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

create table agent\_loging\_report(

sl\_no int,

agent string,

date string,

login\_time string,

logout\_time string,

duration string)

row format delimited

fields terminated by ','

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

create table agent\_performance(

sl\_no int,

date string,

agent string,

total\_chats int,

average\_reponse\_time string,

average\_resolution\_time string,

average\_rating float,

total\_feedback int)

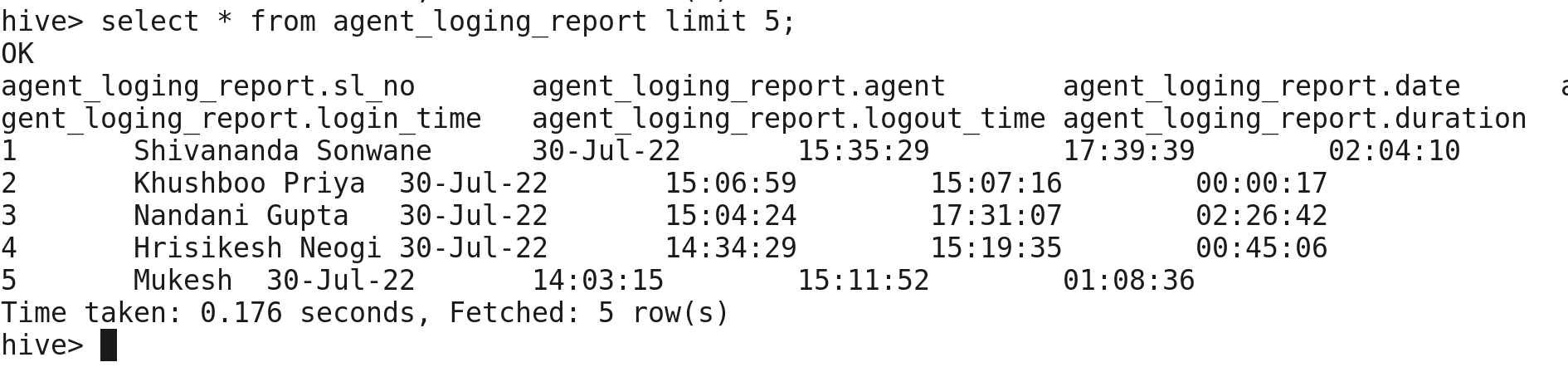
row format delimited

fields terminated by ','

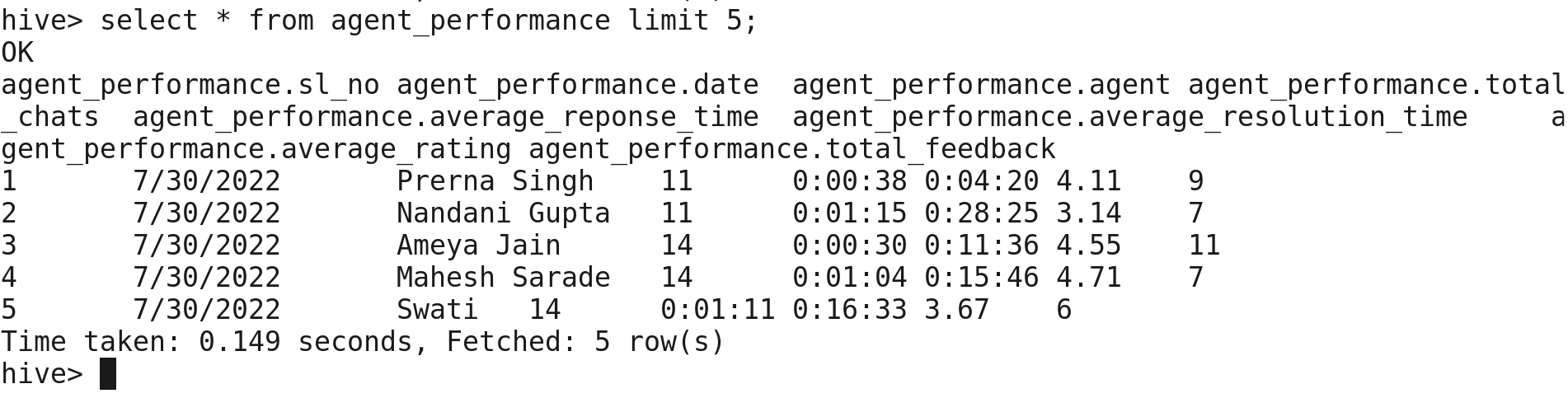
tblproperties("skip.header.line.count"="1");

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

load data local inpath 'file:///home/cloudera/data/AgentLogingReport.csv' into table agent\_loging\_report;

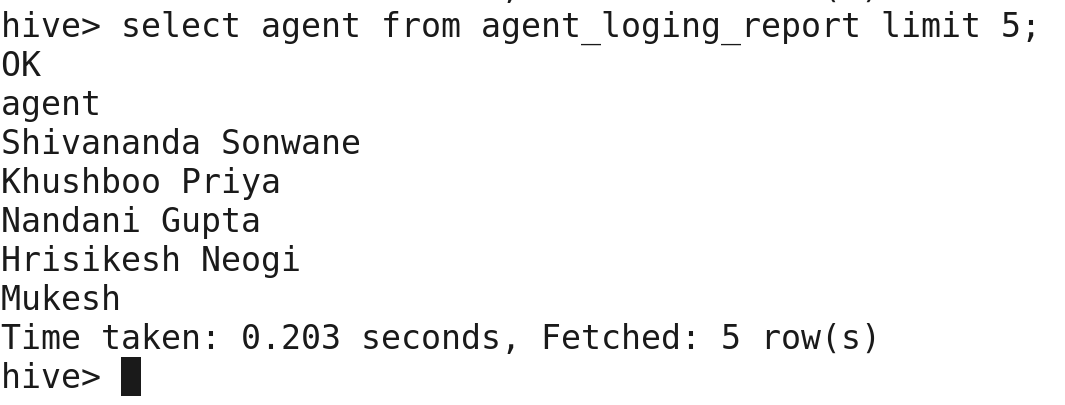


load data local inpath 'file:///home/cloudera/data/AgentPerformance.csv' into table agent\_performance;



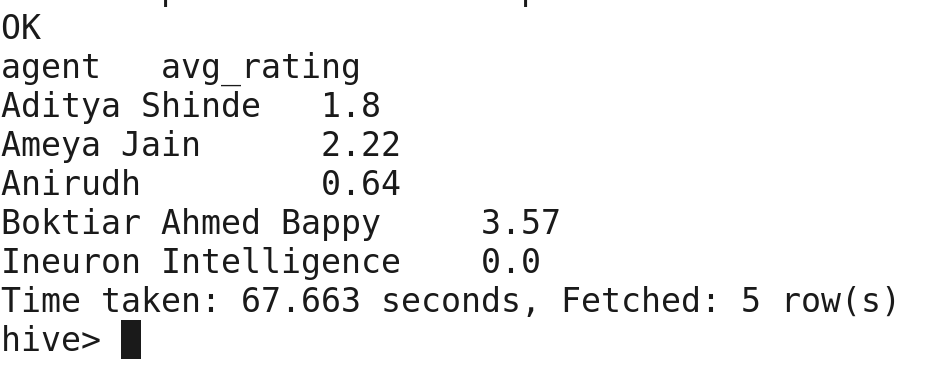
3. List of all agents’ names.

select agent from agent\_loging\_report limit 5;



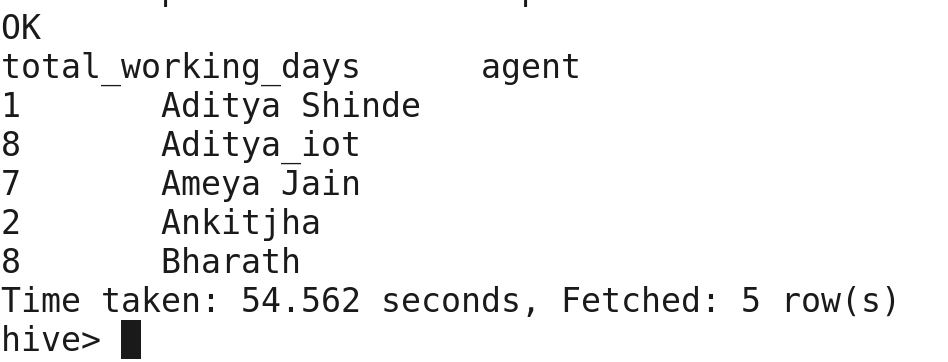
4. Find out agent average rating.

select agent,round(avg(average\_rating),2)avg\_rating from agent\_performance group by agent limit 5;



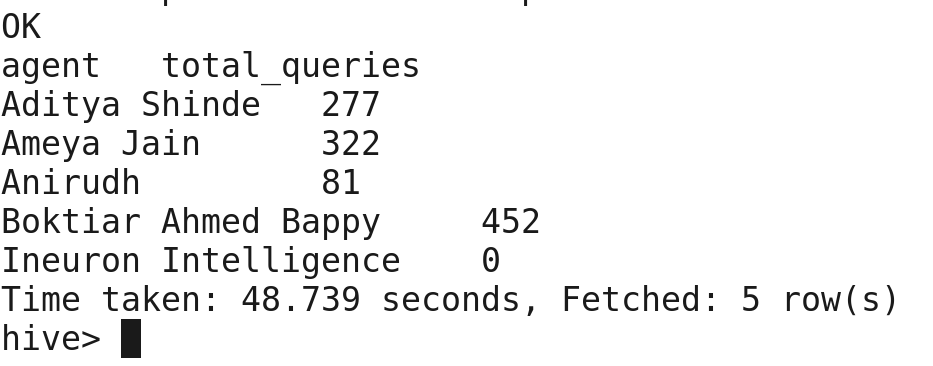
5. Total working days for each agent

select count (distinct date) as total\_working\_days, agent from agent\_loging\_report group by agent limit 5;



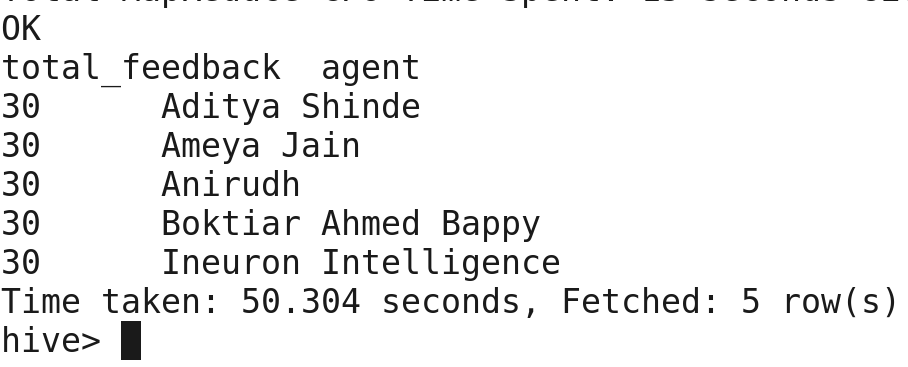
6. Total query that each agent has taken

select agent,sum(total\_chats) total\_queries from agent\_performance group by agent limit 5;



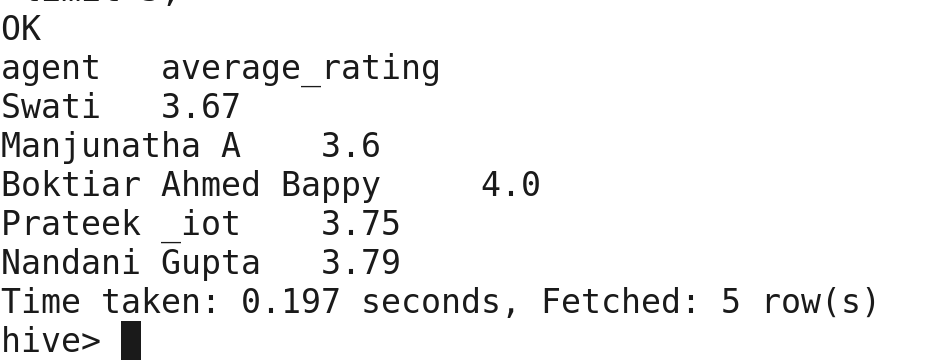
7. Total Feedback that each agent has received

select count(total\_feedback) as total\_feedback, agent from agent\_performance group by agent limit 5;



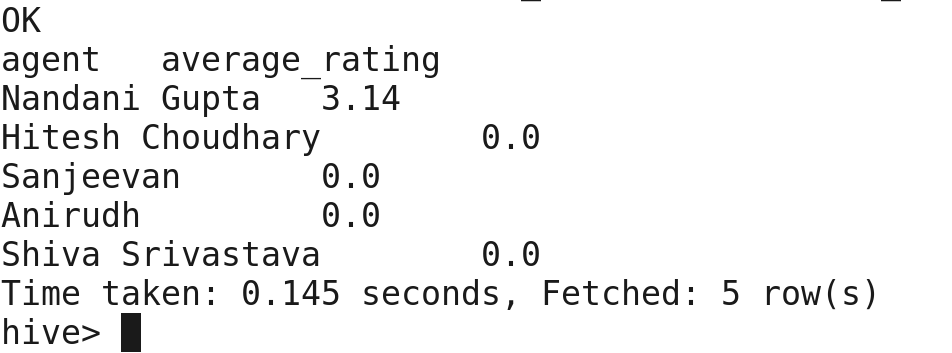
8. Agent name who have average rating between 3.5 to 4

select agent,average\_rating from agent\_performance where average\_rating between 3.5 and 4 limit 5;



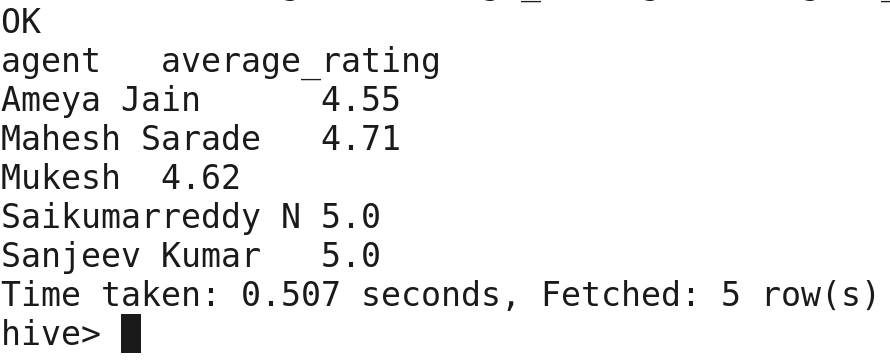
9. Agent name who have rating less than 3.5

select agent,average\_rating from agent\_performance where average\_rating < 3.5 limit 5;



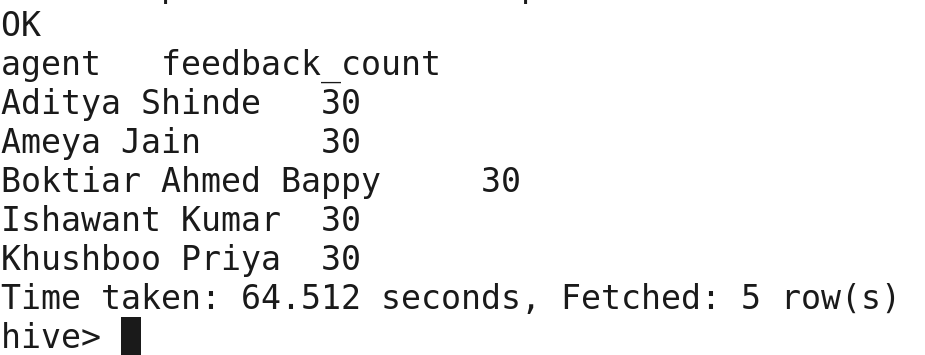
10. Agent name who have rating more than 4.5

select agent,average\_rating from agent\_performance where average\_rating > 4.5 limit 5;



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

select agent,feedback\_count from (select agent, avg(total\_feedback)avg\_feedback, count(total\_feedback)feedback\_count from agent\_performance group by agent)a where avg\_feedback > 4.5 limit 5;



12. average weekly response time for each agent

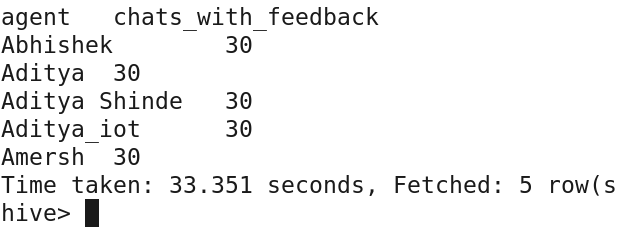
tried a lot but getting date formatting error.

13. average weekly resolution time for each agent

tried a lot but getting date formatting error.

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

Select agent,count(total\_feedback) as chats\_with\_feedback from agent\_performance group by agent limit 5;



15. Total contribution hour for each and every agent’s weekly basis

tried a lot but getting date formatting error.

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.

set hive.enforce.sortmergebucketmapjoin = false;

set hive.auto.convert.sortmerge.join=true;

set hive.optimise.bucketmapjoin=true;

set hive.optimise.bucketmapjoin.sortedmerge=true;

inner join -

hive -e 'select \* from agent\_loging\_report\_part\_buck l inner join agent\_performance\_part\_buck p on l.agent=p.agent limit 5' > file:///home/cloudera/data/innerjoin.csv

left join -

hive -e 'select \* from agent\_loging\_report\_part\_buck l left join agent\_performance\_part\_buck p on l.agent=p.agent limit 5' > <file:///home/cloudera/data/leftjoin.csv>

right join -

hive -e 'select \* from agent\_loging\_report\_part\_buck l right join agent\_performance\_part\_buck p on l.agent=p.agent limit 5' > file:///home/cloudera/data/rightjoin.csv

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

create table agent\_loging\_report\_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 6 buckets

row format delimited

fields terminated by ',';

set hive.exec.dynamic.partition=true;

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

insert overwrite table agent\_loging\_report\_part\_buck partition (agent) select sl\_no, agent, date, login\_time, logout\_time,duration from agent\_loging\_report;

create table agent\_performance\_part\_buck(

sl\_no int,

date string,

total\_chats int,

average\_reponse\_time string,

average\_resolution\_time string,

average\_rating float,

total\_feedback int)

partitioned by (agent string)

clustered by (date)

sorted by (date)

into 3 buckets

row format delimited

fields terminated by ',';

set hive.exec.dynamic.partition=true;

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

insert overwrite table agent\_performance\_part\_buck partition (agent) select \* from agent\_performance;

inserted data

