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

A)

#Creating a database named Project

create database Project;

use Project;

#Creating a schema for Agent\_Loging\_Report dataset

create table Agent\_Loging\_Report

(

SL\_No int,

Agent\_Name string,

Date\_Date string,

Login\_Time string,

Logout\_Time string,

Duration string

)

row format delimited

fields terminated by ',';

#Creating a schema for Agent\_Performance dataset

create table Agent\_Performance

(

SL\_No int,

Date\_Date string,

Agent\_Name string,

Total\_Chats int,

Average\_Response\_Time string,

Average\_Resolution\_Time string,

Average\_Ratings 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.

A)

#loading data into Agent\_Loging\_Report table

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

#loading data into Agent\_Performance table

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

3)List of all agents' names.

A) select distinct agent\_name from agent\_loging\_report;

4)Find out agent average rating.

A) select agent\_name,avg(average\_ratings) from agent\_performance group by agent\_name;

5)Total working days for each agents.

A) hive> select Agent\_Name,count(distinct Date\_Date) from Agent\_Loging\_Report group by Agent\_Name;

6)Total query that each agent have taken.

A) hive> select Agent\_Name,sum(Total\_Chats) from Agent\_Performance group by Agent\_name;

7)Total Feedback that each agent have received.

A) select agent\_name,sum(total\_feedback) from agent\_performance group by agent\_name;

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

A) hive> select Agent\_Name,avg(Average\_Ratings) from Agent\_Performance group by Agent\_Name having avg(Average\_Ratings) between 3.5 and 4;

9)Agent name who have rating less than 3.5.

A) hive> select DISTINCT Agent\_Name,Average\_Ratings from Agent\_Performance where Average\_Ratings<3.5;

10)Agent name who have rating more than 4.5.

A) hive> select DISTINCT Agent\_Name,Average\_Ratings from Agent\_Performance where Average\_Ratings > 4.5;

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

A) hive> select agent\_name,avg(total\_feedback) as average from agent\_performance group by agent\_name having average>4.5;

12)average weekly response time for each agent.

A) hive> select agent\_name,year(date\_date) as years,weekofyear(date\_date) as weeks,avg(average\_response\_time) from agent\_performance group by agent\_name,year(date\_date),weekofyear(date\_date);

13)average weekly resolution time for each agents.

A) hive> select agent\_name,year(date\_date) as years,weekofyear(date\_date) as weeks,avg(average\_resolution\_time) from agent\_performance group by agent\_name,year(date\_date),weekofyear(date\_date);

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

A) hive> select sum(total\_chats)-(sum(total\_chats)-sum(total\_feedback)) as total from agent\_performance;

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

A)hive> SELECT agent\_name, YEAR(date\_date) as year, WEEKOFYEAR(date\_date) as week,abs(avg(HOUR(Duration) + MINUTE(Duration) / 60.0)) as total\_hours from agent\_loging\_report GROUP BY agent\_name, YEAR(date\_date), WEEKOFYEAR(date\_date);

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.

A) ----------------------------Inner join----------------------------

select AL.\*,AP.\* from agent\_loging\_report AL inner join agent\_performance AP on AL.agent\_name=AP.agent\_name;

----------------------------Left join------------------------------

select AL.\*,AP.\* from agent\_loging\_report AL left join agent\_performance AP on AL.agent\_name=AP.agent\_name;

----------------------------Right join------------------------------

select AL.\*,AP.\* from agent\_loging\_report AL right join agent\_performance AP on AL.agent\_name=AP.agent\_name;

----------------------------After joining the table export that data into your local system----------------------------

insert overwrite local directory '/one/agent.txt' row format delimited fields terminated by ',' stored as textfile select AL.\*,AP.\* from agent\_loging\_report AL inner join agent\_performance AP on AL.agent\_name=AP.agent\_name;

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

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

hive> create table part\_agent(

SL\_No int,

Date\_Date string,

Login\_Time string,

Logout\_Time string,

Duration string

)

partitioned by (agent\_name string);

hive> insert into part\_agent partition(agent\_name) select SL\_NO,Agent\_Name,Date\_Date,Login\_Time,Logout\_Time,Duration from agent\_loging\_report;