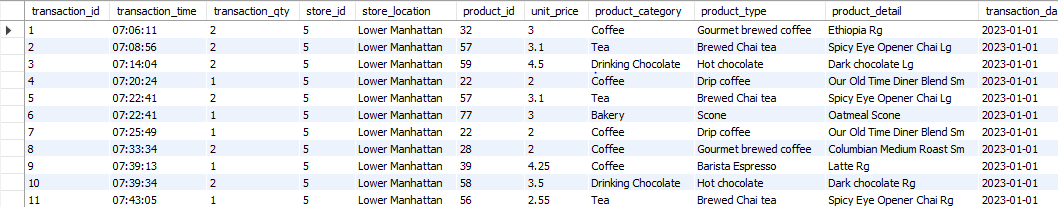
**COFFEE SALES SQL ANALYSIS RESULTS**

1. Viewing the table coffee\_shop\_sales

select \* from coffee\_shop\_sales;



1. Changing data type to date and column

converting transaction\_date into dd/mm/yy

UPDATE coffee\_shop\_sales

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

------------------------------------------------------------------

-- changing data type of transaction\_date from text to date

ALTER TABLE coffee\_shop\_sales

MODIFY COLUMN transaction\_date DATE;

-- The 2 above query threw an error as the date in the table does not match the MYSQL standard date format 'YYYY-MM-DD'.

-- So, I used following steps to solve the problem.

-- Step 1: Adding a temporary column

ALTER TABLE coffee\_shop\_sales

ADD COLUMN temp\_transaction\_date DATE;

-- Step 2: Update the temporary column with converted date values

UPDATE coffee\_shop\_sales

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

-- Step 4: Drop the original column

ALTER TABLE coffee\_shop\_sales

DROP COLUMN transaction\_date;

-- Step 5: Rename the temporary column to the original column name

ALTER TABLE coffee\_shop\_sales

CHANGE COLUMN temp\_transaction\_date transaction\_date DATE;

select \* from coffee\_shop\_sales;

describe coffee\_shop\_sales;

---------------------------------------------------------------------------

-- converting transaction\_time into HH:MM:SS

UPDATE coffee\_shop\_sales

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

--------------------------------------------------------------------

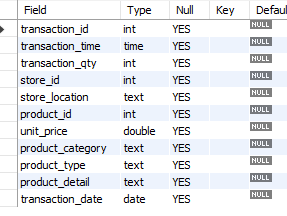
-- changing data type of transaction\_time from text to time

ALTER TABLE coffee\_shop\_sales

MODIFY COLUMN transaction\_time TIME;

-- viewing the data type of columns

describe coffee\_shop\_sales;



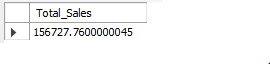
1. Finding total sales for any month

select \* from coffee\_shop\_sales;

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

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date) = 5;



1. Finding monthly count increase in percentage

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)

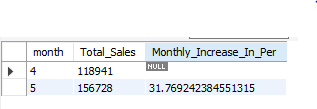
OVER(ORDER BY MONTH(transaction\_date))\*100 as MonthlyIncreaseinOrders

FROM coffee\_shop\_Sales

WHERE MONTH(transaction\_date) IN (4,5)

GROUP BY MONTH(transaction\_date)

ORDER BY MONTH(transaction\_date);



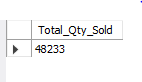
1. Finding total quantity sold in particular month

select \* from coffee\_shop\_sales;

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

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5;



1. Finding total sales, total quantity sold and total orders in a particulat date

SELECT

MONTH(transaction\_date) As month,

ROUND(SUM(transaction\_qty)) AS Total\_Orders,

(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 MonthlyIncreaseinQuantity

FROM coffee\_shop\_Sales

WHERE MONTH(transaction\_date) IN (4,5)

GROUP BY MONTH(transaction\_date)

ORDER BY MONTH(transaction\_date);



1. Total Sales, Total Quantity Sold and Total orders for particular date

SELECT

CONCAT(ROUND(SUM(unit\_price\*transaction\_qty)/1000,1),'k') As Total\_Sales,

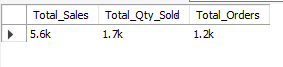
CONCAT(ROUND(SUM(transaction\_qty)/1000,1),'k') As Total\_Qty\_Sold,

CONCAT(ROUND(COUNT(transaction\_id)/1000,1),'k') as Total\_Orders

FROM coffee\_shop\_Sales

WHERE transaction\_date='2023-05-18'

;



1. Sales analysis by weekdays and weekends

Weekends - Sat-Sun

Weekdays - Mon-Fri

SELECT

CASE

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

ELSE

'Weekdays'

END AS Day\_Type,

CONCAT(ROUND(SUM(unit\_price\*transaction\_qty)/1000,1),'k') AS TotalSales

FROM coffee\_Shop\_sales

WHERE MONTH(transaction\_date)=5

GROUP BY

CASE

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

ELSE

'Weekdays'

END

;



1. Total Sales by store location in particular month

SELECT

store\_location,

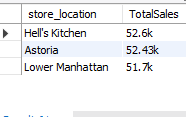
CONCAT(ROUND(SUM(unit\_price\*transaction\_qty)/1000,2),'k') as TotalSales

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5

GROUP BY store\_location

ORDER BY SUM(unit\_price\*transaction\_qty) desc ;



1. Average sales in a particular month

SELECT

AVG(total\_Sales) as AverageSales

FROM

(

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

FROM

coffee\_Shop\_Sales

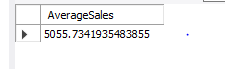
WHERE

MONTH(transaction\_date)=5

GROUP BY transaction\_date

) as InnerQuery

;



1. Daily sales for particular month

SELECT

DAY(transaction\_date) as Day\_Of\_Month,

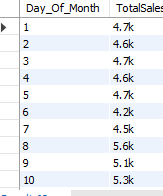
CONCAT(ROUND(SUM(unit\_price\*transaction\_qty)/1000,1),'k') as TotalSales

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5

GROUP BY DAY(transaction\_date)

ORDER BY DAY(transaction\_date);



1. Finding if total sales is below average or above average

SELECT

CASE

WHEN total\_Sales>avg\_sales THEN 'Above Average'

WHEN total\_Sales<avg\_sales THEN 'Below Average'

ELSE 'Average'

END as Sales\_Status,

total\_sales

FROM

(

SELECT

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

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

AVG(SUM(unit\_price\*transaction\_qty)) OVER () AS AVG\_SALES

FROM

COFFEE\_SHOP\_SALES

WHERE MONTH(transaction\_date)=5

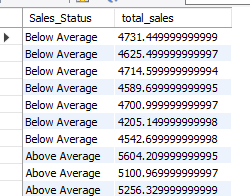
group by

DAY(transaction\_date)

) as Sales\_data

ORDER BY

day\_of\_month;



1. Sales by product category for particular month

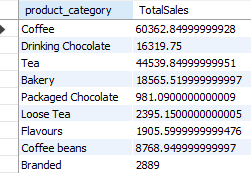
SELECT product\_category,SUM(unit\_price\*transaction\_qty) as TotalSales

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5

GROUP BY

product\_category;



1. Total sales, Total quantity and Total order for particular hour, day and month

SELECT CONCAT(ROUND(SUM(unit\_price\*transaction\_qty)/1000,1),'k') as TotalSales,

SUM(transaction\_qty) as Total\_qty\_sold,

COUNT(\*) as TotalOrders

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5

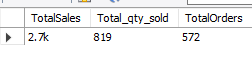
AND

DAYOFWEEK(transaction\_date)=2

AND

HOUR(transaction\_time)=8

;



1. Total sales by hour in a particular month

SELECT HOUR(transaction\_time) as hourofday,

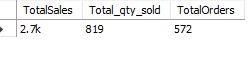
SUM(unit\_price\*transaction\_qty) as TotalSales

FROM coffee\_shop\_sales

WHERE MONTH(transaction\_date)=5

GROUP BY HOUR(transaction\_time)

ORDER BY HOUR(transaction\_time);



1. Total Sales by days in a 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 '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

coffee\_shop\_sales

WHERE MONTH(transaction\_date) =5

GROUP BY

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

;

