MySQL Queries

**CONVERT DATE (transaction\_date) COLUMN TO PROPER DATE FORMAT**

UPDATE coffee\_shop\_sales

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

**ALTER DATE (transaction\_date) COLUMN TO DATE DATA TYPE**

ALTER TABLE coffee\_shop\_sales

MODIFY COLUMN transaction\_date DATE;

**CONVERT TIME (transaction\_time) COLUMN TO PROPER DATE FORMAT**

UPDATE coffee\_shop\_sales

SET transaction\_time = STR\_TO\_DATE(transaction\_time, '%H:%i:%s');

**ALTER TIME (transaction\_time) COLUMN TO DATE DATA TYPE**

ALTER TABLE coffee\_shop\_sales

MODIFY COLUMN transaction\_time TIME;

**DATA TYPES OF DIFFERENT COLUMNS**

DESCRIBE coffee\_shop\_sales;

**CHANGE COLUMN NAME `ï»¿transaction\_id` to transaction\_id**

ALTER TABLE coffee\_shop\_sales

CHANGE COLUMN `ï»¿transaction\_id` transaction\_id INT;

**TOTAL SALES**

SELECT ROUND(SUM(unit\_price \* transaction\_qty)) as Total\_Sales

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date) = 5 -- for month of (CM-May)

**TOTAL SALES KPI - MOM DIFFERENCE AND MOM GROWTH**

SELECT

MONTH(transaction\_date) AS month,

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

(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 AS mom\_increase\_percentage

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) IN (4, 5) -- for months of April and May

GROUP BY

MONTH(transaction\_date)

ORDER BY

MONTH(transaction\_date);

**TOTAL ORDERS**

SELECT COUNT(transaction\_id) as Total\_Orders

FROM coffee\_shop\_sales

WHERE MONTH (transaction\_date)= 5 -- for month of (CM-May)

**TOTAL ORDERS KPI - MOM DIFFERENCE AND MOM GROWTH**

SELECT

MONTH(transaction\_date) AS month,

ROUND(COUNT(transaction\_id)) AS total\_orders,

(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 mom\_increase\_percentage

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) IN (4, 5) -- for April and May

GROUP BY

MONTH(transaction\_date)

ORDER BY

MONTH(transaction\_date);

**TOTAL QUANTITY SOLD**

SELECT SUM(transaction\_qty) as Total\_Quantity\_Sold

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date) = 5 -- for month of (CM-May)

**TOTAL QUANTITY SOLD KPI - MOM DIFFERENCE AND MOM GROWTH**

SELECT

MONTH(transaction\_date) AS month,

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

(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 mom\_increase\_percentage

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) IN (4, 5) -- for April and May

GROUP BY

MONTH(transaction\_date)

ORDER BY

MONTH(transaction\_date);

**CALENDAR TABLE – DAILY SALES, QUANTITY and TOTAL ORDERS**

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

**SALES TREND OVER PERIOD**

SELECT AVG(total\_sales) AS average\_sales

FROM (

SELECT

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

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) = 5 -- Filter for May

GROUP BY

transaction\_date

) AS internal\_query;

**DAILY SALES FOR MONTH SELECTED**

SELECT

DAY(transaction\_date) AS day\_of\_month,

ROUND(SUM(unit\_price \* transaction\_qty),1) AS total\_sales

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) = 5 -- Filter for May

GROUP BY

DAY(transaction\_date)

ORDER BY

DAY(transaction\_date);

**SALES BY WEEKDAY / WEEKEND:**

SELECT

CASE

WHEN DAYOFWEEK(transaction\_date) IN (1, 7) THEN 'Weekends'

ELSE 'Weekdays'

END AS day\_type,

ROUND(SUM(unit\_price \* transaction\_qty),2) AS total\_sales

FROM

coffee\_shop\_sales

WHERE

MONTH(transaction\_date) = 5 -- Filter for May

GROUP BY

CASE

WHEN DAYOFWEEK(transaction\_date) IN (1, 7) THEN 'Weekends'

ELSE 'Weekdays'

END;

**SALES BY STORE LOCATION**

SELECT

store\_location,

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

FROM coffee\_shop\_sales

WHERE

MONTH(transaction\_date) =5

GROUP BY store\_location

ORDER BY SUM(unit\_price \* transaction\_qty) DESC

**SALES BY PRODUCT CATEGORY**

SELECT

product\_category,

ROUND(SUM(unit\_price \* transaction\_qty),1) as Total\_Sales

FROM coffee\_shop\_sales

WHERE

MONTH(transaction\_date) = 5

GROUP BY product\_category

ORDER BY SUM(unit\_price \* transaction\_qty) DESC

**SALES BY PRODUCTS (TOP 10)**

SELECT

product\_type,

ROUND(SUM(unit\_price \* transaction\_qty),1) as Total\_Sales

FROM coffee\_shop\_sales

WHERE

MONTH(transaction\_date) = 5

GROUP BY product\_type

ORDER BY SUM(unit\_price \* transaction\_qty) DESC

LIMIT 10

**SALES BY DAY | HOUR**

SELECT

ROUND(SUM(unit\_price \* transaction\_qty)) AS Total\_Sales,

SUM(transaction\_qty) AS Total\_Quantity,

COUNT(\*) AS Total\_Orders

FROM

coffee\_shop\_sales

WHERE

DAYOFWEEK(transaction\_date) = 3 -- Filter for Tuesday (1 is Sunday, 2 is Monday, ..., 7 is Saturday)

AND HOUR(transaction\_time) = 8 -- Filter for hour number 8

AND MONTH(transaction\_date) = 5; -- Filter for May (month number 5)