CoffeeShop SQL Query Summary

create database Coffeeshop\_db;

use coffeeshop\_db;

select \* from coffeeshop;

update coffeeshop

set transaction\_date = str\_to\_date(transaction\_date,'%d-%m-%Y');

alter table coffeeshop

modify column transaction\_date date;

desc coffeeshop;

select \* from coffeeshop;

update coffeeshop

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

alter table coffeeshop

modify column transaction\_time Time;

desc coffeeshop;

-- Total Sales Analysis--

-- Total sales for each month--

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_January

from coffeeshop

where Month(transaction\_date) = 1; -- for January month

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_February

from coffeeshop

where Month(transaction\_date) = 2; -- for month February

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_March

from coffeeshop

where Month(transaction\_date) = 3; -- for March Month

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_April

from coffeeshop

where Month(transaction\_date) = 4; -- for April month

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_May

from coffeeshop

where Month(transaction\_date) = 5 ; -- for May month

Select

Round(Sum(transaction\_qty \* unit\_price),1)as Total\_Sales\_June

from coffeeshop

where Month(transaction\_date) = 6; -- for June month

-- Month-on-Month Decrease or Increase in sales—

-- Previous month is May and Current Month is June--

select

month(transaction\_date) as months,

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

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

over(order by month(transaction\_date))) / lag(sum(transaction\_qty \* unit\_price), 1)

over(order by month(transaction\_date)) \* 100 as Month\_on\_Month\_sales

from coffeeshop

where month(transaction\_date) in (4,5)

group by month(transaction\_date)

order by month(transaction\_date);

-- Total Order Analysis—

select

count(transaction\_id) as Total\_Orders\_January

from coffeeshop

where

month(transaction\_date)= 1; -- for January month

select

count(transaction\_id) as Total\_Orders\_February

from coffeeshop

where

month(transaction\_date)= 2 ; -- for February month

select

count(transaction\_id) as Total\_Orders\_March

from coffeeshop

where

month(transaction\_date)= 3; -- for March month

select

count(transaction\_id) as Total\_Orders\_April

from coffeeshop

where

month(transaction\_date)= 4; -- for April month

select

count(transaction\_id) as Total\_Orders\_May

from coffeeshop

where

month(transaction\_date)= 5; -- for May month

select

count(transaction\_id) as Total\_Orders\_June

from coffeeshop

where

month(transaction\_date)= 4; -- for June month

-- Month-on-Month Total Orders--

select

month(transaction\_date) as Months, -- Month Number

count(transaction\_id) as Total\_Orders, -- Total Orders

concat((count(transaction\_id)- lag(count(transaction\_id),1) -- Order Difference

over (order by month(transaction\_date))) / lag(count(transaction\_id),1)

over(order by month(transaction\_date)) \* 100,'%') as Month\_On\_Month\_Order\_change\_Pct

From

coffeeshop

where

month(transaction\_date) in (5,6)

group by month(transaction\_date)

order by month(transaction\_date);

-- Total Quantity Sold Analysis--

-- Total Quantity Sold for Each Month --

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_January

from coffeeshop

where month(transaction\_date) = 1;

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_February

from coffeeshop

where month(transaction\_date) = 2;

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_March

from coffeeshop

where month(transaction\_date) = 3;

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_April

from coffeeshop

where month(transaction\_date) = 4;

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_May

from coffeeshop

where month(transaction\_date) = 5;

Select

sum(transaction\_qty) as Total\_Quantity\_Sold\_June

from coffeeshop

where month(transaction\_date) = 6;

-- Percentage change in Total Quantity sold Month-on-Month--

-- Previous month is May and Current Month is June--

select

sum(transaction\_qty) as Total\_Qty\_Sold,

concat((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 , '%')

as Total\_Qty\_Sold\_Pct

from coffeeshop

where month(transaction\_date) in (5,6)

group by month(transaction\_date);

-- Total Sales, Total Quantity Sold, Total Orders on any particular date--

select

concat(round(count(transaction\_id)/1000,1),'K') as Total\_Orders,

concat(round(sum(transaction\_qty)/1000,1),'K') as Total\_Quantity\_sold,

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

from coffeeshop

where transaction\_date= '2023-05-18';

-- Total Sales on Weekdays and Weekends for the month of May--

select

case when

dayofweek(transaction\_date) in (1,7) then 'Weekends'

else 'Weekdays'

End as Days,

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

from coffeeshop

where Month(transaction\_date)= 5

Group by Days;

-- Sales Analysis by Location for each month--

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=1 -- for Janurary

group by Location

Order by Total\_sales\_january desc;

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=2 -- for February

group by Location

Order by Total\_sales\_February desc;

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=3 -- for March

group by Location

Order by Total\_sales\_March desc;

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=4 -- for April

group by Location

Order by Total\_sales\_April desc;

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=5 -- for May

group by Location

Order by Total\_sales\_May desc;

select

store\_location as Location,

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

from coffeeshop

where month(transaction\_date)=6 -- for June

group by Location

Order by Total\_sales\_June desc;

-- Average Sales for Different Months--

select

concat(round(Avg(Total\_Sales)/1000,1),'K') as Avg\_Sales

from

(select sum(transaction\_qty \* unit\_price) as Total\_Sales

from coffeeshop

where Month(transaction\_date)=1

Group by transaction\_date)

as Internal\_Query;

-- Total Sales Day wise of any particular month--

Select

Day(transaction\_date) as Days\_of\_Month,

Round(Sum(transaction\_qty \* unit\_price),2) as Total\_Sales

From

coffeeshop

Where Month(transaction\_date)=1 -- For January Month--

Group by

transaction\_date

Order by

transaction\_date;

-- Compare the Total Sales of each day is Below Average , Above Average or Average for any particular month--

-- For January--

Select

Day\_of\_month,

Total\_Sales,

Avg\_Sales,

case

when Total\_sales > Avg\_sales then 'Above Average'

when Total\_sales < Avg\_sales then 'Below Average'

Else 'Average'

End as Sales\_status

from

(Select

Day(transaction\_date) as Day\_of\_month,

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

Avg(sum(transaction\_qty \* unit\_price)) over() as Avg\_sales

from coffeeshop

where month(transaction\_date)=1

group by transaction\_date

order by transaction\_date

) as Avg\_sales\_Query ;

-- Total Sales by Product categories for each month--

Select

Product\_category as Product\_Categories,

concat(round(sum(transaction\_qty \* unit\_price)/1000,2), 'K') as Total\_sales

from

coffeeshop

where month(transaction\_date)= 1 -- for January--

group by

Product\_categories

order by

Total\_sales desc;

-- Total Sales by Product categories for each month--

Select

Product\_category as Product\_Categories,

concat(round(sum(transaction\_qty \* unit\_price)/1000,2), 'K') as Total\_sales

from

coffeeshop

where month(transaction\_date)= 5 -- for January--

group by Product\_categories

order by Total\_sales desc ;

-- Top Ten Product by sales volume--

Select

product\_type as Products,

sum(transaction\_qty \* unit\_price) as Total\_Sales

from

coffeeshop

where month(transaction\_date) = 1 -- For January--

Group by product\_type

Order by Total\_sales desc

Limit 10;

-- Sales Analysis for any Days and Hours--

Select

count(\*) as Total\_Orders,

sum(transaction\_qty) as Total\_Quantity\_Sold,

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

from coffeeshop

where month(transaction\_date) = 1 and -- for January--

dayofweek(transaction\_date) = 1 and -- for Sunday--

hour(transaction\_time) = 8; -- for 8 hour—

-- Hourly Sales of any Particular Month--

Select

Hour(transaction\_time) as Hours,

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

from coffeeshop

where month(transaction\_date) = 1 -- for January--

Group by Hours

Order by Hours;

-- Total Sales according to days of week of any particular month--

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 'Friday'

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

else 'Sunday'

End as Days\_of\_week,

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

from coffeeshop

where month(transaction\_date)= 1 -- for Janruary--

group by Days\_of\_week;