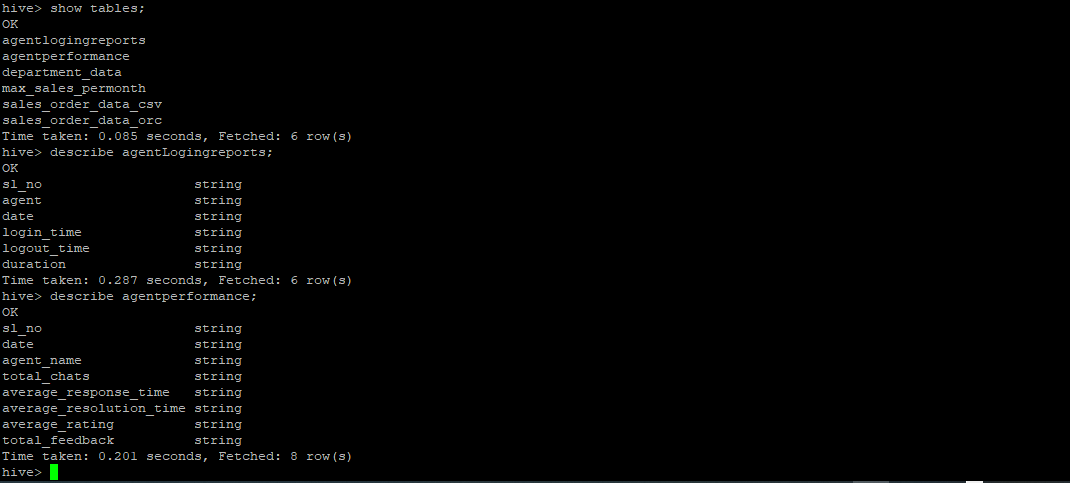
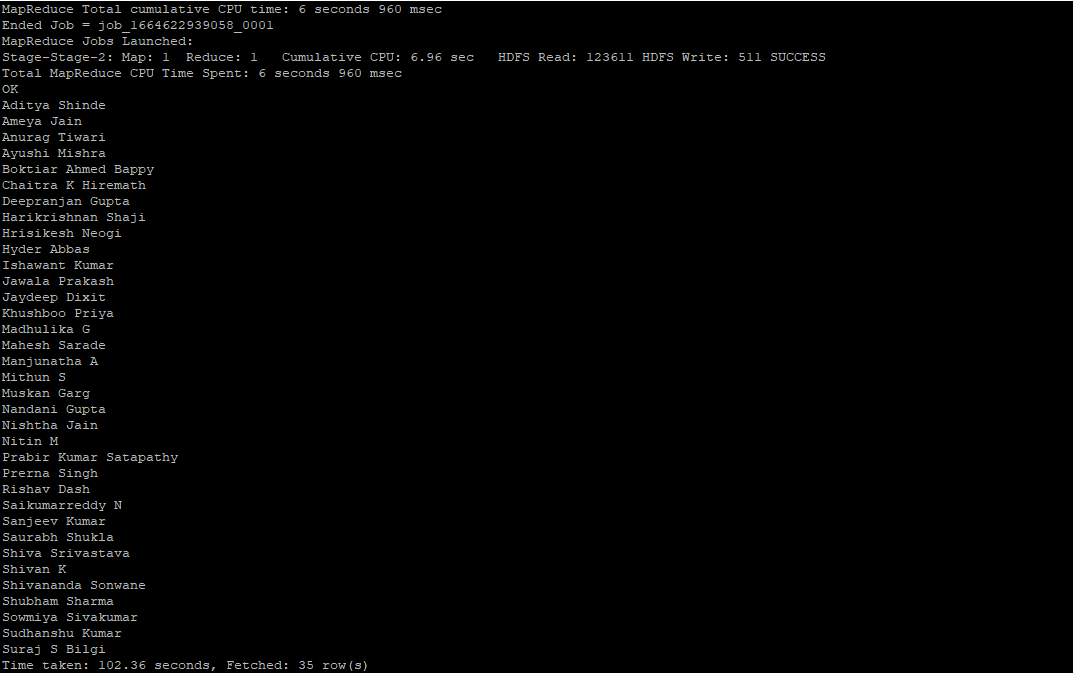
1. Create a schema based on the given dataset



1. Dump the data inside the hdfs in the given schema location.
2. List of all agents' names.

select distinct(Agent) from agentlogingreport inner join

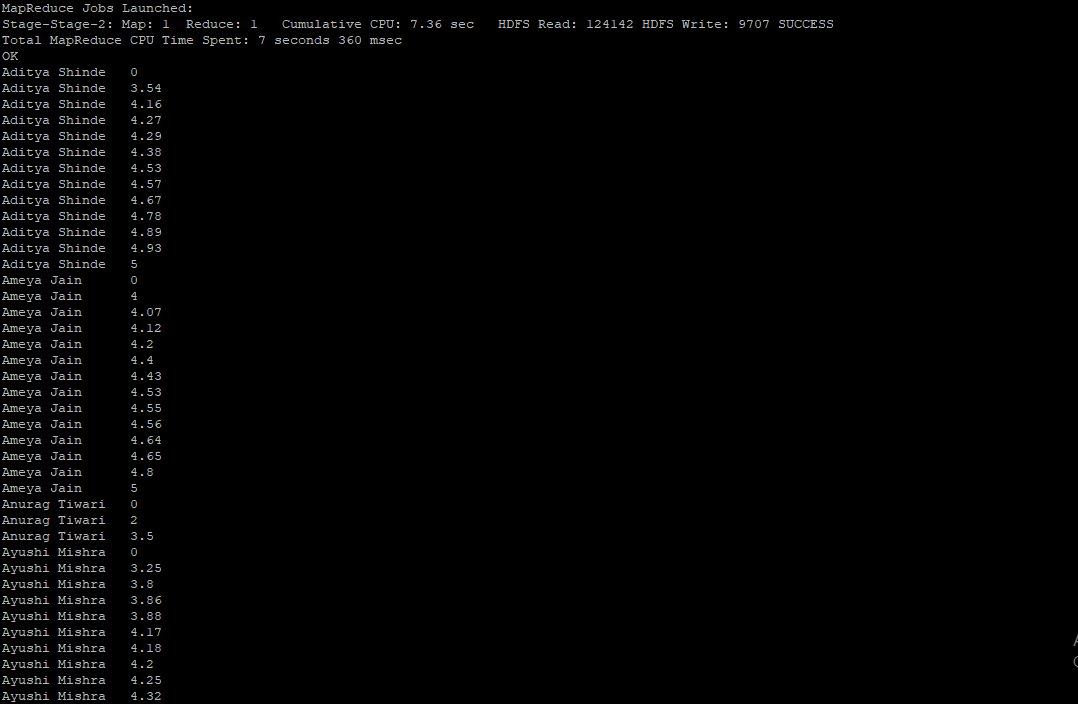
agentperformance on agentlogingreport.Agent = agentperformance.Agent\_Name;



1. Find out agent average rating.

Select distinct(Agent\_Name), Average\_Rating from agentperformance inner join

agentlogingreports on agentlogingreports.Agent = agentperformance.Agent\_Name



1. Total working days for each agents

Select Agent, count(Date) from agentlogingreports

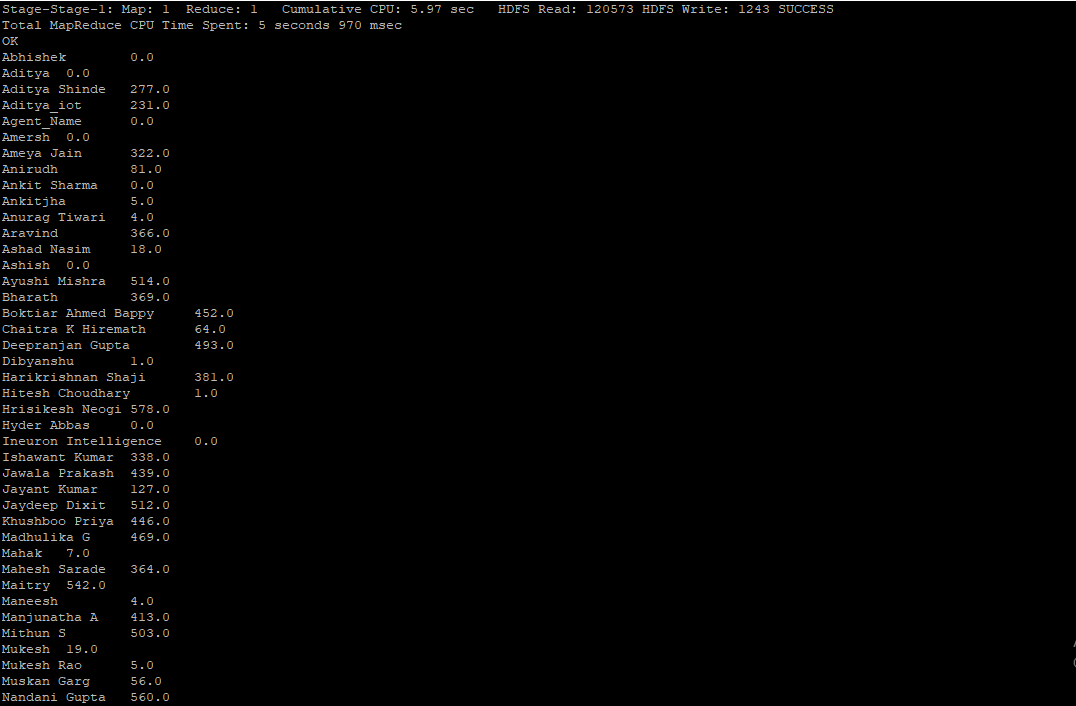
group by Agent;



1. Total query that each agent have taken

Select Agent\_Name , sum(Total\_Chats) from agentperformance

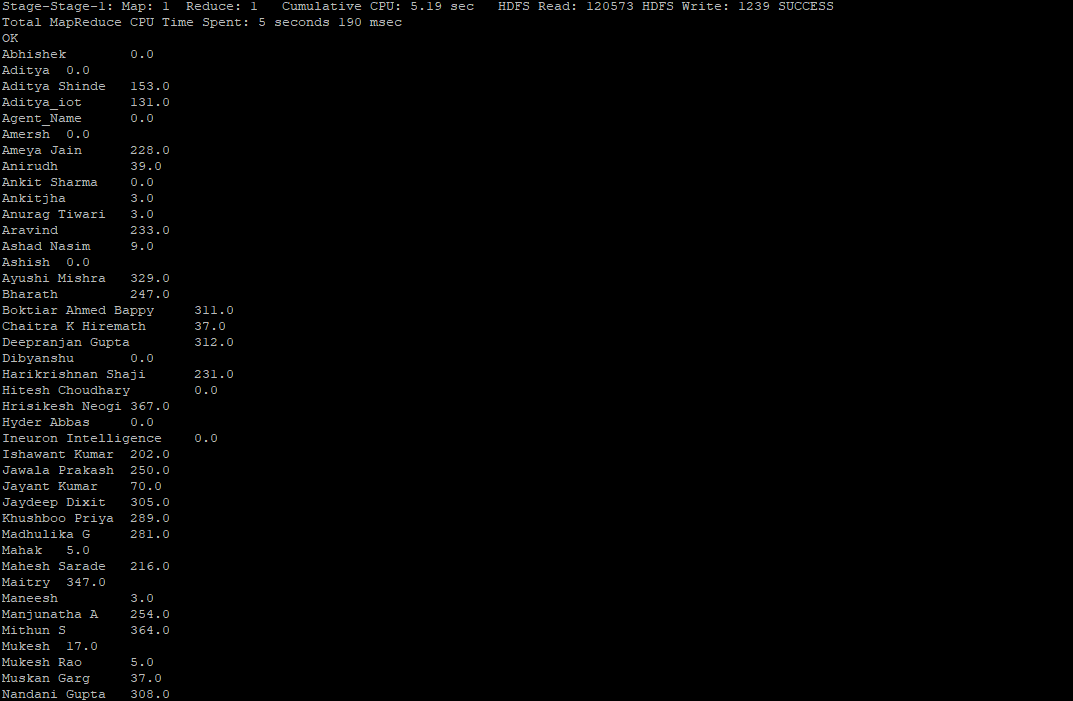
group by Agent\_Name;



1. Total Feedback that each agent have received

Select Agent\_Name , sum(Total\_Feedback) from agentperformance

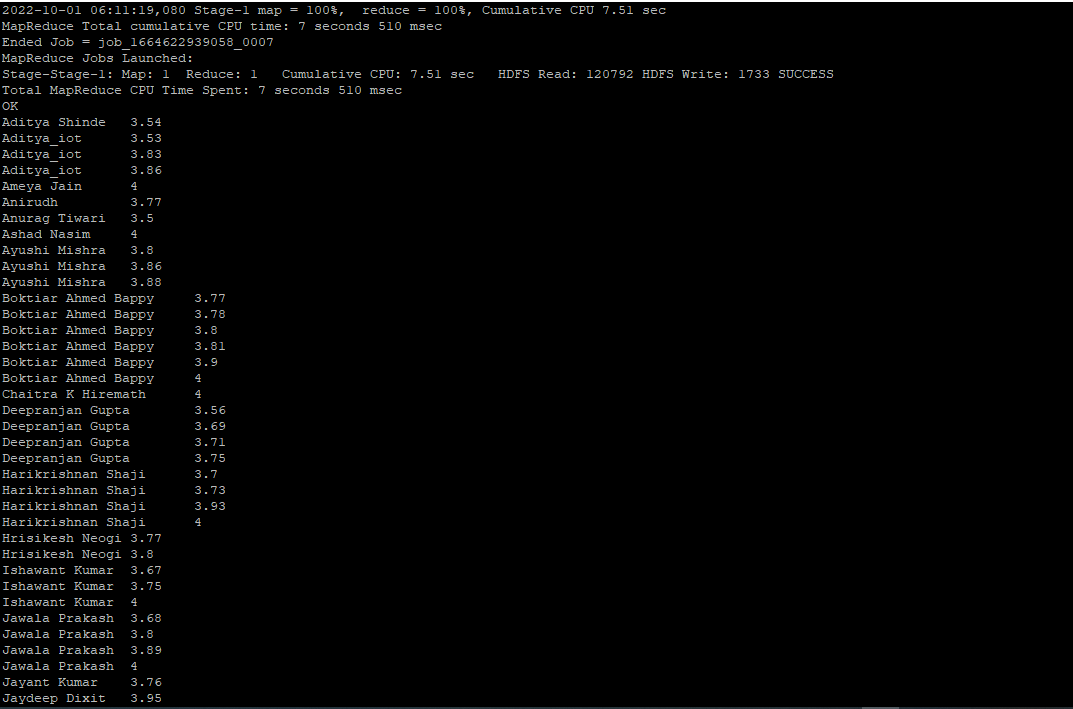
group by Agent\_Name;



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

select Agent\_Name , Average\_Rating from agentperformance where Average\_Rating

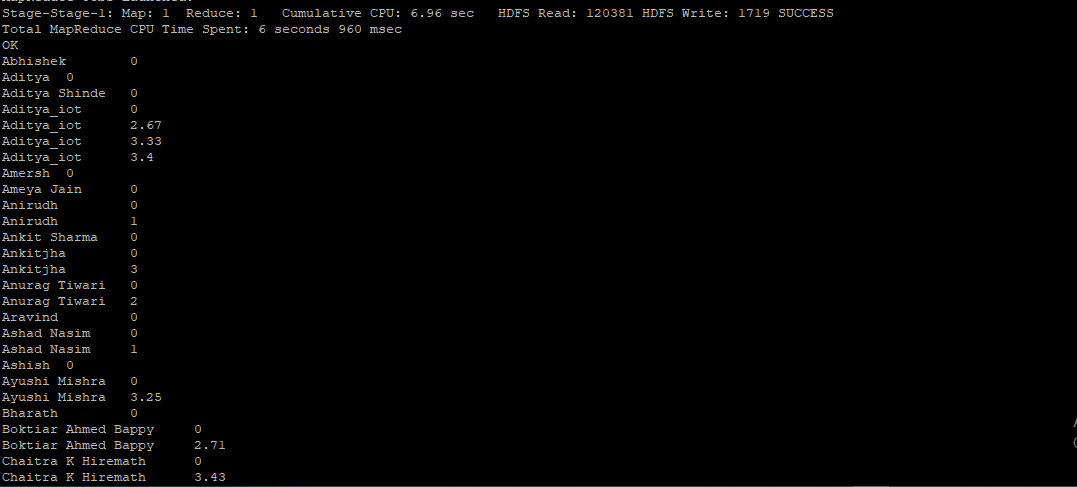
between 3.5 and 4 group by Agent\_Name, Average\_Rating;



1. Agent name who have rating less than 3.5

select Agent\_Name , Average\_Rating from agentperformance where Average\_Rating < 3.5

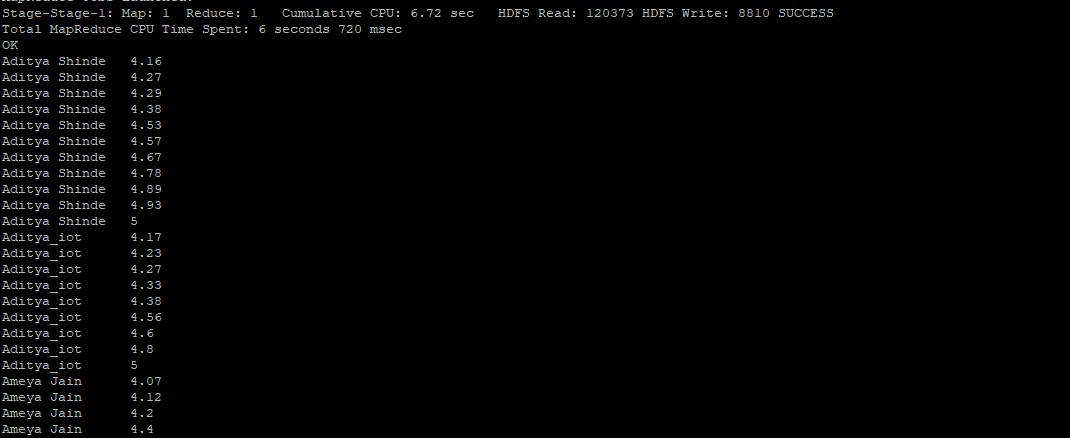
group by Agent\_Name, Average\_Rating;



1. Agent name who have rating more than 4.5

select Agent\_Name , Average\_Rating from agentperformance where Average\_Rating >4.5

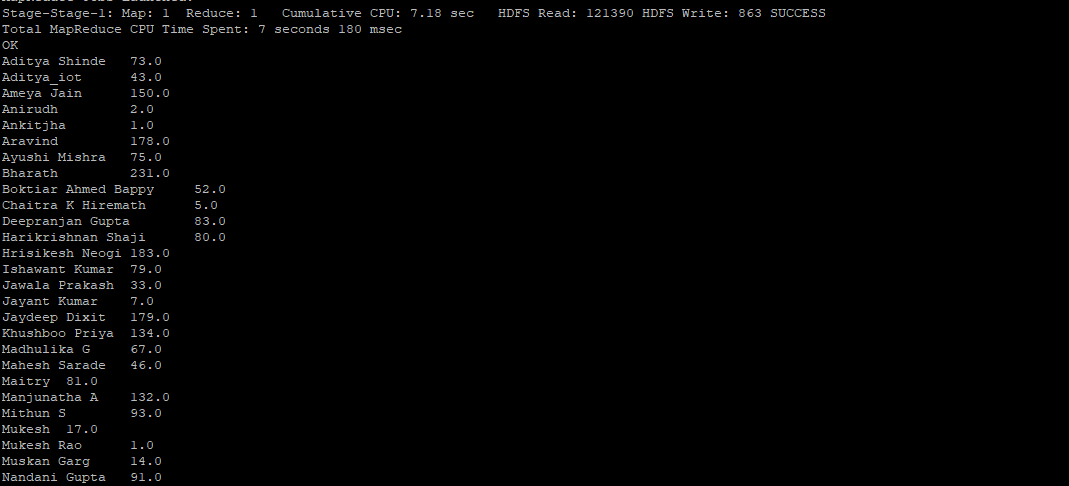
group by Agent\_Name, Average\_Rating;



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

Select Agent\_Name , sum(Total\_Feedback) from agentperformance

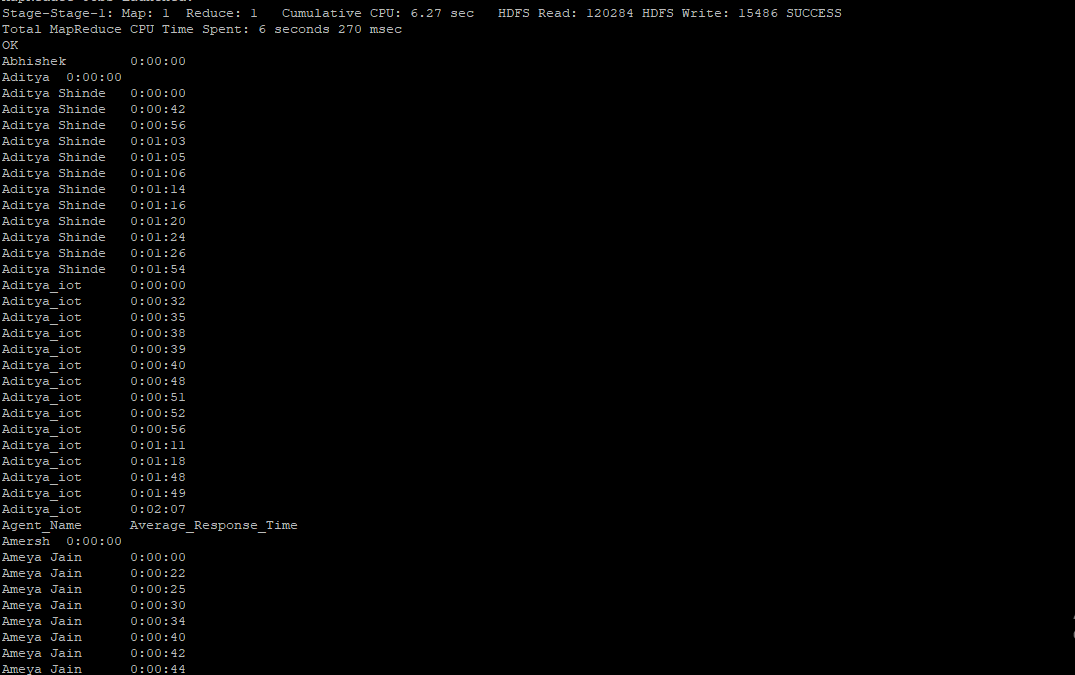
where Average\_Rating > 4.5 group by Agent\_Name;



1. average weekly response time for each agent

Select Agent\_Name, Average\_Response\_Time from agentperformance

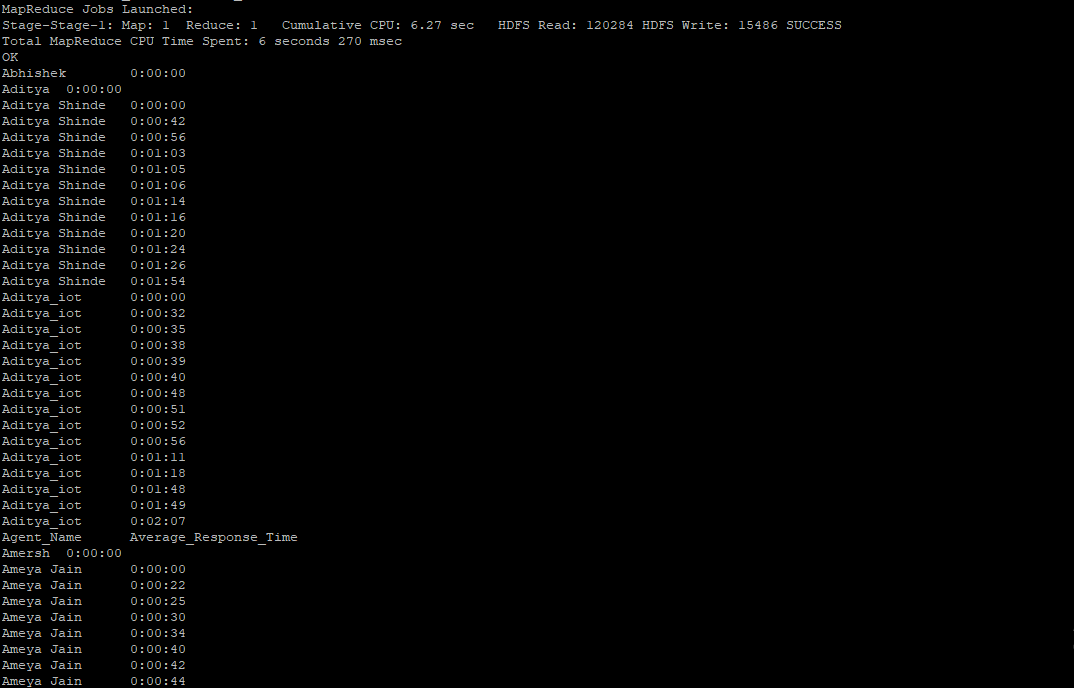
group by Agent\_Name, Average\_Response\_Time;



1. average weekly resolution time for each agents

Select Agent\_Name, Average\_Resolution\_Time from agentperformance

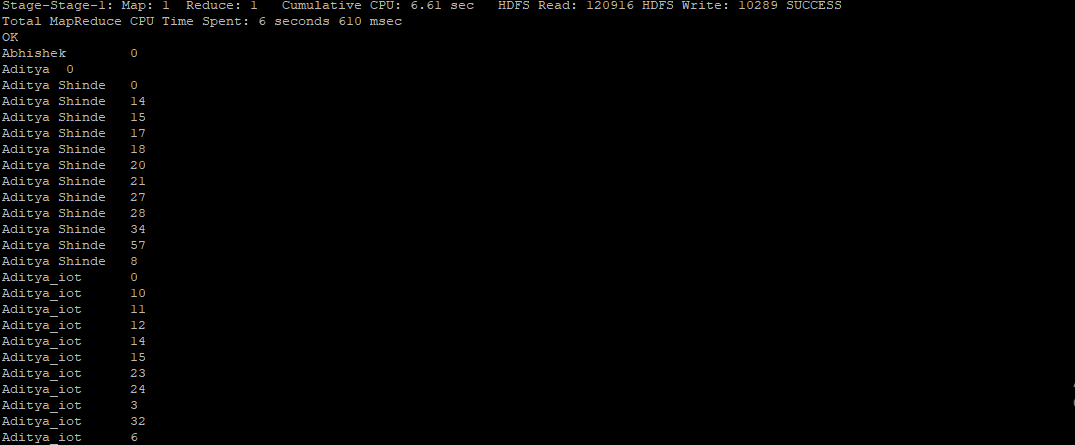
group by Agent\_Name, Average\_Resolution\_Time;



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

select Agent\_Name, Total\_Chats from agentperformance where Total\_Feedback is not null

group by Agent\_Name,Total\_Chats;

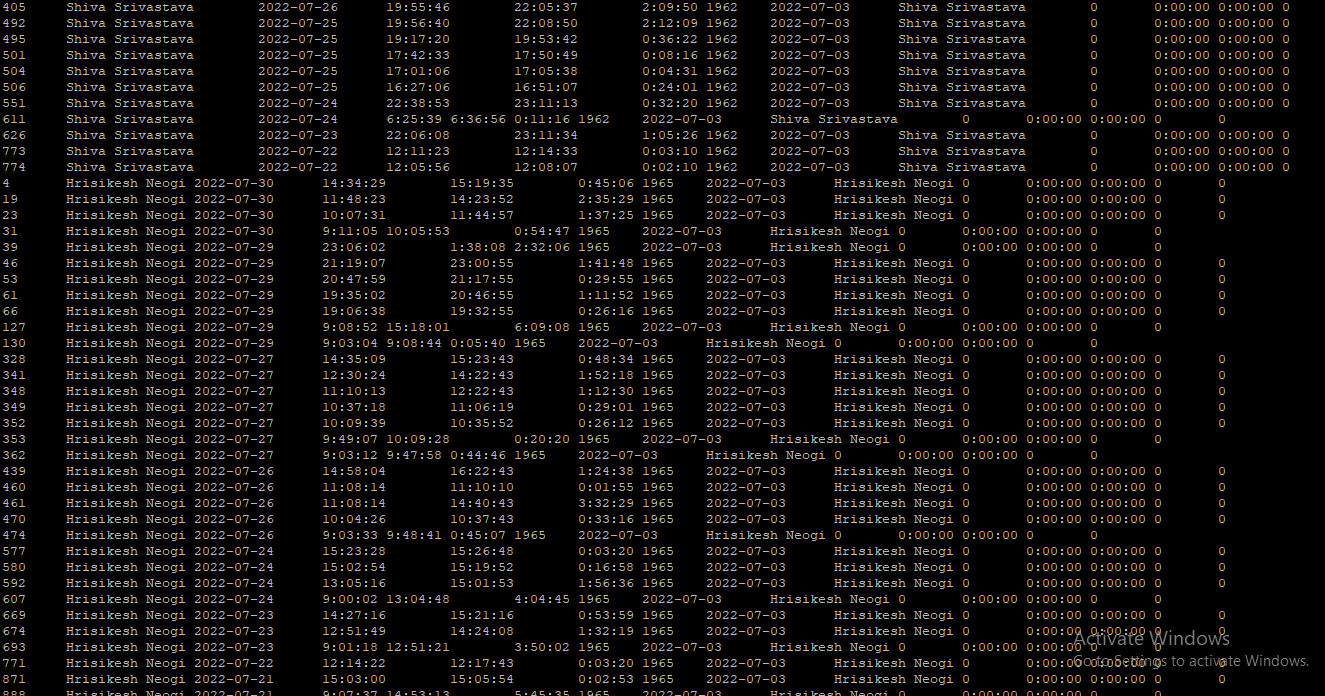


15. Total contribution hour for each and every agents weekly basis

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.

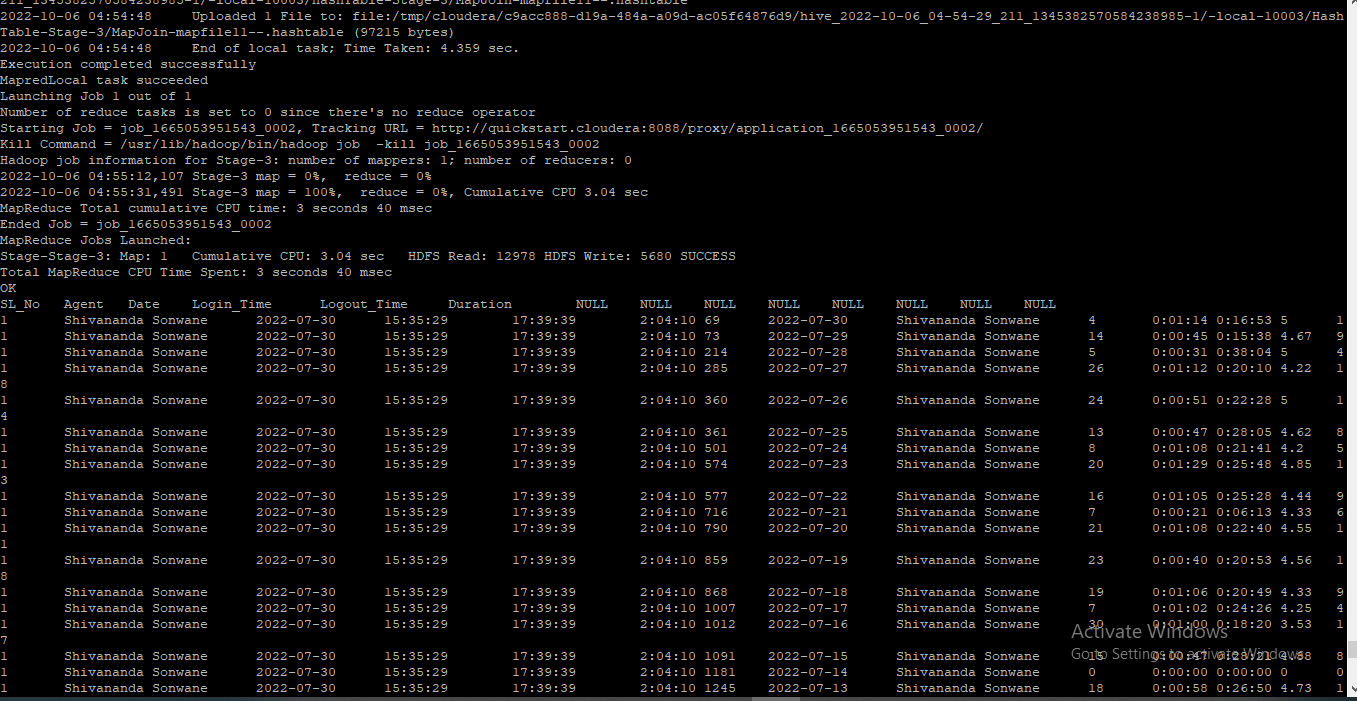
Innerjoin: Select \* from agentlogingreports inner join agentperformance

on agentlogingreports.Agent = agentperformance.Agent\_Name



Left join:

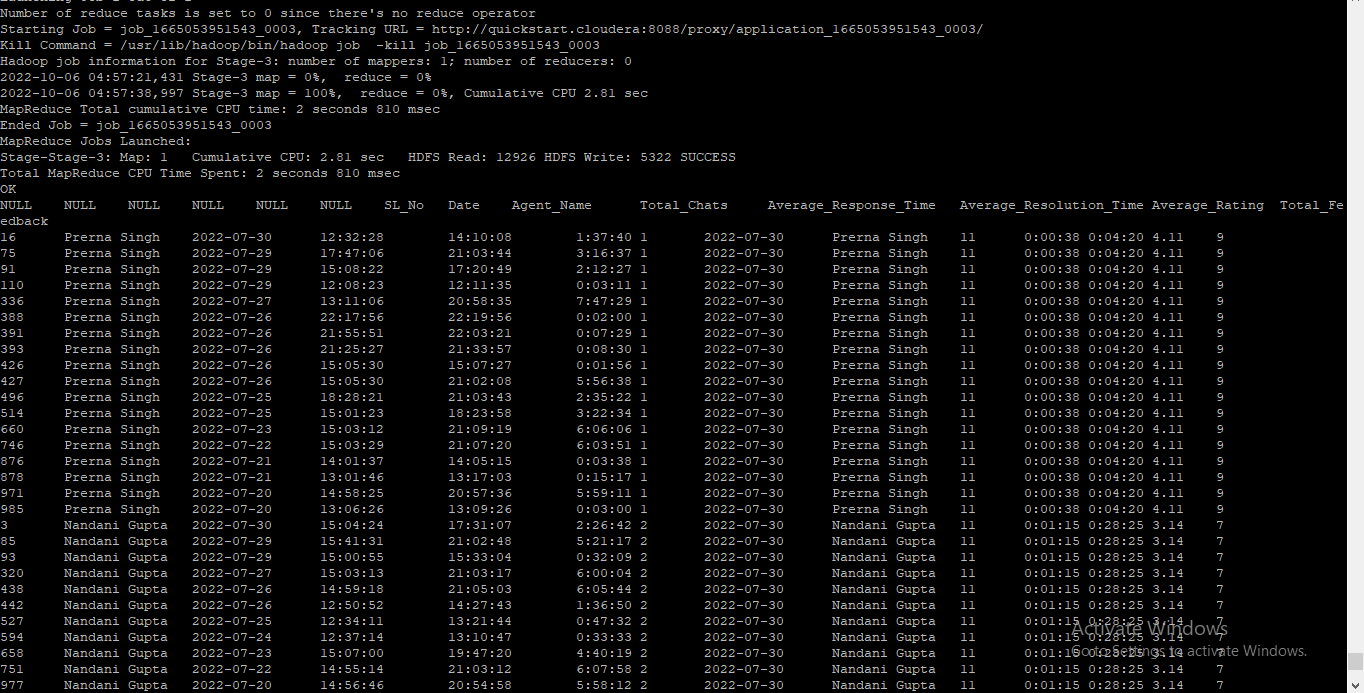
Select \* from agentlogingreports left join agentperformance on agentlogingreports.Agent = agentperformance.Agent\_Name limit 50;



Right join:

Select \* from agentlogingreports right join agentperformance

on agentlogingreports.Agent = agentperformance.Agent\_Name limit 50;



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

Creation of static partition table based on agent

hive> create table agent\_partition

> (

> SL\_No string,

> Date string,

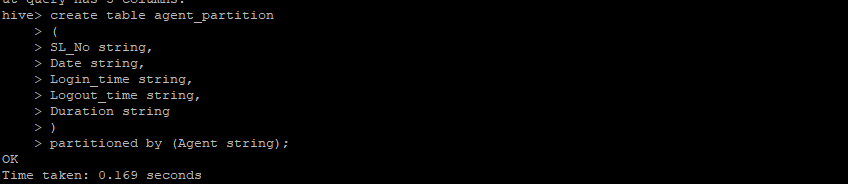
> Login\_time string,

> Logout\_time string,

> Duration string

> )

> partitioned by (Agent string);



Overwriting the data into the newly created table based on the partition criteria

Query: insert overwrite table agent\_partition partition(Agent = 'Mukesh') select sl\_no,date,login\_time,logout\_time,duration from agentlogingreports where agent = 'Mukesh';

