My SQL Query

Coffee Shop Sales Project

1. convert data column to proper format

UPDATE coffe\_shop

SET transaction\_date = STR\_TO\_DATE(transaction\_date, '%m/%d/%Y');

alter table coffe\_shop

modify column transaction\_date date ;

1. Convert time column into proper format

update coffe\_shop

set transaction\_time = str\_to\_date(transaction\_time, '%H:%i:%s') ;

alter table coffe\_shop

modify transaction\_time time ;

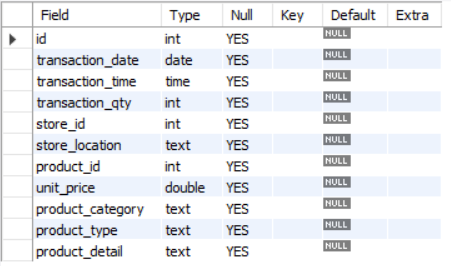
1. Rename the column transaction\_id to id

alter table coffe\_shop

change transaction\_id id int ;

1. Data type of all column

describe coffe\_shop ;



KPI's Requirements

1. Total sales

select concat(round(sum(unit\_price \* transaction\_qty))/1000 , 'K') as total\_sales

from coffe

where

month(transaction\_date) = 5 ;



1. Calculate Groth by month

select

month(transaction\_date) as month ,

round(sum(unit\_price \* transaction\_qty)) as Total\_sales ,

round((sum(unit\_price \* transaction\_qty) - lag(sum(unit\_price \* transaction\_qty) , 1)

over(order by month(transaction\_date)))

/

lag(sum(unit\_price \*transaction\_qty ) , 1) over(order by month(transaction\_date)) \* 100,2 )as mom\_percentage

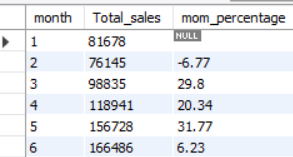
from coffe

group by

month(transaction\_date)

order by

month(transaction\_date) ;



1. Total orders

select count(distinct id) as total\_orders

from coffe

where month(transaction\_date) = 5 ;



MOM regarding orders

select

month(transaction\_date) as Month ,-- frist col for month

round(count(id)) as Total\_orders , -- second\_col for total orders

(count(id) - lag(count(id) , 1) over(order by month(transaction\_date)))

/

lag(count(id) , 1) over(order by month(transaction\_date)) \* 100 as MOM\_percentag

from

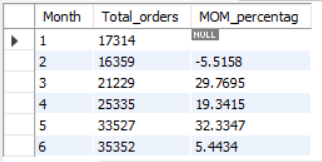
coffe

group by

month(transaction\_date)

order by

month(transaction\_date) ;



1. Total quantity sold

select sum(transaction\_qty) as total\_quantit\_sold

from coffe

where month(transaction\_date) = 5;



1. Calculate groth for quantity sold

select

month(transaction\_date) as Month ,

sum(transaction\_qty) as total\_quantit\_sold ,

round( (sum(transaction\_qty) - lag(sum(transaction\_qty) , 1) over(order by month(transaction\_date)))

/

Lag(sum(transaction\_qty) , 1) over(order by month(transaction\_date)) \*100 ,2)as MOM\_quantit\_sold

from

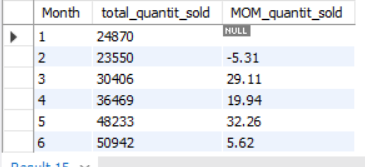
coffe

group by

Month

order by

Month ;



1. Calculate total sales in weekdays and weekend

select

case when dayofweek(transaction\_date) in (1,7) then 'Weekend'

else 'Weekdays'

end as day\_type ,

concat(round(sum(unit\_price \*transaction\_qty) /1000 , 1),'K') as total\_sales

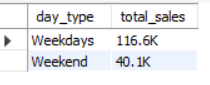
from

coffe

where month(transaction\_date) = 5

group by

day\_type ;



1. Total\_sales by store location

select

store\_location ,

concat(round(sum(unit\_price \*transaction\_qty) /1000,1) , 'K') as total\_sales

from

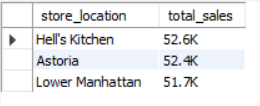
coffe

where

month(transaction\_date) = 5

group by store\_location

order by total\_sales desc ;



1. average for may

select

round(avg(total\_sales)) as Avg\_sales

from

(

select sum(unit\_price \*transaction\_qty) as total\_sales

from coffe

where month(transaction\_date) = 5

group by transaction\_date

) as intern\_query

;



1. which day is greater than the avg of the specefied month

select day\_of\_month ,

case

when total\_sales > avg\_sales then "Above Avg"

when total\_sales < avg\_sales then "Below Avg"

else 'equal to avg'

end as Sales\_status ,

total\_sales , avg\_sales

from (

select

day(transaction\_date) as day\_of\_month , -- for day number

round(sum(unit\_price \*transaction\_qty)) as total\_sales ,

round(avg(sum(unit\_price \*transaction\_qty))over(),1) as avg\_sales

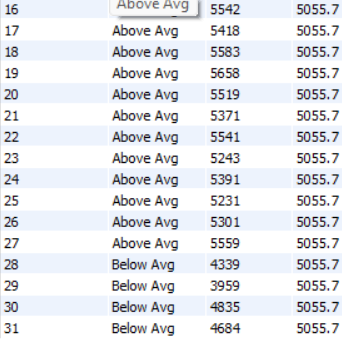
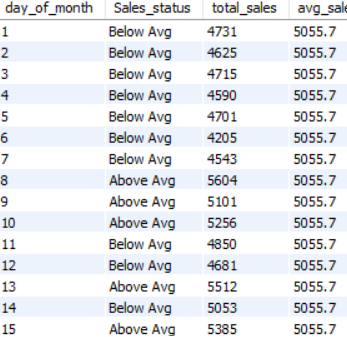
from coffe

where

month(transaction\_date) = 5

group by

day\_of\_month ) as internal\_table ;



1. sales with respect to product\_category

select product\_category ,

round(sum(unit\_price \*transaction\_qty)) as total\_sales

from

coffe

where

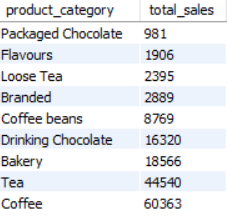
month(transaction\_date) = 5

group by

product\_category

order by

total\_sales ;



-- sales with respect to product\_type top 10

select product\_type ,

round(sum(unit\_price \*transaction\_qty)) as total\_sales

from coffe

where

month(transaction\_date) = 5

group by

product\_type

order by total\_sales desc

limit 10 ;



1. Total sales total\_quanitiy and aorders with respect to may monday hour number 8

select

round(sum(unit\_price \*transaction\_qty) , 1) as total\_sales ,

sum(transaction\_qty) as total\_quantity ,

count(\*) as total\_orders

from coffe

where month(transaction\_date) = 5 -- may

and dayofweek(transaction\_date) = 2 -- monday

and hour(transaction\_time) = 8;



1. total sales with respect to hours

select

hour(transaction\_time) as Hour ,

round(sum(unit\_price \*transaction\_qty) , 1) as total\_sales ,

sum(transaction\_qty) as total\_quantity ,

count(\*) as total\_orders

from coffe

where

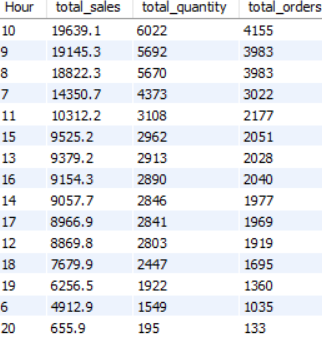
month(transaction\_date) = 5 -- may

group by

Hour

order by

total\_sales desc ;



1. Total sales in each day in may

select

case

when dayofweek(transaction\_date) = 2 then 'Monday'

when dayofweek(transaction\_date) = 3 then 'Tuesday'

when dayofweek(transaction\_date) = 4 then 'Wednesday'

when dayofweek(transaction\_date) = 5 then 'Thursday'

when dayofweek(transaction\_date) = 6 then 'Friday'

when dayofweek(transaction\_date) = 7 then 'Saturday'

else 'Sunday'

end as Day\_of\_week ,

round(sum(unit\_price \*transaction\_qty)) as total\_sales

from

coffe

where

month(transaction\_date) = 5

group by

Day\_of\_week

order by

total\_sales desc ;

