create database miniprojects;

use miniprojects;

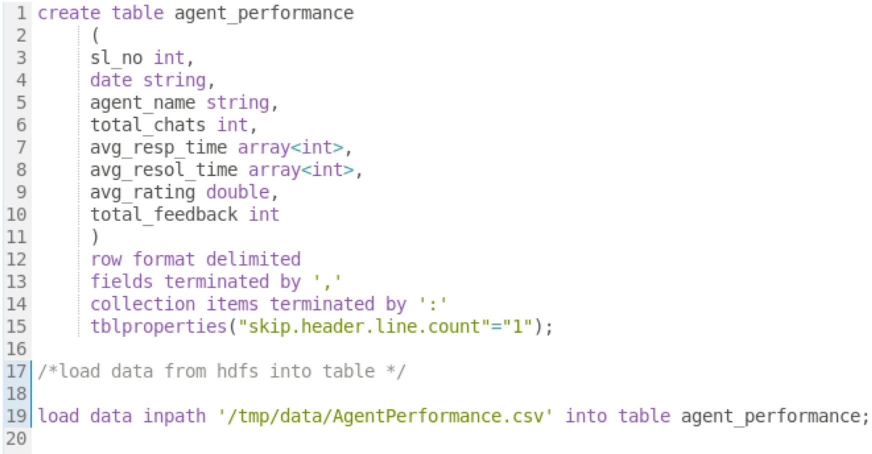
```
hadoop fs -cp file:///home/cloudera/Desktop/AgentPerformance.csv/tmp/data
```

```
hadoop fs -cp
file:///home/cloudera/Desktop/AgentLogingReport.csv/tmp/data
```

create table agent_performance

1.) and 2.) ----------

```
create table agent_performance
(
sl_no int,
date string,
agent_name string,
total_chats int,
avg_resp_time array<int>,
avg_resol_time array<int>,
avg_rating double,
total_feedback int
)
row format delimited
fields terminated by ','
collection items terminated by ':'
tblproperties("skip.header.line.count"="1");
```



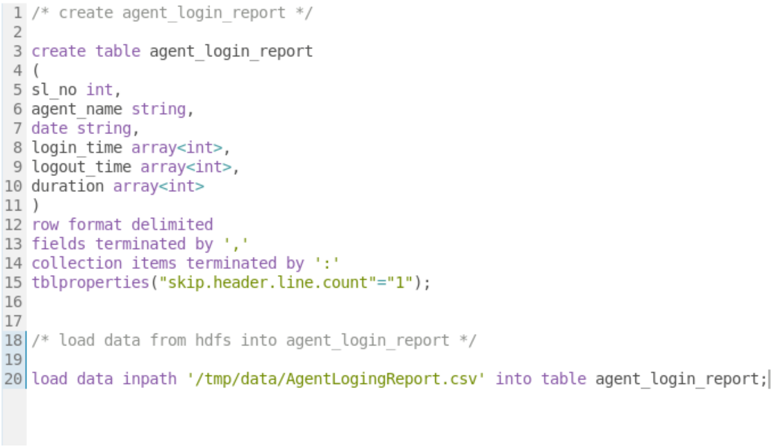
```
create table agent_performance
        (
        sl_no int,
        date string,
        agent_name string,
        total_chats int,
        avg_resp_time array<int>,
        avg_resol_time array<int>,
        avg_rating double,
        total_feedback int
        )
        row format delimited
        fields terminated by ','
        collection items terminated by ':'
        tblproperties("skip.header.line.count"="1");

/*load data from hdfs into table */

load data inpath '/tmp/data/AgentPerformance.csv' into table agent_performance;

```

# load data from hdfs into table

```
load data inpath '/tmp/data/AgentPerformance.csv' into table
agent_performance;
```

# create agent_login_report

```
create table agent_login_report
(
sl_no int,
agent_name string,
date string,
login_time array<int>,
logout_time array<int>,
duration array<int>
)
row format delimited
fields terminated by ','
collection items terminated by ':'
tblproperties("skip.header.line.count"="1");
```



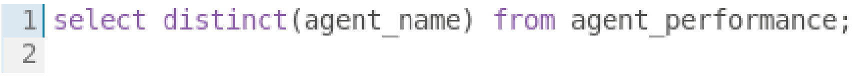
```
/* create agent_login_report */

create table agent_login_report
(
sl_no int,
agent_name string,
date string,
login_time array<int>,
logout_time array<int>,
duration array<int>
)
row format delimited
fields terminated by ','
collection items terminated by ':'
tblproperties("skip.header.line.count"="1");


/* load data from hdfs into agent_login_report */

load data inpath '/tmp/data/AgentLogingReport.csv' into table agent_login_report;
```

# load data from hdfs into agent_login_report

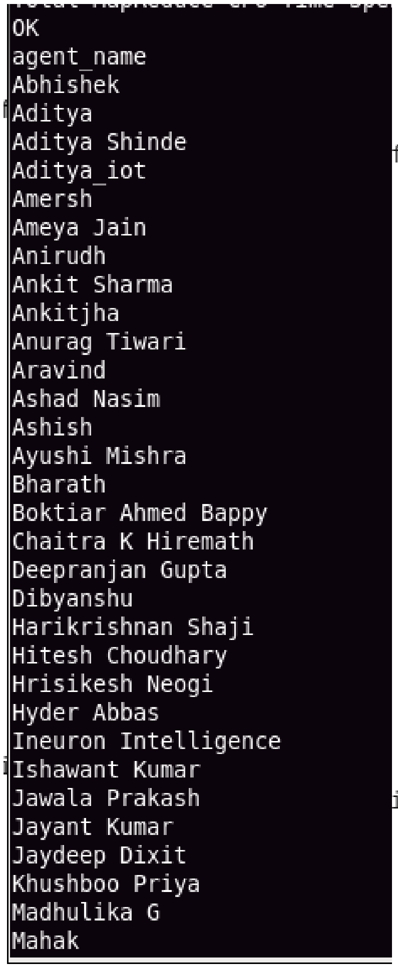
```
load data inpath '/tmp/data/AgentLogingReport.csv' into table
agent_login_report;
```

3.) --------

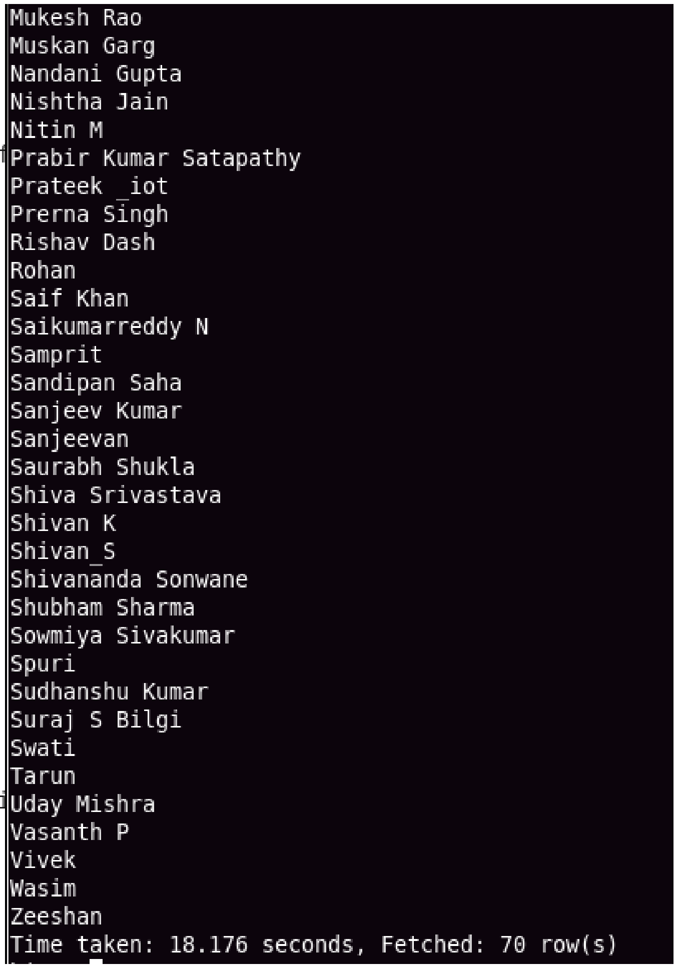
```
select distinct(agent_name) from agent_performance;
```



```
select distinct(agent_name) from agent_performance;

```



```
Total MapReduce CPU Time Spe
OK
agent_name
Abhishek
Aditya
Aditya Shinde
Aditya_iot
Amersh
Ameya Jain
Anirudh
Ankit Sharma
Ankitjha
Anurag Tiwari
Aravind
Ashad Nasim
Ashish
Ayushi Mishra
Bharath
Boktiar Ahmed Bappy
Chaitra K Hiremath
Deepranjan Gupta
Dibyanshu
Harikrishnan Shaji
Hitesh Choudhary
Hrisikesh Neogi
Hyder Abbas
Ineuron Intelligence
Ishawant Kumar
Jawala Prakash
Jayant Kumar
Jaydeep Dixit
Khushboo Priya
Madhulika G
Mahak
```

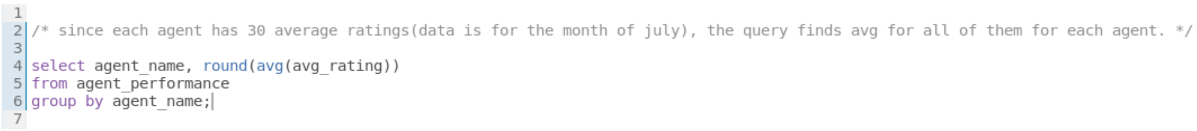


```
Mukesh Rao
Muskan Garg
Nandani Gupta
Nishtha Jain
Nitin M
Prabir Kumar Satapathy
Prateek _iot
Prerna Singh
Rishav Dash
Rohan
Saif Khan
Saikumarreddy N
Samprit
Sandipan Saha
Sanjeev Kumar
Sanjeevan
Saurabh Shukla
Shiva Srivastava
Shivan K
Shivan_S
Shivananda Sonwane
Shubham Sharma
Sowmiya Sivakumar
Spuri
Sudhanshu Kumar
Suraj S Bilgi
Swati
Tarun
Uday Mishra
Vasanth P
Vivek
Wasim
Zeeshan
Time taken: 18.176 seconds, Fetched: 70 row(s)
```

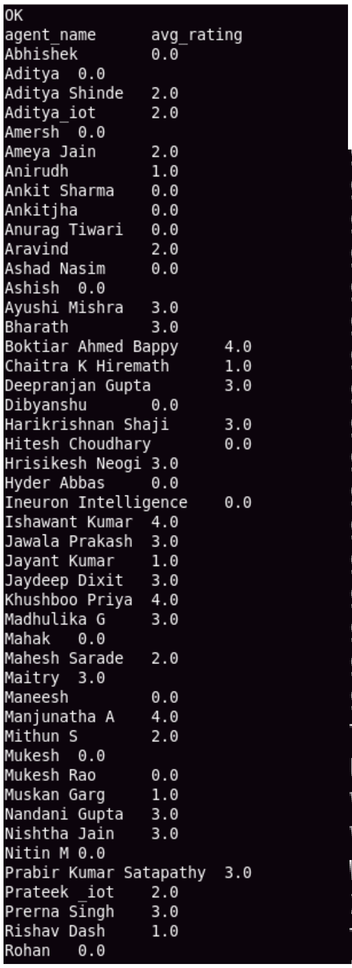
4.) ------

# since each agent has 30 average ratings(data is for the month of july), the query finds avg for all of them for each agent.

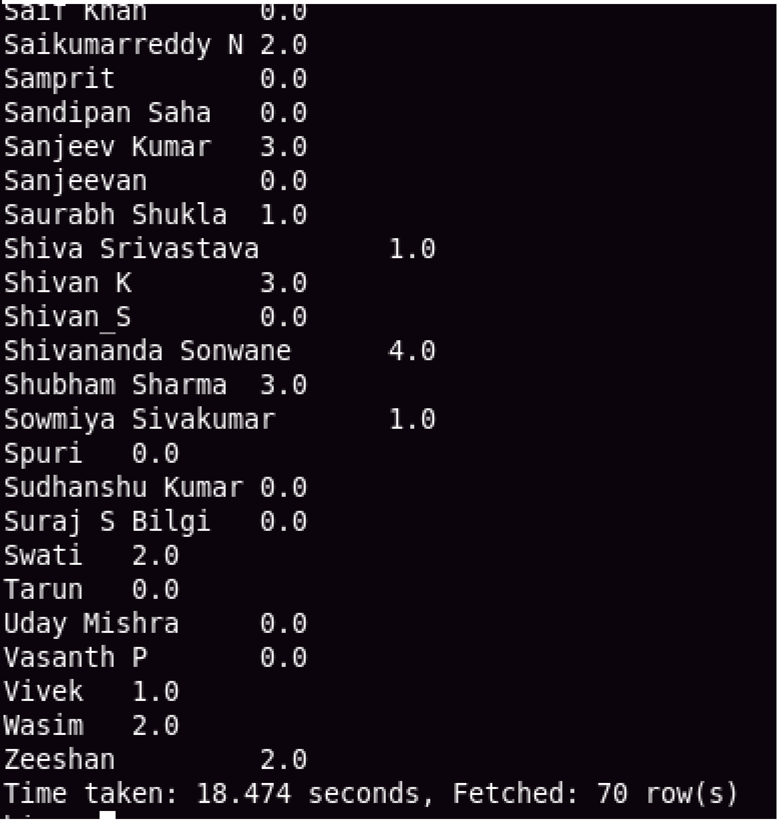
```
select agent_name, round(avg(avg_rating))
from agent_performance
group by agent_name;
```



```

/* since each agent has 30 average ratings(data is for the month of july), the query finds avg for all of them for each agent. */

select agent_name, round(avg(avg_rating))
from agent_performance
group by agent_name;

```



```
OK
agent_name      avg_rating
Abhishek        0.0
Aditya  0.0
Aditya Shinde   2.0
Aditya_iot      2.0
Amersh  0.0
Ameya Jain      2.0
Anirudh         1.0
Ankit Sharma    0.0
Ankitjha        0.0
Anurag Tiwari   0.0
Aravind         2.0
Ashad Nasim     0.0
Ashish  0.0
Ayushi Mishra   3.0
Bharath         3.0
Boktiar Ahmed Bappy     4.0
Chaitra K Hiremath      1.0
Deepranjan Gupta        3.0
Dibyanshu       0.0
Harikrishnan Shaji      3.0
Hitesh Choudhary        0.0
Hrisikesh Neogi 3.0
Hyder Abbas     0.0
Ineuron Intelligence    0.0
Ishawant Kumar  4.0
Jawala Prakash  3.0
Jayant Kumar    1.0
Jaydeep Dixit   3.0
Khushboo Priya  4.0
Madhulika G     3.0
Mahak   0.0
Mahesh Sarade   2.0
Maitry  3.0
Maneesh         0.0
Manjunatha A    4.0
Mithun S        2.0
Mukesh  0.0
Mukesh Rao      0.0
Muskan Garg     1.0
Nandani Gupta   3.0
Nishtha Jain    3.0
Nitin M 0.0
Prabir Kumar Satapathy  3.0
Prateek _iot    2.0
Prerna Singh    3.0
Rishav Dash     1.0
Rohan   0.0
```



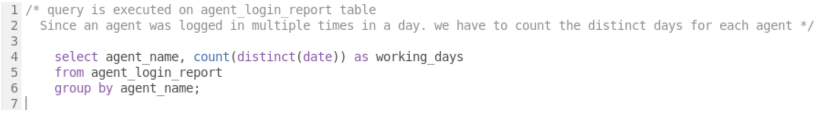
```
Saif Khan       0.0
Saikumarreddy N 2.0
Samprit         0.0
Sandipan Saha   0.0
Sanjeev Kumar   3.0
Sanjeevan       0.0
Saurabh Shukla  1.0
Shiva Srivastava        1.0
Shivan K        3.0
Shivan_S        0.0
Shivananda Sonwane      4.0
Shubham Sharma  3.0
Sowmiya Sivakumar       1.0
Spuri   0.0
Sudhanshu Kumar 0.0
Suraj S Bilgi   0.0
Swati   2.0
Tarun   0.0
Uday Mishra     0.0
Vasanth P       0.0
Vivek   1.0
Wasim   2.0
Zeeshan         2.0
Time taken: 18.474 seconds, Fetched: 70 row(s)
```

5.)--------

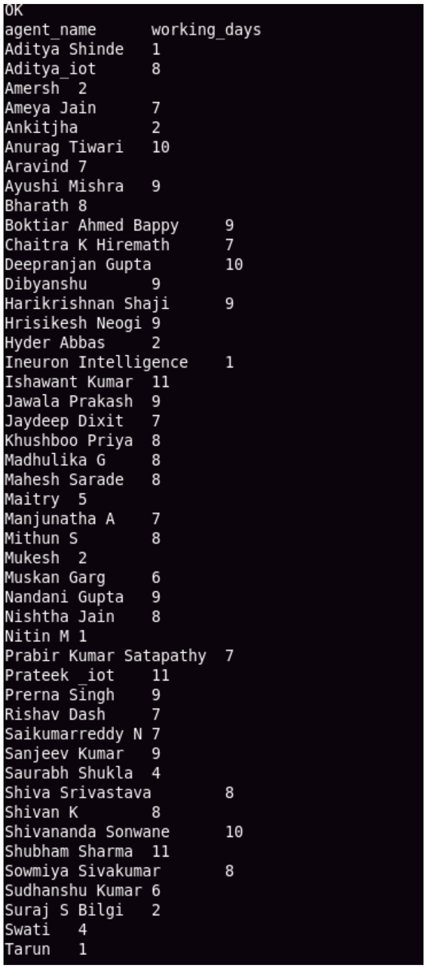
# query is executed on agent_login_report table

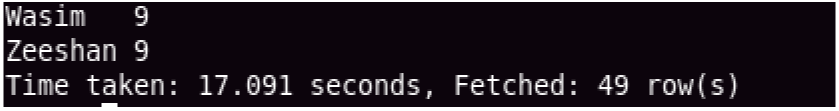
# Since an agent was logged in multiple times in a day. we have to count the distinct days for each agent

```sql
select agent_name, count(distinct(date)) as working_days
from agent_login_report
group by agent_name;
```



```
/* query is executed on agent_login_report table
  Since an agent was logged in multiple times in a day. we have to count the distinct days for each agent */

    select agent_name, count(distinct(date)) as working_days
    from agent_login_report
    group by agent_name;
```



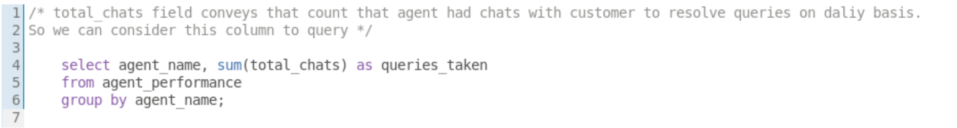


```
OK
agent_name      working_days
Aditya Shinde   1
Aditya_iot      8
Amersh  2
Ameya Jain      7
Ankitjha        2
Anurag Tiwari   10
Aravind 7
Ayushi Mishra   9
Bharath 8
Boktiar Ahmed Bappy     9
Chaitra K Hiremath      7
Deepranjan Gupta        10
Dibyanshu       9
Harikrishnan Shaji      9
Hrisikesh Neogi 9
Hyder Abbas     2
Ineuron Intelligence    1
Ishawant Kumar  11
Jawala Prakash  9
Jaydeep Dixit   7
Khushboo Priya  8
Madhulika G     8
Mahesh Sarade   8
Maitry  5
Manjunatha A    7
Mithun S        8
Mukesh  2
Muskan Garg     6
Nandani Gupta   9
Nishtha Jain    8
Nitin M 1
Prabir Kumar Satapathy  7
Prateek _iot    11
Prerna Singh    9
Rishav Dash     7
Saikumarreddy N 7
Sanjeev Kumar   9
Saurabh Shukla  4
Shiva Srivastava        8
Shivan K        8
Shivananda Sonwane      10
Shubham Sharma  11
Sowmiya Sivakumar       8
Sudhanshu Kumar 6
Suraj S Bilgi   2
Swati   4
Tarun   1
Wasim   9
Zeeshan 9
Time taken: 17.091 seconds, Fetched: 49 row(s)
```

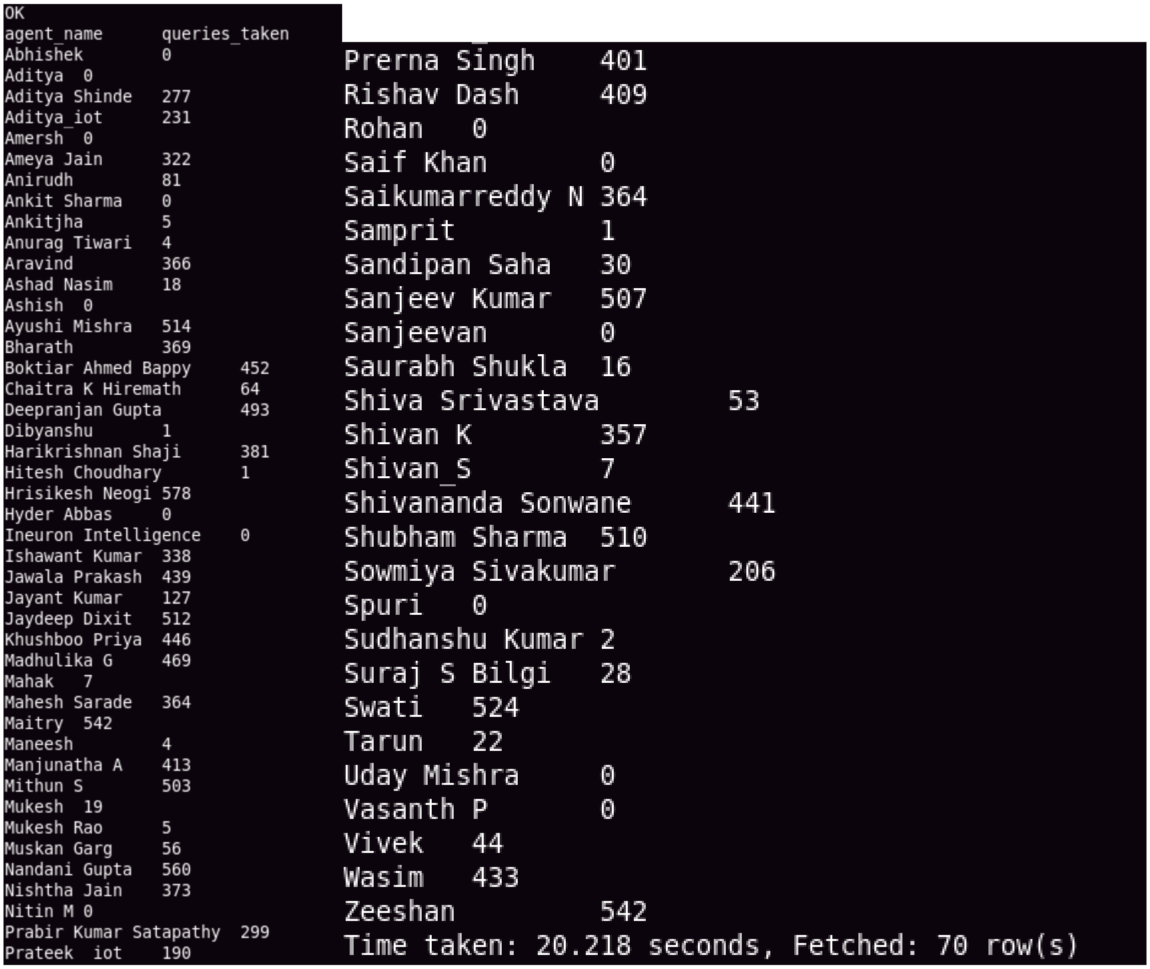
6.)------

# total_chats field conveys that count that agent had chats with customer to resolve queries on daliy basis. So we can consider this column to query

```
select agent_name, sum(total_chats) as queries_taken
from agent_performance
group by agent_name;
```



```
/* total_chats field conveys that count that agent had chats with customer to resolve queries on daliy basis.
So we can consider this column to query */

    select agent_name, sum(total_chats) as queries_taken
    from agent_performance
    group by agent_name;
```

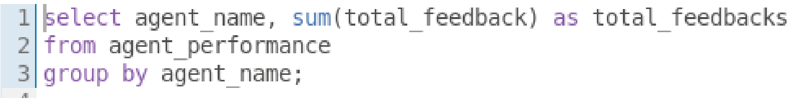


```
OK
agent_name      queries_taken
Abhishek        0
Aditya  0
Aditya Shinde   277
Aditya_iot      231
Amersh  0
Ameya Jain      322
Anirudh         81
Ankit Sharma    0
Ankitjha        5
Anurag Tiwari   4
Aravind         366
Ashad Nasim     18
Ashish  0
Ayushi Mishra   514
Bharath         369
Boktiar Ahmed Bappy     452
Chaitra K Hiremath      64
Deepranjan Gupta        493
Dibyanshu       1
Harikrishnan Shaji      381
Hitesh Choudhary        1
Hrisikesh Neogi 578
Hyder Abbas     0
Ineuron Intelligence    0
Ishawant Kumar  338
Jawala Prakash  439
Jayant Kumar    127
Jaydeep Dixit   512
Khushboo Priya  446
Madhulika G     469
Mahak   7
Mahesh Sarade   364
Maitry  542
Maneesh         4
Manjunatha A    413
Mithun S        503
Mukesh  19
Mukesh Rao      5
Muskan Garg     56
Nandani Gupta   560
Nishtha Jain    373
Nitin M 0
Prabir Kumar Satapathy  299
Prateek  iot    190
```

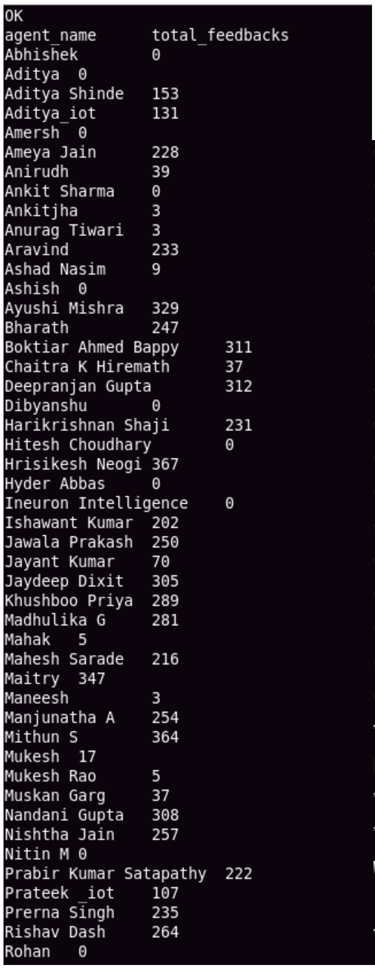
```
Prerna Singh    401
Rishav Dash     409
Rohan   0
Saif Khan       0
Saikumarreddy N 364
Samprit         1
Sandipan Saha   30
Sanjeev Kumar   507
Sanjeevan       0
Saurabh Shukla  16
Shiva Srivastava        53
Shivan K        357
Shivan_S        7
Shivananda Sonwane      441
Shubham Sharma  510
Sowmiya Sivakumar       206
Spuri   0
Sudhanshu Kumar 2
Suraj S Bilgi   28
Swati   524
Tarun   22
Uday Mishra     0
Vasanth P       0
Vivek   44
Wasim   433
Zeeshan         542
Time taken: 20.218 seconds, Fetched: 70 row(s)
```

7.) ------

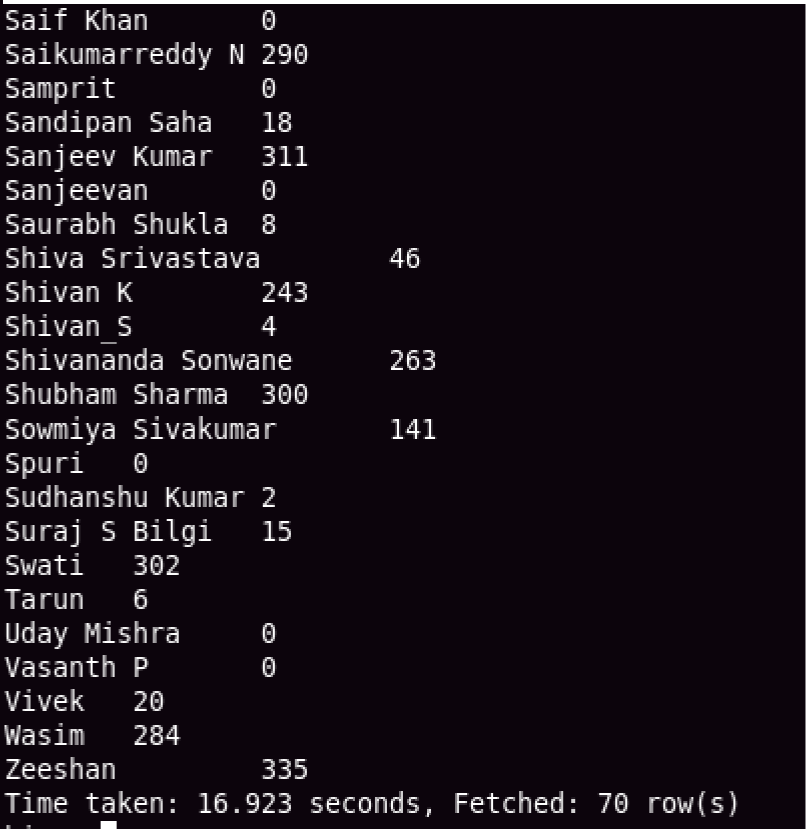
```
select agent_name, sum(total_feedback) as total_feedbacks
from agent_performance
group by agent_name;
```



```
select agent_name, sum(total_feedback) as total_feedbacks
from agent_performance
group by agent_name;
```



```
OK
agent_name      total_feedbacks
Abhishek        0
Aditya  0
Aditya Shinde   153
Aditya_iot      131
Amersh  0
Ameya Jain      228
Anirudh         39
Ankit Sharma    0
Ankitjha        3
Anurag Tiwari   3
Aravind         233
Ashad Nasim     9
Ashish  0
Ayushi Mishra   329
Bharath         247
Boktiar Ahmed Bappy     311
Chaitra K Hiremath      37
Deepranjan Gupta        312
Dibyanshu       0
Harikrishnan Shaji      231
Hitesh Choudhary        0
Hrisikesh Neogi 367
Hyder Abbas     0
Ineuron Intelligence    0
Ishawant Kumar  202
Jawala Prakash  250
Jayant Kumar    70
Jaydeep Dixit   305
Khushboo Priya  289
Madhulika G     281
Mahak   5
Mahesh Sarade   216
Maitry  347
Maneesh         3
Manjunatha A    254
Mithun S        364
Mukesh  17
Mukesh Rao      5
Muskan Garg     37
Nandani Gupta   308
Nishtha Jain    257
Nitin M 0
Prabir Kumar Satapathy  222
Prateek _iot    107
Prerna Singh    235
Rishav Dash     264
Rohan   0
```

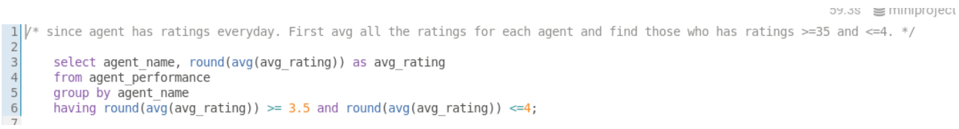


```
Saif Khan       0
Saikumarreddy N 290
Samprit         0
Sandipan Saha   18
Sanjeev Kumar   311
Sanjeevan       0
Saurabh Shukla  8
Shiva Srivastava        46
Shivan K        243
Shivan_S        4
Shivananda Sonwane      263
Shubham Sharma  300
Sowmiya Sivakumar       141
Spuri   0
Sudhanshu Kumar 2
Suraj S Bilgi   15
Swati   302
Tarun   6
Uday Mishra     0
Vasanth P       0
Vivek   20
Wasim   284
Zeeshan         335
Time taken: 16.923 seconds, Fetched: 70 row(s)
```

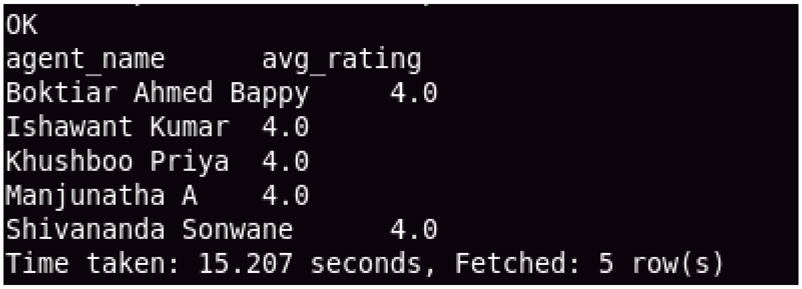
8.) -------

# since agent has ratings everyday. First avg all the ratings for each agent and find those who has ratings >=35 and <=4.

```
select agent_name, round(avg(avg_rating)) as avg_rating
from agent_performance
group by agent_name
having round(avg(avg_rating)) >= 3.5 and round(avg(avg_rating)) <=4;
```



```
/* since agent has ratings everyday. First avg all the ratings for each agent and find those who has ratings >=35 and <=4. */

    select agent_name, round(avg(avg_rating)) as avg_rating
    from agent_performance
    group by agent_name
    having round(avg(avg_rating)) >= 3.5 and round(avg(avg_rating)) <=4;
```

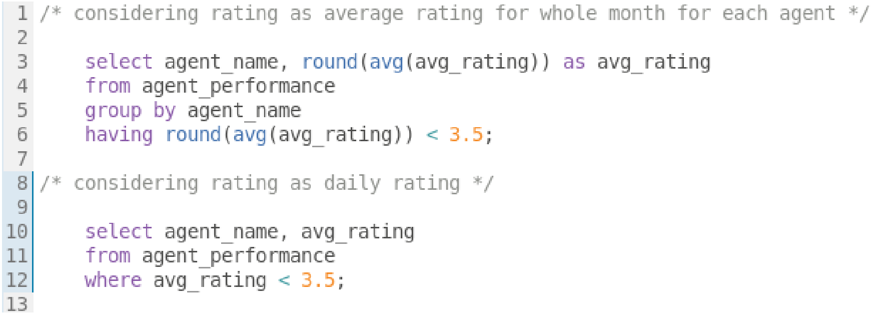


```
OK
agent_name      avg_rating
Boktiar Ahmed Bappy     4.0
Ishawant Kumar  4.0
Khushboo Priya  4.0
Manjunatha A    4.0
Shivananda Sonwane      4.0
Time taken: 15.207 seconds, Fetched: 5 row(s)
```

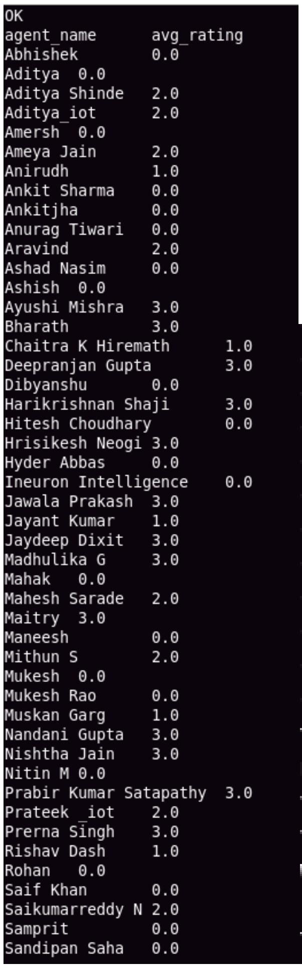
9.) ---------

# considering rating as average rating for whole month for each agent

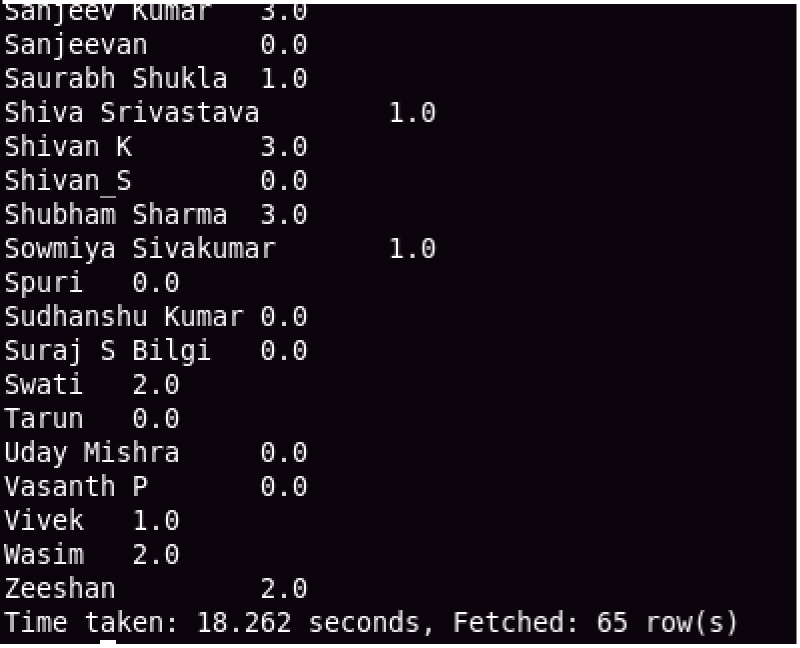
```
select agent_name, round(avg(avg_rating)) as avg_rating
from agent_performance
group by agent_name
having round(avg(avg_rating)) < 3.5;
```



```
/* considering rating as average rating for whole month for each agent */

    select agent_name, round(avg(avg_rating)) as avg_rating
    from agent_performance
    group by agent_name
    having round(avg(avg_rating)) < 3.5;

/* considering rating as daily rating */

    select agent_name, avg_rating
    from agent_performance
    where avg_rating < 3.5;
```



```
OK
agent_name      avg_rating
Abhishek        0.0
Aditya  0.0
Aditya Shinde   2.0
Aditya_iot      2.0
Amersh  0.0
Ameya Jain      2.0
Anirudh         1.0
Ankit Sharma    0.0
Ankitjha        0.0
Anurag Tiwari   0.0
Aravind         2.0
Ashad Nasim     0.0
Ashish  0.0
Ayushi Mishra   3.0
Bharath         3.0
Chaitra K Hiremath      1.0
Deepranjan Gupta        3.0
Dibyanshu       0.0
Harikrishnan Shaji      3.0
Hitesh Choudhary        0.0
Hrisikesh Neogi 3.0
Hyder Abbas     0.0
Ineuron Intelligence    0.0
Jawala Prakash  3.0
Jayant Kumar    1.0
Jaydeep Dixit   3.0
Madhulika G     3.0
Mahak   0.0
Mahesh Sarade   2.0
Maitry  3.0
Maneesh         0.0
Mithun S        2.0
Mukesh  0.0
Mukesh Rao      0.0
Muskan Garg     1.0
Nandani Gupta   3.0
Nishtha Jain    3.0
Nitin M 0.0
Prabir Kumar Satapathy  3.0
Prateek _iot    2.0
Prerna Singh    3.0
Rishav Dash     1.0
Rohan   0.0
Saif Khan       0.0
Saikumarreddy N 2.0
Samprit         0.0
Sandipan Saha   0.0
```



```
Sanjeev Kumar   3.0
Sanjeevan       0.0
Saurabh Shukla  1.0
Shiva Srivastava        1.0
Shivan K        3.0
Shivan_S        0.0
Shubham Sharma  3.0
Sowmiya Sivakumar       1.0
Spuri   0.0
Sudhanshu Kumar 0.0
Suraj S Bilgi   0.0
Swati   2.0
Tarun   0.0
Uday Mishra     0.0
Vasanth P       0.0
Vivek   1.0
Wasim   2.0
Zeeshan         2.0
Time taken: 18.262 seconds, Fetched: 65 row(s)
```

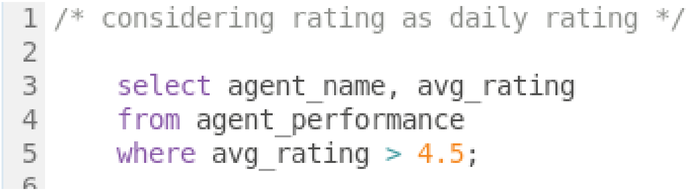
# considering rating as daily rating

```
select agent_name, avg_rating
from agent_performance
where avg_rating < 3.5;
```

10.) -------

# considering rating as daily rating

```
select agent_name, avg_rating
from agent_performance
where avg_rating > 4.5;
```

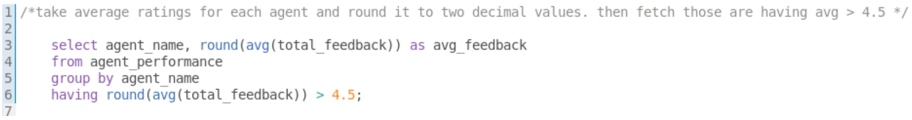


```
/* considering rating as daily rating */

    select agent_name, avg_rating
    from agent_performance
    where avg_rating > 4.5;
```

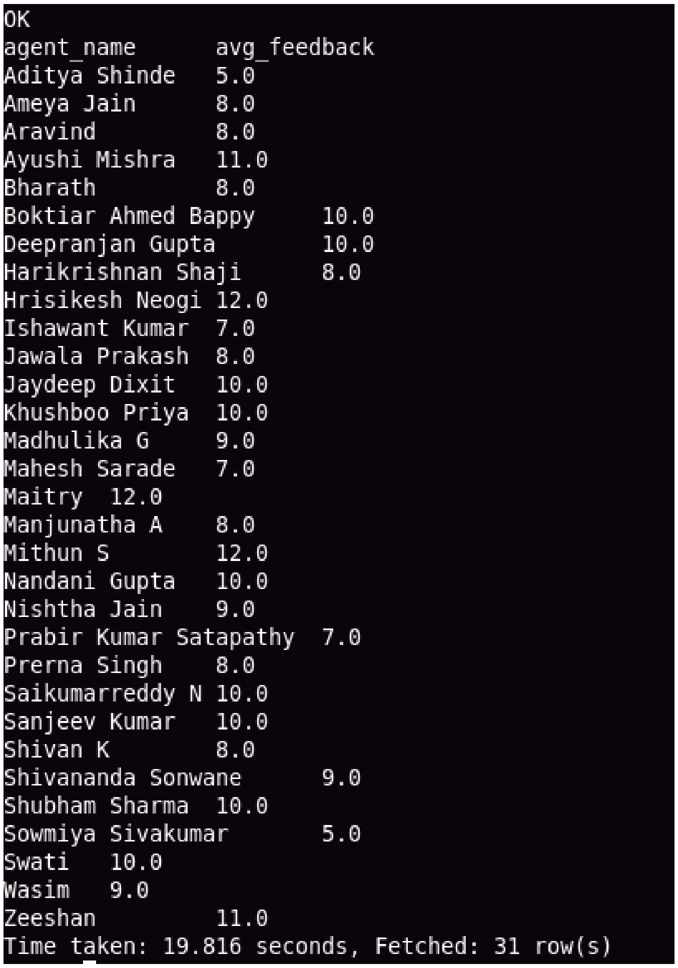
11.)

# take average ratings for each agent and round it to two decimal values. then fetch those are having avg > 4.5

```
select agent_name, round(avg(total_feedback)) as avg_feedback
from agent_performance
group by agent_name
having round(avg(total_feedback)) > 4.5;
```



```
/*take average ratings for each agent and round it to two decimal values. then fetch those are having avg > 4.5 */

    select agent_name, round(avg(total_feedback)) as avg_feedback
    from agent_performance
    group by agent_name
    having round(avg(total_feedback)) > 4.5;
```

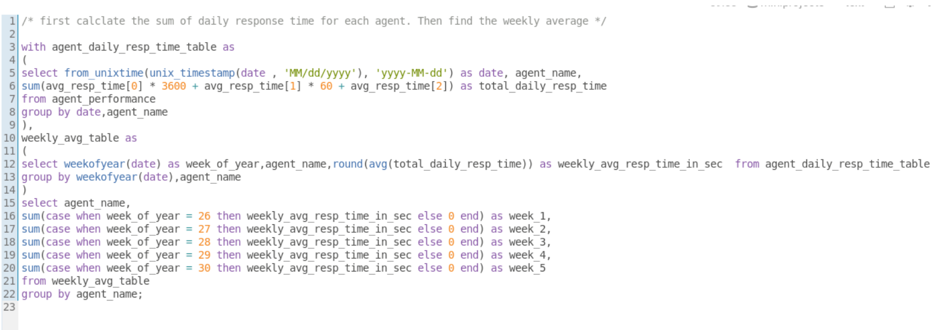


```
OK
agent_name      avg_feedback
Aditya Shinde   5.0
Ameya Jain      8.0
Aravind         8.0
Ayushi Mishra   11.0
Bharath         8.0
Boktiar Ahmed Bappy     10.0
Deepranjan Gupta        10.0
Harikrishnan Shaji      8.0
Hrisikesh Neogi 12.0
Ishawant Kumar  7.0
Jawala Prakash  8.0
Jaydeep Dixit   10.0
Khushboo Priya  10.0
Madhulika G     9.0
Mahesh Sarade   7.0
Maitry  12.0
Manjunatha A    8.0
Mithun S        12.0
Nandani Gupta   10.0
Nishtha Jain    9.0
Prabir Kumar Satapathy  7.0
Prerna Singh    8.0
Saikumarreddy N 10.0
Sanjeev Kumar   10.0
Shivan K        8.0
Shivananda Sonwane      9.0
Shubham Sharma  10.0
Sowmiya Sivakumar       5.0
Swati   10.0
Wasim   9.0
Zeeshan         11.0
Time taken: 19.816 seconds, Fetched: 31 row(s)
```

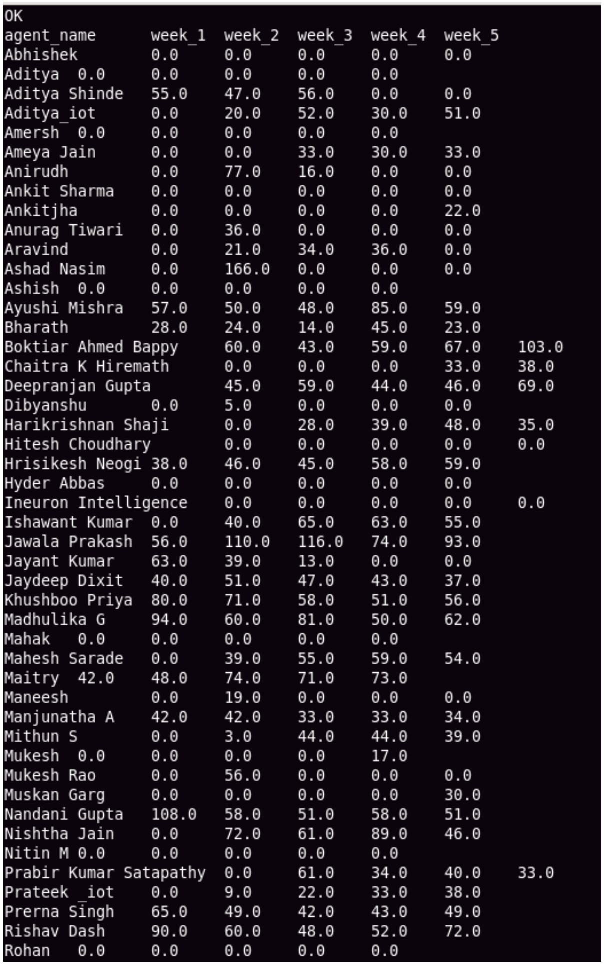
12.) -------

# first calclate the sum of daily response time for each agent. Then find the weekly average.

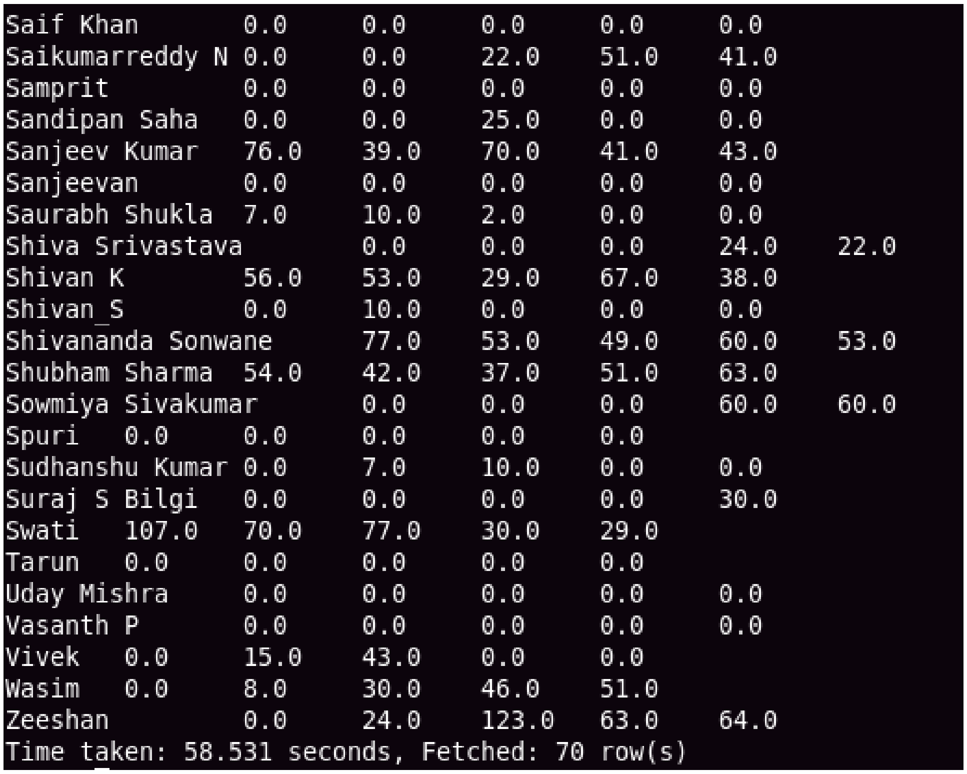
```
with agent_daily_resp_time_table as
(
select from_unixtime(unix_timestamp(date , 'MM/dd/yyyy'), 'yyyy-MM-dd') as
date, agent_name,
sum(avg_resp_time[0] * 3600 + avg_resp_time[1] * 60 + avg_resp_time[2]) as
total_daily_resp_time
from agent_performance
group by date,agent_name
),

weekly_avg_table as
(
select weekofyear(date) as
week_of_year,agent_name,round(avg(total_daily_resp_time)) as
weekly_avg_resp_time_in_sec  from agent_daily_resp_time_table
group by weekofyear(date),agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg_resp_time_in_sec else 0 end)
as week_1,
sum(case when week_of_year = 27 then weekly_avg_resp_time_in_sec else 0 end)
as week_2,
sum(case when week_of_year = 28 then weekly_avg_resp_time_in_sec else 0 end)
as week_3,
sum(case when week_of_year = 29 then weekly_avg_resp_time_in_sec else 0 end)
as week_4,
sum(case when week_of_year = 30 then weekly_avg_resp_time_in_sec else 0 end)
as week_5
from weekly_avg_table
group by agent_name;
```



```
/* first calclate the sum of daily response time for each agent. Then find the weekly average */

with agent_daily_resp_time_table as
(
select from_unixtime(unix_timestamp(date , 'MM/dd/yyyy'), 'yyyy-MM-dd') as date, agent_name,
sum(avg_resp_time[0] * 3600 + avg_resp_time[1] * 60 + avg_resp_time[2]) as total_daily_resp_time
from agent_performance
group by date,agent_name
),
weekly_avg_table as
(
select weekofyear(date) as week_of_year,agent_name,round(avg(total_daily_resp_time)) as weekly_avg_resp_time_in_sec  from agent_daily_resp_time_table
group by weekofyear(date),agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg_resp_time_in_sec else 0 end) as week_1,
sum(case when week_of_year = 27 then weekly_avg_resp_time_in_sec else 0 end) as week_2,
sum(case when week_of_year = 28 then weekly_avg_resp_time_in_sec else 0 end) as week_3,
sum(case when week_of_year = 29 then weekly_avg_resp_time_in_sec else 0 end) as week_4,
sum(case when week_of_year = 30 then weekly_avg_resp_time_in_sec else 0 end) as week_5
from weekly_avg_table
group by agent_name;
```



```
OK
agent_name      week_1  week_2  week_3  week_4  week_5
Abhishek        0.0     0.0     0.0     0.0     0.0
Aditya  0.0     0.0     0.0     0.0     0.0
Aditya Shinde   55.0    47.0    56.0    0.0     0.0
Aditya_iot      0.0     20.0    52.0    30.0    51.0
Amersh  0.0     0.0     0.0     0.0     0.0
Ameya Jain      0.0     0.0     33.0    30.0    33.0
Anirudh         0.0     77.0    16.0    0.0     0.0
Ankit Sharma    0.0     0.0     0.0     0.0     0.0
Ankitjha        0.0     0.0     0.0     0.0     22.0
Anurag Tiwari   0.0     36.0    0.0     0.0     0.0
Aravind         0.0     21.0    34.0    36.0    0.0
Ashad Nasim     0.0     166.0   0.0     0.0     0.0
Ashish  0.0     0.0     0.0     0.0     0.0
Ayushi Mishra   57.0    50.0    48.0    85.0    59.0
Bharath         28.0    24.0    14.0    45.0    23.0
Boktiar Ahmed Bappy     60.0    43.0    59.0    67.0    103.0
Chaitra K Hiremath      0.0     0.0     0.0     33.0    38.0
Deepranjan Gupta        45.0    59.0    44.0    46.0    69.0
Dibyanshu       0.0     5.0     0.0     0.0     0.0
Harikrishnan Shaji      0.0     28.0    39.0    48.0    35.0
Hitesh Choudhary        0.0     0.0     0.0     0.0     0.0
Hrisikesh Neogi 38.0    46.0    45.0    58.0    59.0
Hyder Abbas     0.0     0.0     0.0     0.0     0.0
Ineuron Intelligence    0.0     0.0     0.0     0.0     0.0
Ishawant Kumar  0.0     40.0    65.0    63.0    55.0
Jawala Prakash  56.0    110.0   116.0   74.0    93.0
Jayant Kumar    63.0    39.0    13.0    0.0     0.0
Jaydeep Dixit   40.0    51.0    47.0    43.0    37.0
Khushboo Priya  80.0    71.0    58.0    51.0    56.0
Madhulika G     94.0    60.0    81.0    50.0    62.0
Mahak   0.0     0.0     0.0     0.0     0.0
Mahesh Sarade   0.0     39.0    55.0    59.0    54.0
Maitry  42.0    48.0    74.0    71.0    73.0
Maneesh         0.0     19.0    0.0     0.0     0.0
Manjunatha A    42.0    42.0    33.0    33.0    34.0
Mithun S        0.0     3.0     44.0    44.0    39.0
Mukesh  0.0     0.0     0.0     0.0     17.0
Mukesh Rao      0.0     56.0    0.0     0.0     0.0
Muskan Garg     0.0     0.0     0.0     0.0     30.0
Nandani Gupta   108.0   58.0    51.0    58.0    51.0
Nishtha Jain    0.0     72.0    61.0    89.0    46.0
Nitin M 0.0     0.0     0.0     0.0     0.0
Prabir Kumar Satapathy  0.0     61.0    34.0    40.0    33.0
Prateek _iot    0.0     9.0     22.0    33.0    38.0
Prerna Singh    65.0    49.0    42.0    43.0    49.0
Rishav Dash     90.0    60.0    48.0    52.0    72.0
Rohan   0.0     0.0     0.0     0.0     0.0
```

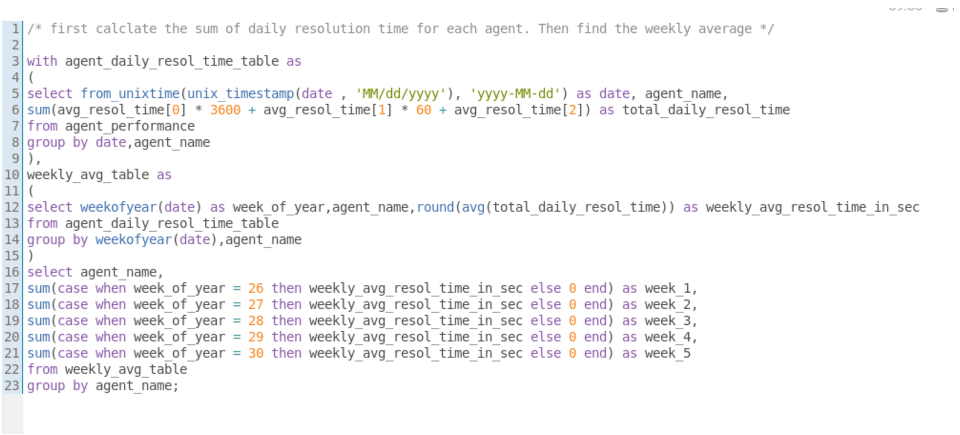


```
Saif Khan       0.0     0.0     0.0     0.0     0.0
Saikumarreddy N 0.0     0.0     22.0    51.0    41.0
Samprit         0.0     0.0     0.0     0.0     0.0
Sandipan Saha   0.0     0.0     25.0    0.0     0.0
Sanjeev Kumar   76.0    39.0    70.0    41.0    43.0
Sanjeevan       0.0     0.0     0.0     0.0     0.0
Saurabh Shukla  7.0     10.0    2.0     0.0     0.0
Shiva Srivastava        0.0     0.0     0.0     24.0    22.0
Shivan K        56.0    53.0    29.0    67.0    38.0
Shivan_S        0.0     10.0    0.0     0.0     0.0
Shivananda Sonwane      77.0    53.0    49.0    60.0    53.0
Shubham Sharma  54.0    42.0    37.0    51.0    63.0
Sowmiya Sivakumar       0.0     0.0     0.0     60.0    60.0
Spuri   0.0     0.0     0.0     0.0     0.0
Sudhanshu Kumar 0.0     7.0     10.0    0.0     0.0
Suraj S Bilgi   0.0     0.0     0.0     0.0     30.0
Swati   107.0   70.0    77.0    30.0    29.0
Tarun   0.0     0.0     0.0     0.0     0.0
Uday Mishra     0.0     0.0     0.0     0.0     0.0
Vasanth P       0.0     0.0     0.0     0.0     0.0
Vivek   0.0     15.0    43.0    0.0     0.0
Wasim   0.0     8.0     30.0    46.0    51.0
Zeeshan         0.0     24.0    123.0   63.0    64.0
Time taken: 58.531 seconds, Fetched: 70 row(s)
```

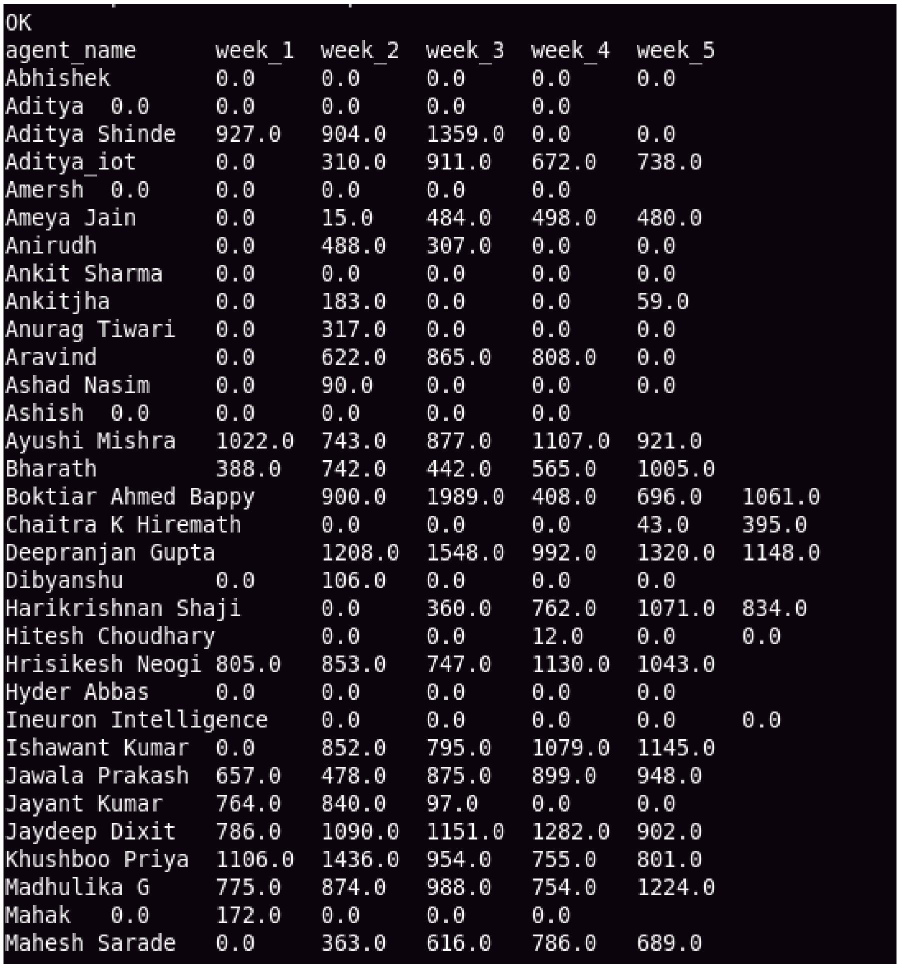
13.) --------

# first calclate the sum of daily resolution time for each agent. Then find the weekly average

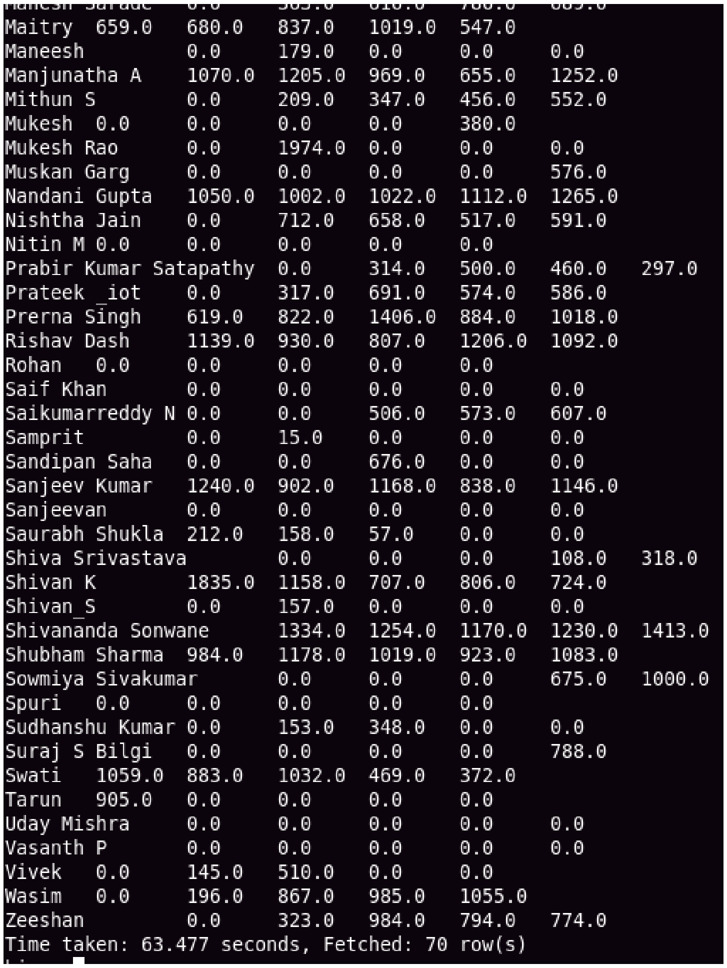
```
with agent_daily_resol_time_table as
(
select from_unixtime(unix_timestamp(date , 'MM/dd/yyyy'), 'yyyy-MM-dd') as
date, agent_name,
sum(avg_resol_time[0] * 3600 + avg_resol_time[1] * 60 + avg_resol_time[2])
as total_daily_resol_time
from agent_performance
group by date,agent_name
),
weekly_avg_table as
(
select weekofyear(date) as
week_of_year,agent_name,round(avg(total_daily_resol_time)) as
weekly_avg_resol_time_in_sec  from agent_daily_resol_time_table
group by weekofyear(date),agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg_resol_time_in_sec else 0
end) as week_1,
sum(case when week_of_year = 27 then weekly_avg_resol_time_in_sec else 0
end) as week_2,
sum(case when week_of_year = 28 then weekly_avg_resol_time_in_sec else 0
end) as week_3,
sum(case when week_of_year = 29 then weekly_avg_resol_time_in_sec else 0
end) as week_4,
sum(case when week_of_year = 30 then weekly_avg_resol_time_in_sec else 0
end) as week_5
from weekly_avg_table
group by agent_name;
```



```
/* first calclate the sum of daily resolution time for each agent. Then find the weekly average */

with agent_daily_resol_time_table as
(
select from_unixtime(unix_timestamp(date , 'MM/dd/yyyy'), 'yyyy-MM-dd') as date, agent_name,
sum(avg_resol_time[0] * 3600 + avg_resol_time[1] * 60 + avg_resol_time[2]) as total_daily_resol_time
from agent_performance
group by date,agent_name
),
weekly_avg_table as
(
select weekofyear(date) as week_of_year,agent_name,round(avg(total_daily_resol_time)) as weekly_avg_resol_time_in_sec
from agent_daily_resol_time_table
group by weekofyear(date),agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg_resol_time_in_sec else 0 end) as week_1,
sum(case when week_of_year = 27 then weekly_avg_resol_time_in_sec else 0 end) as week_2,
sum(case when week_of_year = 28 then weekly_avg_resol_time_in_sec else 0 end) as week_3,
sum(case when week_of_year = 29 then weekly_avg_resol_time_in_sec else 0 end) as week_4,
sum(case when week_of_year = 30 then weekly_avg_resol_time_in_sec else 0 end) as week_5
from weekly_avg_table
group by agent_name;
```



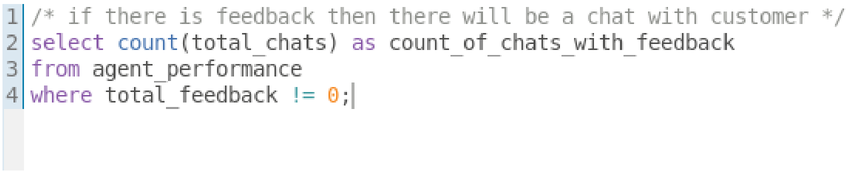
```
OK
agent_name      week_1  week_2  week_3  week_4  week_5
Abhishek        0.0     0.0     0.0     0.0     0.0
Aditya  0.0     0.0     0.0     0.0     0.0
Aditya Shinde   927.0   904.0   1359.0  0.0     0.0
Aditya_iot      0.0     310.0   911.0   672.0   738.0
Amersh  0.0     0.0     0.0     0.0     0.0
Ameya Jain      0.0     15.0    484.0   498.0   480.0
Anirudh         0.0     488.0   307.0   0.0     0.0
Ankit Sharma    0.0     0.0     0.0     0.0     0.0
Ankitjha        0.0     183.0   0.0     0.0     59.0
Anurag Tiwari   0.0     317.0   0.0     0.0     0.0
Aravind         0.0     622.0   865.0   808.0   0.0
Ashad Nasim     0.0     90.0    0.0     0.0     0.0
Ashish  0.0     0.0     0.0     0.0     0.0
Ayushi Mishra   1022.0  743.0   877.0   1107.0  921.0
Bharath         388.0   742.0   442.0   565.0   1005.0
Boktiar Ahmed Bappy     900.0   1989.0  408.0   696.0   1061.0
Chaitra K Hiremath      0.0     0.0     0.0     43.0    395.0
Deepranjan Gupta        1208.0  1548.0  992.0   1320.0  1148.0
Dibyanshu       0.0     106.0   0.0     0.0     0.0
Harikrishnan Shaji      0.0     360.0   762.0   1071.0  834.0
Hitesh Choudhary        0.0     0.0     12.0    0.0     0.0
Hrisikesh Neogi 805.0   853.0   747.0   1130.0  1043.0
Hyder Abbas     0.0     0.0     0.0     0.0     0.0
Ineuron Intelligence    0.0     0.0     0.0     0.0     0.0
Ishawant Kumar  0.0     852.0   795.0   1079.0  1145.0
Jawala Prakash  657.0   478.0   875.0   899.0   948.0
Jayant Kumar    764.0   840.0   97.0    0.0     0.0
Jaydeep Dixit   786.0   1090.0  1151.0  1282.0  902.0
Khushboo Priya  1106.0  1436.0  954.0   755.0   801.0
Madhulika G     775.0   874.0   988.0   754.0   1224.0
Mahak   0.0     172.0   0.0     0.0     0.0
Mahesh Sarade   0.0     363.0   616.0   786.0   689.0
```



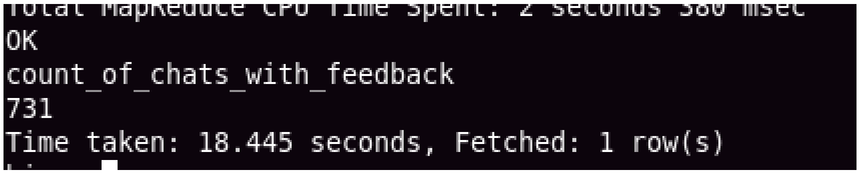
```
Mahesh Sarade   0.0     303.0   616.0   788.0   889.0
Maitry  659.0   680.0   837.0   1019.0  547.0
Maneesh         0.0     179.0   0.0     0.0     0.0
Manjunatha A    1070.0  1205.0  969.0   655.0   1252.0
Mithun S        0.0     209.0   347.0   456.0   552.0
Mukesh  0.0     0.0     0.0     0.0     380.0
Mukesh Rao      0.0     1974.0  0.0     0.0     0.0
Muskan Garg     0.0     0.0     0.0     0.0     576.0
Nandani Gupta   1050.0  1002.0  1022.0  1112.0  1265.0
Nishtha Jain    0.0     712.0   658.0   517.0   591.0
Nitin M 0.0     0.0     0.0     0.0     0.0
Prabir Kumar Satapathy  0.0     314.0   500.0   460.0   297.0
Prateek _iot    0.0     317.0   691.0   574.0   586.0
Prerna Singh    619.0   822.0   1406.0  884.0   1018.0
Rishav Dash     1139.0  930.0   807.0   1206.0  1092.0
Rohan   0.0     0.0     0.0     0.0     0.0
Saif Khan       0.0     0.0     0.0     0.0     0.0
Saikumarreddy N 0.0     0.0     506.0   573.0   607.0
Samprit         0.0     15.0    0.0     0.0     0.0
Sandipan Saha   0.0     0.0     676.0   0.0     0.0
Sanjeev Kumar   1240.0  902.0   1168.0  838.0   1146.0
Sanjeevan       0.0     0.0     0.0     0.0     0.0
Saurabh Shukla  212.0   158.0   57.0    0.0     0.0
Shiva Srivastava        0.0     0.0     0.0     108.0   318.0
Shivan K        1835.0  1158.0  707.0   806.0   724.0
Shivan_S        0.0     157.0   0.0     0.0     0.0
Shivananda Sonwane      1334.0  1254.0  1170.0  1230.0  1413.0
Shubham Sharma  984.0   1178.0  1019.0  923.0   1083.0
Sowmiya Sivakumar       0.0     0.0     0.0     675.0   1000.0
Spuri   0.0     0.0     0.0     0.0     0.0
Sudhanshu Kumar 0.0     153.0   348.0   0.0     0.0
Suraj S Bilgi   0.0     0.0     0.0     0.0     788.0
Swati   1059.0  883.0   1032.0  469.0   372.0
Tarun   905.0   0.0     0.0     0.0     0.0
Uday Mishra     0.0     0.0     0.0     0.0     0.0
Vasanth P       0.0     0.0     0.0     0.0     0.0
Vivek   0.0     145.0   510.0   0.0     0.0
Wasim   0.0     196.0   867.0   985.0   1055.0
Zeeshan         0.0     323.0   984.0   794.0   774.0
Time taken: 63.477 seconds, Fetched: 70 row(s)
```

14.) --------

```sql
select count(total_chats) as count_of_chats_with_feedback
from agent_performance
where total_feedback != 0;
```



```sql
/* if there is feedback then there will be a chat with customer */
select count(total_chats) as count_of_chats_with_feedback
from agent_performance
where total_feedback != 0;
```

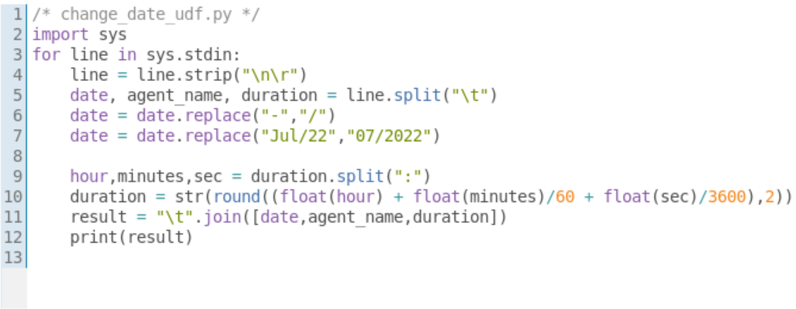


```
Total MapReduce CPU Time Spent: 2 seconds 380 msec
OK
count_of_chats_with_feedback
731
Time taken: 18.445 seconds, Fetched: 1 row(s)
```

15.)---------

change_date_udf.py

```
import sys

for line in sys.stdin:

  line = line.strip("\n\r")
  date, agent_name, duration = line.split("\t")
  date = date.replace("-","/")
  date = date.replace("Jul/22","07/2022")

  hour,minutes,sec = duration.split(":")

  duration = str(round((float(hour) + float(minutes)/60
+float(sec)/3600),2))

  result = "\t".join([date,agent_name,duration])

  print(result)
```



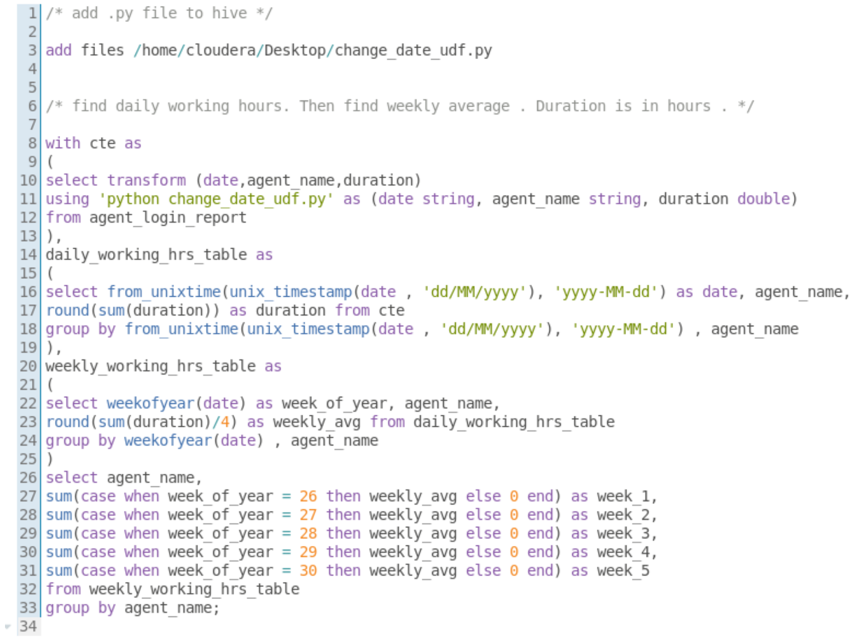
```
/* change_date_udf.py */
import sys
for line in sys.stdin:
    line = line.strip("\n\r")
    date, agent_name, duration = line.split("\t")
    date = date.replace("-","/")
    date = date.replace("Jul/22","07/2022")

    hour,minutes,sec = duration.split(":")
    duration = str(round((float(hour) + float(minutes)/60 + float(sec)/3600),2))
    result = "\t".join([date,agent_name,duration])
    print(result)
```

# add .py file to hive

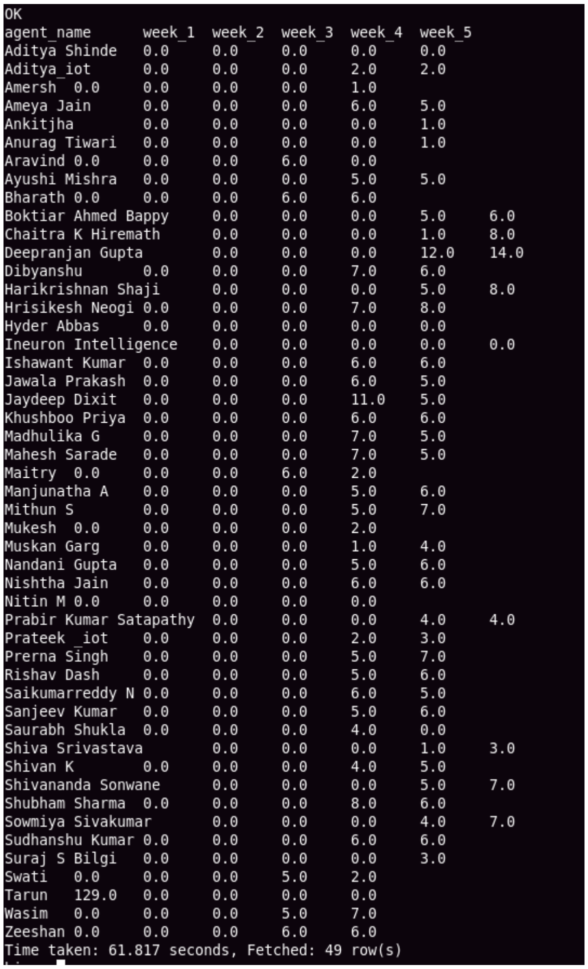
add files /home/cloudera/Desktop/change_date_udf.py

# find daily working hours. Then find weekly average . Duration is in hours .

```
with cte as
(
select transform (date,agent_name,duration)
using 'python change_date_udf.py' as (date string, agent_name string,
duration double)
from agent_login_report
),
daily_working_hrs_table as
(
select from_unixtime(unix_timestamp(date , 'dd/MM/yyyy'), 'yyyy-MM-dd') as
date, agent_name,
round(sum(duration)) as duration from cte
group by from_unixtime(unix_timestamp(date , 'dd/MM/yyyy'), 'yyyy-MM-dd') ,
agent_name
),
weekly_working_hrs_table as
(
select weekofyear(date) as week_of_year, agent_name,
round(sum(duration)/4) as weekly_avg from daily_working_hrs_table
group by weekofyear(date) , agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg else 0 end) as week_1,
sum(case when week_of_year = 27 then weekly_avg else 0 end) as week_2,
sum(case when week_of_year = 28 then weekly_avg else 0 end) as week_3,
sum(case when week_of_year = 29 then weekly_avg else 0 end) as week_4,
sum(case when week_of_year = 30 then weekly_avg else 0 end) as week_5
from weekly_working_hrs_table
group by agent_name;
```



```
/* add .py file to hive */

add files /home/cloudera/Desktop/change_date_udf.py


/* find daily working hours. Then find weekly average . Duration is in hours . */

with cte as
(
select transform (date,agent_name,duration)
using 'python change_date_udf.py' as (date string, agent_name string, duration double)
from agent_login_report
),
daily_working_hrs_table as
(
select from_unixtime(unix_timestamp(date , 'dd/MM/yyyy'), 'yyyy-MM-dd') as date, agent_name,
round(sum(duration)) as duration from cte
group by from_unixtime(unix_timestamp(date , 'dd/MM/yyyy'), 'yyyy-MM-dd') , agent_name
),
weekly_working_hrs_table as
(
select weekofyear(date) as week_of_year, agent_name,
round(sum(duration)/4) as weekly_avg from daily_working_hrs_table
group by weekofyear(date) , agent_name
)
select agent_name,
sum(case when week_of_year = 26 then weekly_avg else 0 end) as week_1,
sum(case when week_of_year = 27 then weekly_avg else 0 end) as week_2,
sum(case when week_of_year = 28 then weekly_avg else 0 end) as week_3,
sum(case when week_of_year = 29 then weekly_avg else 0 end) as week_4,
sum(case when week_of_year = 30 then weekly_avg else 0 end) as week_5
from weekly_working_hrs_table
group by agent_name;

```



```
OK
agent_name      week_1  week_2  week_3  week_4  week_5
Aditya Shinde   0.0     0.0     0.0     0.0     0.0
Aditya_iot      0.0     0.0     0.0     2.0     2.0
Amersh  0.0     0.0     0.0     0.0     1.0
Ameya Jain      0.0     0.0     0.0     6.0     5.0
Ankitjha        0.0     0.0     0.0     0.0     1.0
Anurag Tiwari   0.0     0.0     0.0     0.0     1.0
Aravind 0.0     0.0     0.0     6.0     0.0
Ayushi Mishra   0.0     0.0     0.0     5.0     5.0
Bharath 0.0     0.0     0.0     6.0     6.0
Boktiar Ahmed Bappy     0.0     0.0     0.0     5.0     6.0
Chaitra K Hiremath      0.0     0.0     0.0     1.0     8.0
Deepranjan Gupta        0.0     0.0     0.0     12.0    14.0
Dibyanshu       0.0     0.0     0.0     7.0     6.0
Harikrishnan Shaji      0.0     0.0     0.0     5.0     8.0
Hrisikesh Neogi 0.0     0.0     0.0     7.0     8.0
Hyder Abbas     0.0     0.0     0.0     0.0     0.0
Ineuron Intelligence    0.0     0.0     0.0     0.0     0.0
Ishawant Kumar  0.0     0.0     0.0     6.0     6.0
Jawala Prakash  0.0     0.0     0.0     6.0     5.0
Jaydeep Dixit   0.0     0.0     0.0     11.0    5.0
Khushboo Priya  0.0     0.0     0.0     6.0     6.0
Madhulika G     0.0     0.0     0.0     7.0     5.0
Mahesh Sarade   0.0     0.0     0.0     7.0     5.0
Maitry  0.0     0.0     0.0     6.0     2.0
Manjunatha A    0.0     0.0     0.0     5.0     6.0
Mithun S        0.0     0.0     0.0     5.0     7.0
Mukesh  0.0     0.0     0.0     0.0     2.0
Muskan Garg     0.0     0.0     0.0     1.0     4.0
Nandani Gupta   0.0     0.0     0.0     5.0     6.0
Nishtha Jain    0.0     0.0     0.0     6.0     6.0
Nitin M 0.0     0.0     0.0     0.0     0.0
Prabir Kumar Satapathy  0.0     0.0     0.0     4.0     4.0
Prateek _iot    0.0     0.0     0.0     2.0     3.0
Prerna Singh    0.0     0.0     0.0     5.0     7.0
Rishav Dash     0.0     0.0     0.0     5.0     6.0
Saikumarreddy N 0.0     0.0     0.0     6.0     5.0
Sanjeev Kumar   0.0     0.0     0.0     5.0     6.0
Saurabh Shukla  0.0     0.0     0.0     4.0     0.0
Shiva Srivastava        0.0     0.0     0.0     1.0     3.0
Shivan K        0.0     0.0     0.0     4.0     5.0
Shivananda Sonwane      0.0     0.0     0.0     5.0     7.0
Shubham Sharma  0.0     0.0     0.0     8.0     6.0
Sowmiya Sivakumar       0.0     0.0     0.0     4.0     7.0
Sudhanshu Kumar 0.0     0.0     0.0     6.0     6.0
Suraj S Bilgi   0.0     0.0     0.0     0.0     3.0
Swati   0.0     0.0     0.0     5.0     2.0
Tarun   129.0   0.0     0.0     0.0     0.0
Wasim   0.0     0.0     0.0     5.0     7.0
Zeeshan 0.0     0.0     0.0     6.0     6.0
Time taken: 61.817 seconds, Fetched: 49 row(s)
```

16) -------

a) INNER JOIN

****

# create a table to store the left joined data

```
create  table inner_join_data
row format delimited
fields terminated by ','
lines terminated by '\n'
stored as textfile
as
select
'a.sl_no' as a_sl_no,
'a.date' as a_date,
'a.agent_name' as a_agent_name,
'a.total_chats' as a_total_chats,
'a.avg_resp_time' as a_avg_resp_time,
'a.avg_resol_time' as a_avg_resol_time,
'a.avg_rating' as a_avg_rating,
'a.total_feedback' as a_total_feedback,
'b.sl_no' as b_sl_no,
'b.agent_name' as b_agent_name,
'b.date' as b_date,
'b.login_time' as b_login_time,
'b.logout_time' as b_logout_time,
'b.duration' as b_duration;
```

#insert into that table after join operation

```
insert into inner_join_data
select * from agent_performance a
inner join agent_login_report b
on a.agent_name = b.agent_name;
```

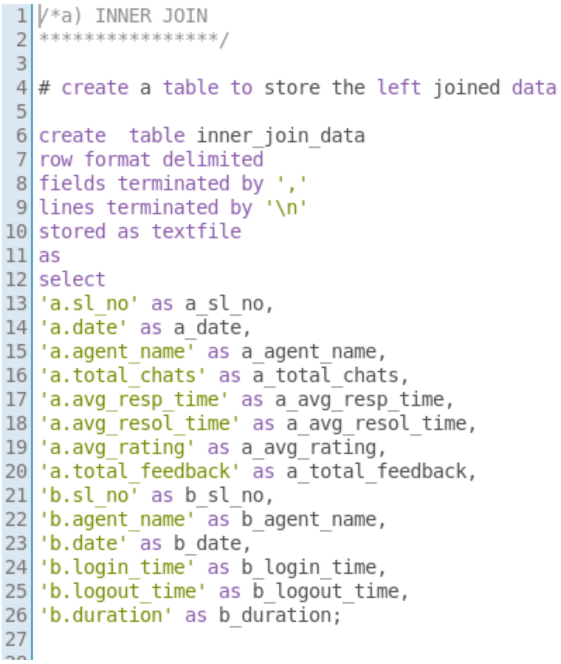
# find the location of table

```
describe formatted inner_join_data;
```

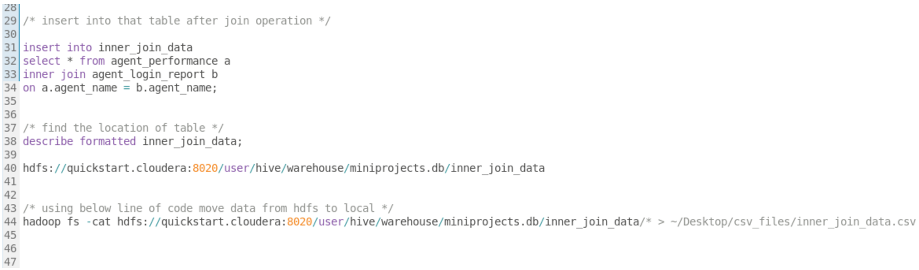
hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/inner_join_data

# using below line of code move data from hdfs to local

```
hadoop fs -cat
hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/inner_join_data/* >
~/Desktop/csv_files/inner_join_data.csv
```



```
/*a) INNER JOIN
****************/

# create a table to store the left joined data

create  table inner_join_data
row format delimited
fields terminated by ','
lines terminated by '\n'
stored as textfile
as
select
'a.sl_no' as a_sl_no,
'a.date' as a_date,
'a.agent_name' as a_agent_name,
'a.total_chats' as a_total_chats,
'a.avg_resp_time' as a_avg_resp_time,
'a.avg_resol_time' as a_avg_resol_time,
'a.avg_rating' as a_avg_rating,
'a.total_feedback' as a_total_feedback,
'b.sl_no' as b_sl_no,
'b.agent_name' as b_agent_name,
'b.date' as b_date,
'b.login_time' as b_login_time,
'b.logout_time' as b_logout_time,
'b.duration' as b_duration;
```



```
/* insert into that table after join operation */

insert into inner_join_data
select * from agent_performance a
inner join agent_login_report b
on a.agent_name = b.agent_name;


/* find the location of table */
describe formatted inner_join_data;

hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/inner_join_data


/* using below line of code move data from hdfs to local */
hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/inner_join_data/* > ~/Desktop/csv_files/inner_join_data.csv
```

LEFT JOIN

****

# we have already a temporary table to store the joined data. Just rename it for our use.

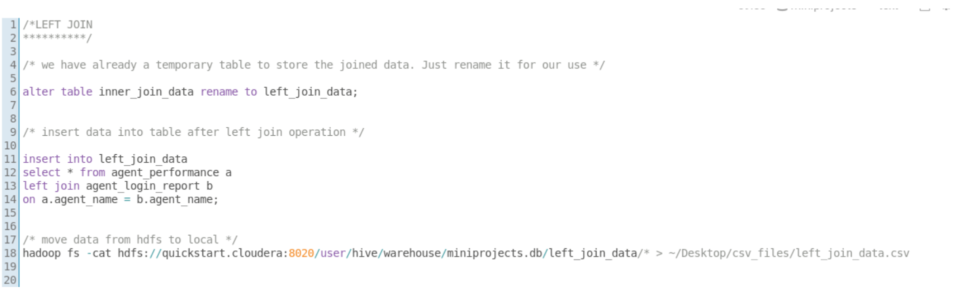
```
alter table inner_join_data rename to left_join_data;
```

# insert data into table after left join operation.

```
insert into left_join_data
select * from agent_performance a
left join agent_login_report b
on a.agent_name = b.agent_name;
```

# move data from hdfs to local

hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/left_join_data/* > ~/Desktop/csv_files/left_join_data.csv



```
/*LEFT JOIN
**********/

/* we have already a temporary table to store the joined data. Just rename it for our use */

alter table inner_join_data rename to left_join_data;


/* insert data into table after left join operation */

insert into left_join_data
select * from agent_performance a
left join agent_login_report b
on a.agent_name = b.agent_name;


/* move data from hdfs to local */
hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/left_join_data/* > ~/Desktop/csv_files/left_join_data.csv
```

RIGHT JOIN

****

we have already a temporary table to store the joined data. Just rename it for our use.

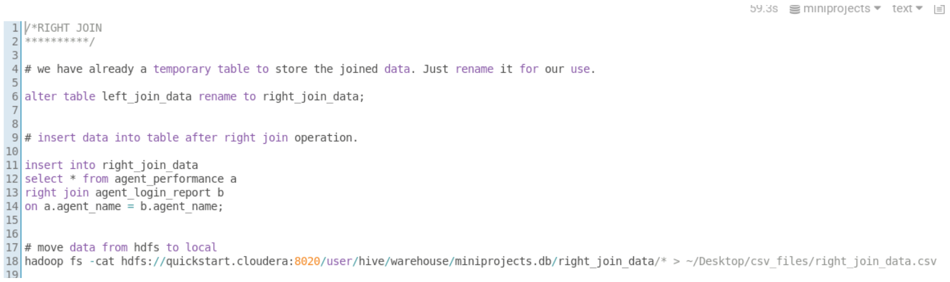
```
alter table left_join_data rename to right_join_data;
```

insert data into table after right join operation.

```
insert into right_join_data
select * from agent_performance a
right join agent_login_report b
on a.agent_name = b.agent_name;
```

move data from hdfs to local

hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/right_join_data/* > ~/Desktop/csv_files/right_join_data.csv



```
/*RIGHT JOIN
**********/

# we have already a temporary table to store the joined data. Just rename it for our use.

alter table left_join_data rename to right_join_data;


# insert data into table after right join operation.

insert into right_join_data
select * from agent_performance a
right join agent_login_report b
on a.agent_name = b.agent_name;


# move data from hdfs to local
hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/right_join_data/* > ~/Desktop/csv_files/right_join_data.csv
```

FULL JOIN

****

# we have already a temporary table to store the joined data. Just rename it for our use.

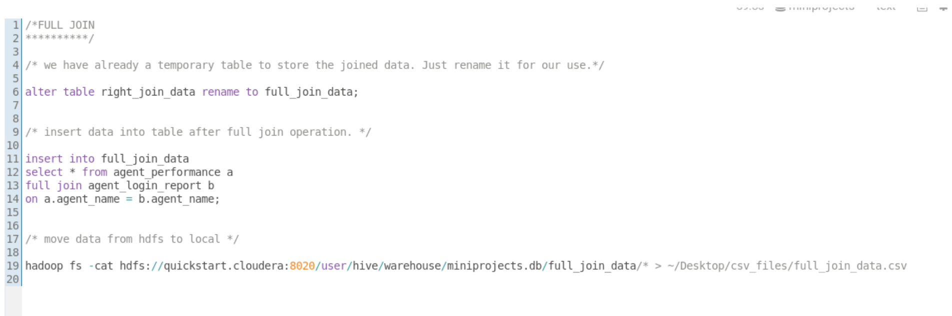
```
alter table right_join_data rename to full_join_data;
```

# insert data into table after full join operation.

```
insert into full_join_data
select * from agent_performance a
full join agent_login_report b
on a.agent_name = b.agent_name;
```

# move data from hdfs to local

hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/full_join_data/* > ~/Desktop/csv_files/full_join_data.csv



```
/*FULL JOIN
**********/

/* we have already a temporary table to store the joined data. Just rename it for our use.*/

alter table right_join_data rename to full_join_data;


/* insert data into table after full join operation. */

insert into full_join_data
select * from agent_performance a
full join agent_login_report b
on a.agent_name = b.agent_name;


/* move data from hdfs to local */

hadoop fs -cat hdfs://quickstart.cloudera:8020/user/hive/warehouse/miniprojects.db/full_join_data/* > ~/Desktop/csv_files/full_join_data.csv

```

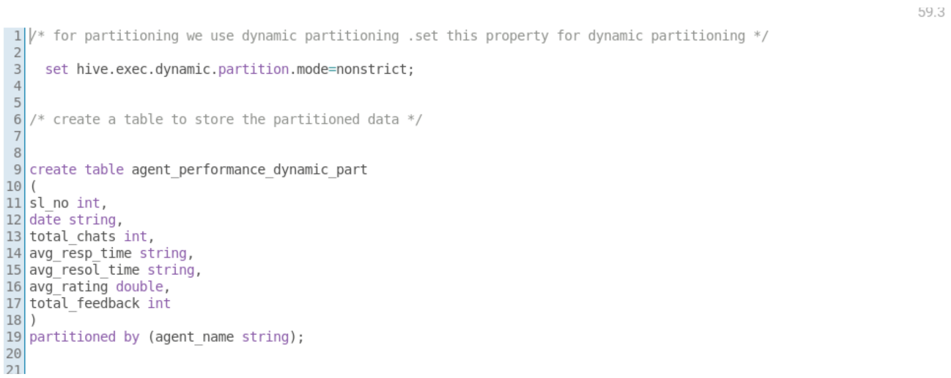
17) --------

# for partitioning we use dynamic partitioning .set this property for dynamic partitioning

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

# create a table to store the partitioned data

```
create table agent_performance_dynamic_part
(
sl_no int,
date string,
total_chats int,
avg_resp_time string,
avg_resol_time string,
avg_rating double,
total_feedback int
)
partitioned by (agent_name string);
```



```
/* for partitioning we use dynamic partitioning .set this property for dynamic partitioning */

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


/* create a table to store the partitioned data */


create table agent_performance_dynamic_part
(
sl_no int,
date string,
total_chats int,
avg_resp_time string,
avg_resol_time string,
avg_rating double,
total_feedback int
)
partitioned by (agent_name string);
```

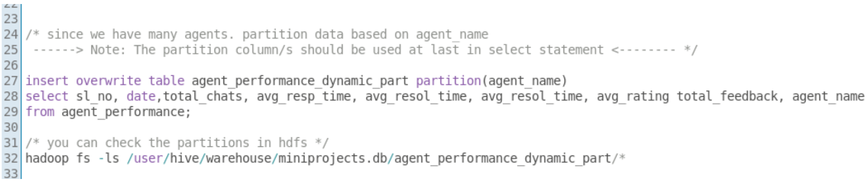
# since we have many agents. partition data based on agent_name

# ------> Note: The partition column/s should be used at last in select statement <--------

```
insert overwrite table agent_performance_dynamic_part
partition(agent_name)
select sl_no, date,total_chats, avg_resp_time, avg_resol_time,
avg_resol_time, avg_rating total_feedback, agent_name
from agent_performance;
```

# you can check the partitions in hdfs

hadoop fs -ls /user/hive/warehouse/miniprojects.db/agent_performance_dynamic_part/*



```

/* since we have many agents. partition data based on agent_name
 ------> Note: The partition column/s should be used at last in select statement <-------- */

insert overwrite table agent_performance_dynamic_part partition(agent_name)
select sl_no, date,total_chats, avg_resp_time, avg_resol_time, avg_resol_time, avg_rating total_feedback, agent_name
from agent_performance;

/* you can check the partitions in hdfs */
hadoop fs -ls /user/hive/warehouse/miniprojects.db/agent_performance_dynamic_part/*

```

Bucketing

****

# first set bucketing to true

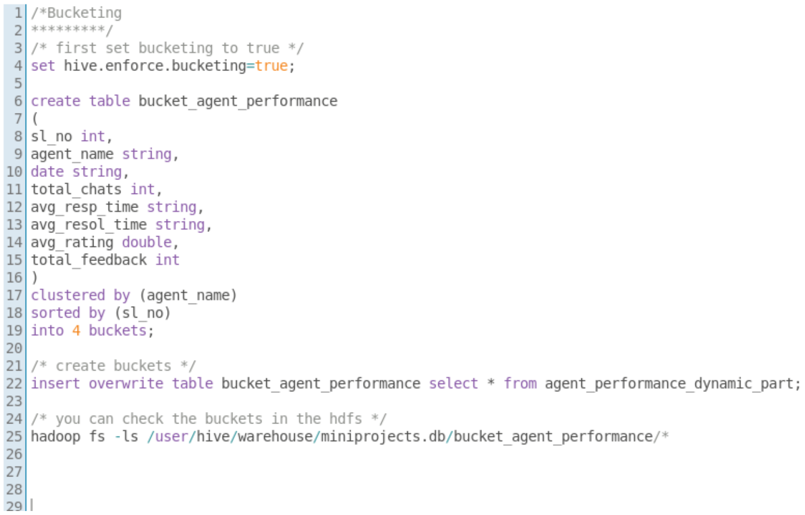
```
set hive.enforce.bucketing=true;
```

```
create table bucket_agent_performance
(
sl_no int,
agent_name string,
date string,
total_chats int,
avg_resp_time string,
avg_resol_time string,
avg_rating double,
total_feedback int
)
clustered by (agent_name)
sorted by (sl_no)
into 4 buckets;
```

# create buckets

```
insert overwrite table bucket_agent_performance select * from
agent_performance_dynamic_part;
```

```
# you can check the buckets in the hdfs

hadoop fs -ls /user/hive/warehouse/miniprojects.db/bucket_agent_performance/*
```



```
/*Bucketing
*********/
/* first set bucketing to true */
set hive.enforce.bucketing=true;

create table bucket_agent_performance
(
sl_no int,
agent_name string,
date string,
total_chats int,
avg_resp_time string,
avg_resol_time string,
avg_rating double,
total_feedback int
)
clustered by (agent_name)
sorted by (sl_no)
into 4 buckets;

/* create buckets */
insert overwrite table bucket_agent_performance select * from agent_performance_dynamic_part;

/* you can check the buckets in the hdfs */
hadoop fs -ls /user/hive/warehouse/miniprojects.db/bucket_agent_performance/*
```