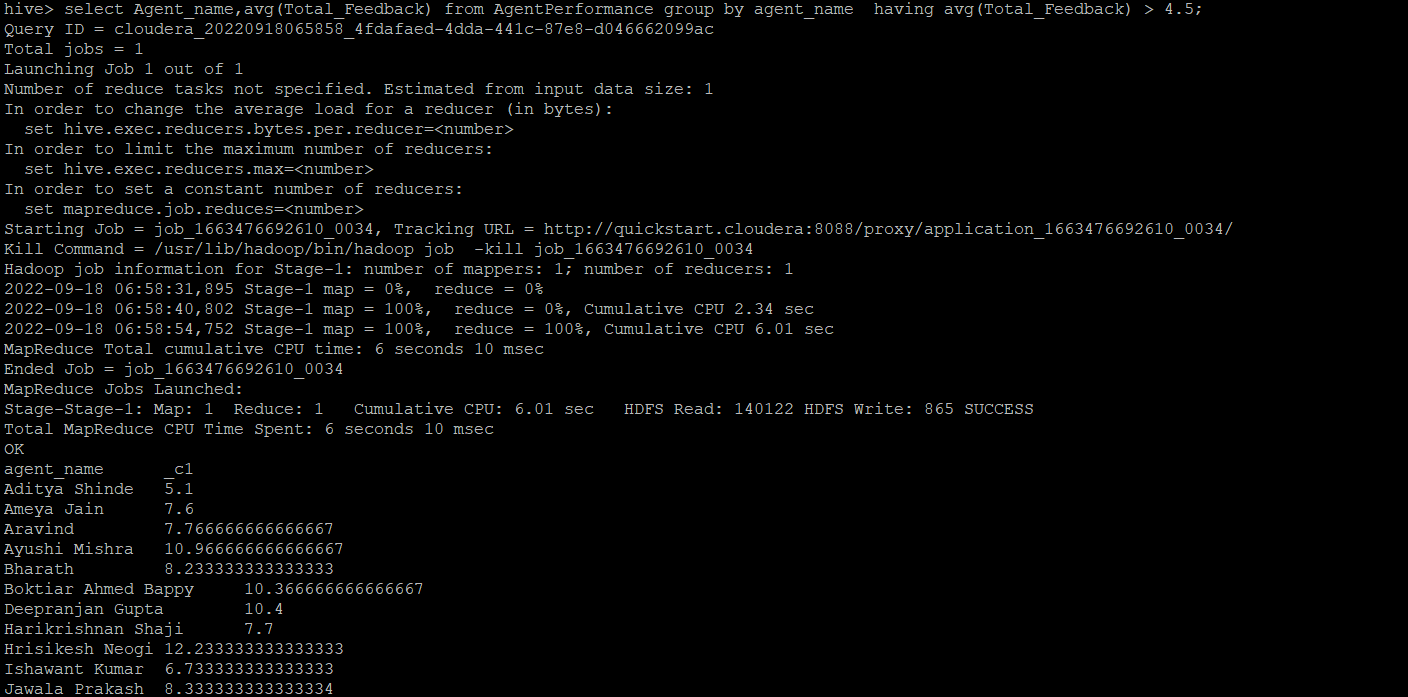
10. Agent name who have rating more than 4.5

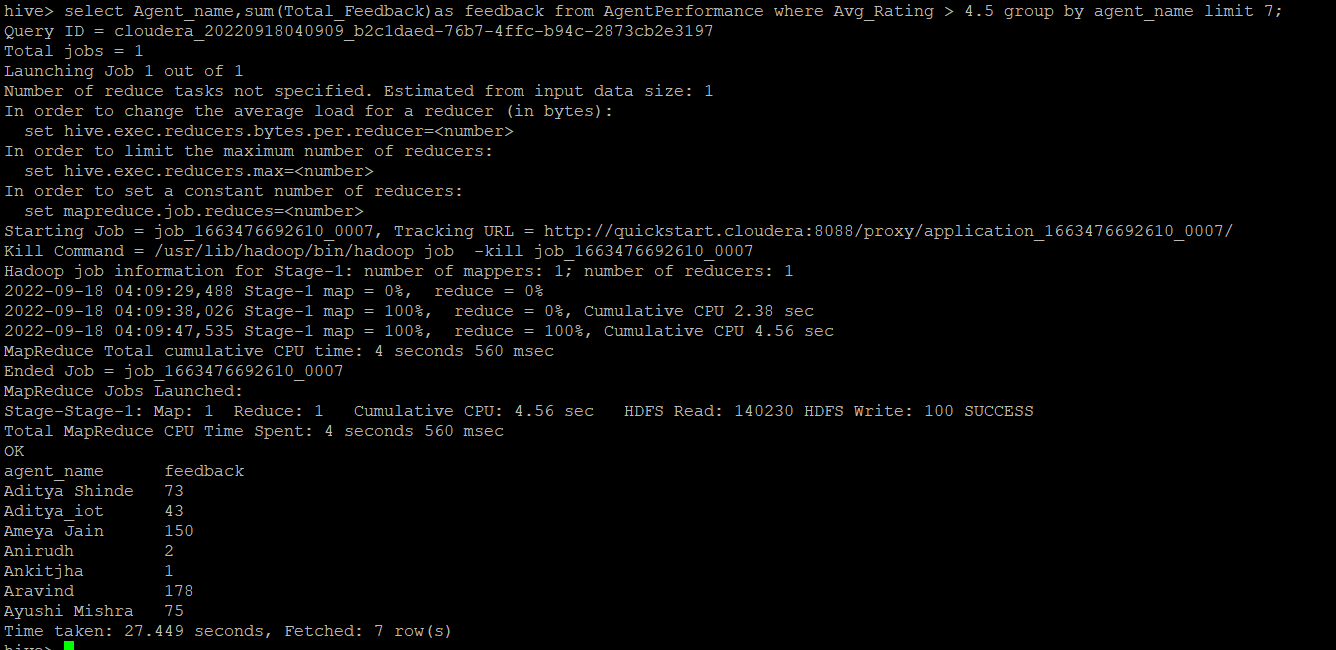
**Hive>select AgentName,AverageRating from AgentPerformance where AverageRating> 4.5;**



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

**Hive>select AgentName,avg(TotalFeedback) from AgentPerformance having avg(TotalFeedback) > 4.5;**

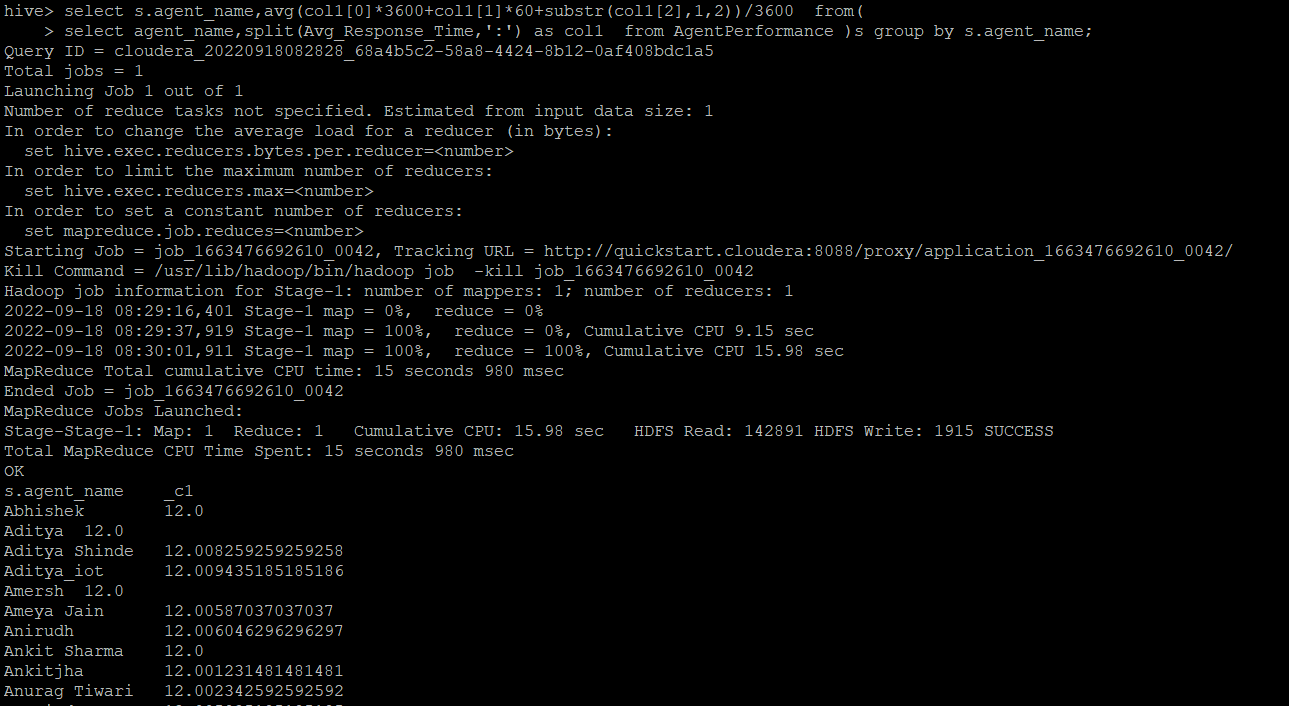




12. average weekly response time for each agent.

**Hive>select s.AgentName,avg(col1[0]\*3600+col1[1]\*60+substr(col1[2],1,2))/3600 from(**

**select AgentName,split(AverageResponseTime,':') as col1 from AgentPerformance )s group by s.AgentName;**

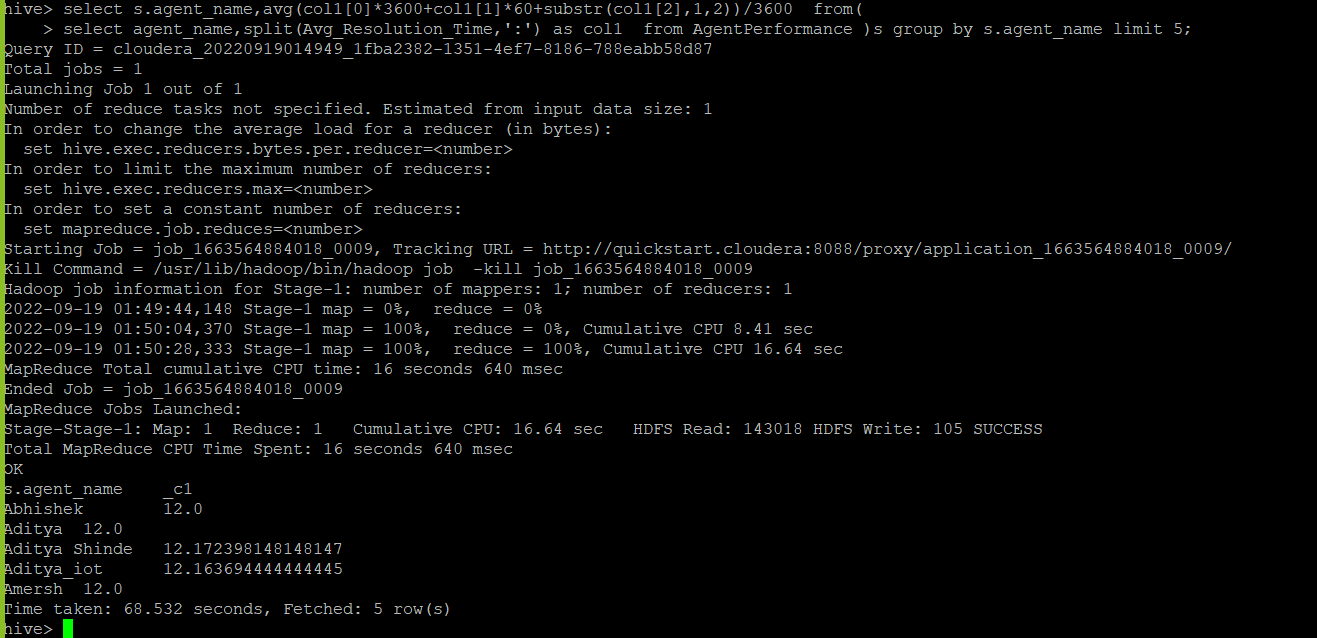


**Hive> select AgentName,avg(AverageResponseTime)as AverageResponseTime,weekofyear(Responsedate) as weekly from AgentPerformance group by AgentName,weekofyear(Responsedate);**

13. average weekly resolution time for each agents

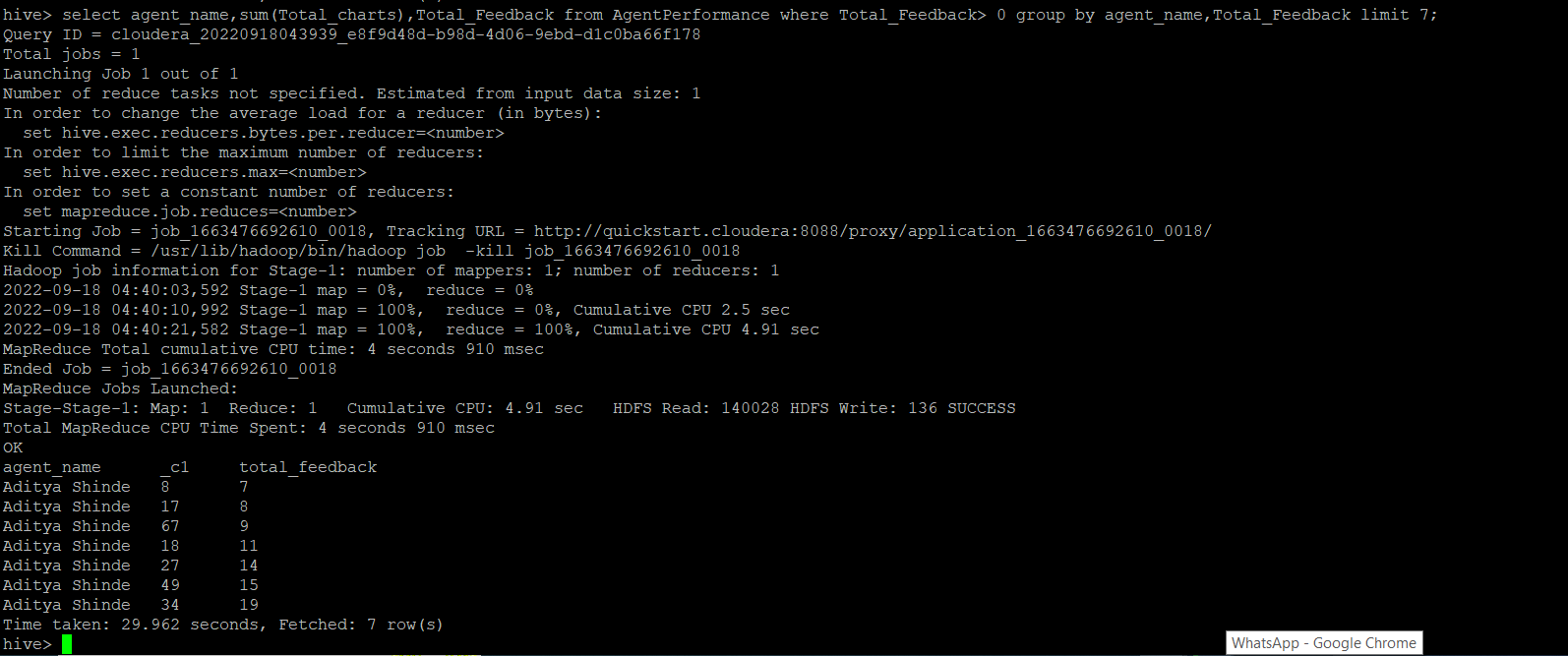
**Hive>select s.AgentName,avg(col1[0]\*3600+col1[1]\*60+substr(col1[2],1,2))/3600 from(**

**select AgentName,split(AverageResolutionTime,':') as col1 from AgentPerformance )s group by s.AgentName;**



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

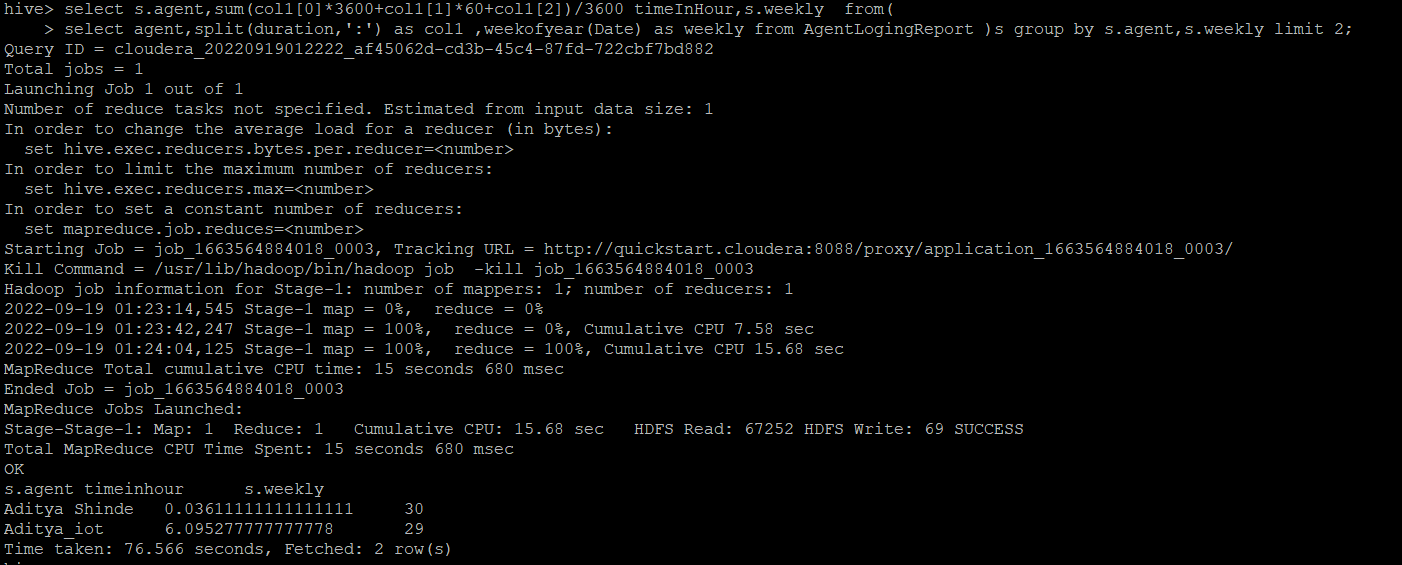
**Hive> select AgentName,sum(TotalChats),TotalFeedback from AgentPerformance where TotalFeedback> 0 group by AgentName,TotalFeedback;**



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 timeInHour,s.weekly from(**

**select Agent,split(duration,':') as col1 ,weekofyear(Reportdate) as weekly from AgentLogingReport )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 -e 'select a.agent,a.date,a.Duration,b.Total\_chats,b.Total\_Feedback from challenge.AgentLogingReport a join challenge.AgentPerformance b on a.agent = b.agent\_name' > file:///config/workspace/inner\_join.csv;**

**left join:**

**hive -e 'select a.agent,a.date,a.Duration,b.Total\_chats,b.Total\_Feedback from challenge.AgentLogingReport a left join challenge.AgentPerformance b on a.agent = b.agent\_name' > file:///config/workspace /left\_join.csv;**

**left join with performance improved due to /\*+ streamtable(a) \*/ hint:**

**hive -e 'select /\*+ streamtable(a) \*/a.agent,a.date,a.Duration,b.Total\_charts,b.Total\_Feedback from challenge.AgentLogingReport a left join challenge.AgentPerformance b on a.agent = b.agent\_name' > file:///config/workspace /left\_join.csv;**

**Right join:**

**hive -e 'select a.agent,a.date,a.Duration,b.Total\_charts,b.Total\_Feedback from challenge.AgentLogingReport a right join challenge.AgentPerformance b on a.agent = b.agent\_name' > file:///config/workspace /right\_join.csv;**

**Right join with performance improved due to /\*+ streamtable(a) \*/ hint:**

**hive -e 'select /\*+ streamtable(a) \*/a.agent,a.date,a.Duration,b.Total\_charts,b.Total\_Feedback from challenge.AgentLogingReport a right join challenge.AgentPerformance b on a.agent = b.agent\_name' > file:///config/workspace /left\_join.csv;**

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

**Create table AgentLogingReport\_partitioned**

**(**

**SLNo int,**

**Reportdate date,**

**LoginTime Timestamp,**

**LogoutTime Timestamp,**

**Duration Timestamp**

**)partitioned by (Agent string)**

**CLUSTERED BY (Reportdate) sorted by (Reportdate) 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 OVERWRITE TABLE AgentLogingReport\_partitioned PARTITION(Agent) SELECT SLNo,Reportdate,LoginTime,LogoutTime,Duration,Agent from AgentLogingReport;**

**Hive>Create table AgentPerformance\_partitioned**

**(**

**SLNo int,**

**Responsedate date,**

**TotalChats string,**

**AverageResponseTime Timestamp,**

**AverageResolutionTime Timestamp,**

**AverageRating float,**

**TotalFeedback int**

**)partitioned by (AgentName string)**

**CLUSTERED BY (Responsedate) sorted by (Responsedate) INTO 8 BUCKETS**

**ROW FORMAT DELIMITED**

**FIELDS TERMINATED BY ',';**

**hive> INSERT OVERWRITE TABLE AgentPerformance\_partitioned**

**PARTITION(AgentName) SELECT SLNo,Responsedate,AgentName,TotalChats,AverageResponseTime,AverageResolutionTime,AverageRating,TotalFeedback from AgentPerformance;**