**Coffee Shop Sales Data Analysis**

**Data Correction**

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;

**-- 1 TOTAL SALES:**

select round(sum(transaction\_qty \* unit\_price),2)as Total\_Sales from coffee\_shop\_sales;

select round(sum(transaction\_qty \* unit\_price),2)as Total\_Sales from coffee\_shop\_sales

where month(transaction\_date) = 5; -- MTD

**-- 2.TOTAL SALES KPI - MOM DIFFERENCE AND MOM GROWTH:**

select month(transaction\_date) as No\_of\_Month,

round(sum(unit\_price \* transaction\_qty),2) 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 Percentage\_sales\_MoM

from coffee\_shop\_sales

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

group by MONTH(transaction\_date)

order by month(transaction\_date);

**-- 3. TOTAL ORDERS:**

select count(transaction\_id) as Total\_qty\_ Order from coffee\_shop\_sales;

select count(transaction\_id) as Total\_qty\_Order from coffee\_shop\_sales

where month(transaction\_date) in (4,5,2); -- for 2 month or more months

select count(transaction\_id) as Total\_qty\_ Order from coffee\_shop\_sales

where month(transaction\_date) = 5; -- MTD

**-- 4.TOTAL ORDERS KPI - MOM DIFFERENCE AND MOM GROWTH:**

select month(transaction\_date) as Month,

count(transaction\_id) as Total\_order,

(count(transaction\_id) - lag(count(transaction\_id),1)

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

over(order by month(transaction\_date)) \* 100 as Percentage\_order\_MOM

from coffee\_shop\_sales

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

group by 1

order by 1;

**-- 5.TOTAL QUANTITY SOLD:**

select sum(transaction\_qty) as Total\_qty\_sold from coffee\_shop\_sales;

select sum(transaction\_qty) as Total\_qty\_sold from coffee\_shop\_sales

where month(transaction\_date) in (4,5,2); -- for 2 month or more months

select sum(transaction\_qty) as Total\_qty\_sold from coffee\_shop\_sales

where month(transaction\_date) = 5; -- MTD

**-- 6. TOTAL QUANTITY SOLD KPI - MOM DIFFERENCE AND MOM GROWTH:**

select month(transaction\_date) as Month,

sum(transaction\_qty) as Total\_order,

(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 Percentage\_Sales\_MOM

from coffee\_shop\_sales

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

group by 1

order by 1;

**-- 7.CALENDAR TABLE – DAILY SALES, QUANTITY and TOTAL ORDERS:**

select

month(transaction\_date),

count(transaction\_id) as Total\_Order,

sum(transaction\_qty) as Total\_Quantity,

sum(unit\_price \* transaction\_qty) as Total\_Sales from coffee\_shop\_sales

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

group by month(transaction\_date);

SELECT

SUM(unit\_price \* transaction\_qty) AS total\_sales,

SUM(transaction\_qty) AS total\_quantity\_sold,

COUNT(transaction\_id) AS total\_orders

FROM

coffee\_shop\_sales

WHERE

transaction\_date = '2023-05-18'; -- For 18 May 2023

**-- 8. SALES TREND OVER PERIOD :**

select sum(transaction\_qty \* unit\_price) as Total\_Sales from coffee\_shop\_sales

where dayofmonth(transaction\_date)= 5;

-- avg sales over period:

select avg(transaction\_qty \* unit\_price)from coffee\_shop\_sales;

-- or

select avg(total\_sales) as Average\_sales from

(

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

where month(transaction\_date)= 5

group by transaction\_date) as internal;

**-- 9.DAILY SALES FOR MONTH SELECTED:**

select day(transaction\_date) as Day\_of\_month, sum(transaction\_qty \* unit\_price) as total\_sales from coffee\_shop\_sales

where month(transaction\_date) = 5

group by day(transaction\_date);

**-- 10.SALES BY WEEKDAY / WEEKEND:**

select

case

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

else 'Weekday'

end as Day\_status ,

round(sum(transaction\_qty \*unit\_price ),2) as Total\_sales

from coffee\_shop\_sales

where month(transaction\_date) =5

group by 1;

**-- 11. SALES BY STORE LOCATION:**

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

from coffee\_shop\_sales

where month(transaction\_date)

group by store\_location

order by 2;

**-- 12. SALES BY PRODUCT CATEGORY:**

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

from coffee\_shop\_sales

where month(transaction\_date)

group by product\_category

order by 2;

**-- 13.SALES BY PRODUCTS (TOP 10):**

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

from coffee\_shop\_sales

where month(transaction\_date)

group by product\_category

order by 2 desc limit 10;

**-- 14.SALES BY DAY | HOUR:**

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

from coffee\_shop\_sales

where dayofweek(transaction\_date) = 3

and hour(transaction\_time)=8

and month(transaction\_date)=5;

**-- 15. TO GET SALES FOR ALL HOURS FOR MONTH OF MAY :**

select hour(transaction\_time) as Sales\_by\_hour,

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

where month(transaction\_date)=5

group by 1

order by 1;

**-- Advanced queries**

**-- 1. Sales Performance by Product (Most Popular Items)**

**-- To identify which products (e.g., coffee, pastries) are sold the most:**

select product\_type as Best\_Product ,dayofweek(transaction\_date) as Day\_of\_Week,

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

from coffee\_shop\_sales

group by product\_type, dayofweek(transaction\_date)

order by dayofweek(transaction\_date) desc limit 5;

**-- 2. Total Sales for Each Product:**

select product\_type as Product,

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

from coffee\_shop\_sales

group by product\_type;

**-- 3. Top Products by Sales in a Specific Period (Last 30 Days)**

**-- This query ranks the products based on their total sales over the last 30 days.:**

SELECT

product\_id,

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

FROM

coffee\_shop\_sales

WHERE

transaction\_date >= CURRENT\_DATE - INTERVAL '30 days'

GROUP BY 1 ORDER BY 2 DESC;

**-- 4. Sales by Day and Time (Morning vs. Afternoon) for particular day of week:**

select hour(transaction\_time) as hour\_of\_Day,

case

when hour(transaction\_time)>12 and hour(transaction\_time)< 16 then 'Afternoon Orders'

when hour(transaction\_time)>=16 and hour(transaction\_time)<24 then 'Evening Orders'

else 'Morning Order'

end as Order\_time,

round(sum(transaction\_qty \* unit\_price),2) as Total\_sales

from coffee\_shop\_sales

where dayofweek(transaction\_date)=6

group by 2,1

order by 1;

**-- 5. -- 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;

**-- 6. 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;

**-- 5. COMPARING DAILY SALES WITH AVERAGE SALES – IF GREATER THAN “ABOVE AVERAGE” and LESSER THAN “BELOW AVERAGE”:**

select day\_of\_month,

case

when total\_sale>avg\_sale then 'Above Average'

when total\_sale<avg\_sale then 'Below Average'

else 'Average'

end As Sales\_status,

total\_sale from

(select day(transaction\_date) as day\_of\_month,

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

AVG(SUM(unit\_price \* transaction\_qty)) OVER () AS avg\_sale -- not understand

from coffee\_shop\_sales

where month(transaction\_date) = 5

group by day(transaction\_date)) as sales\_data;