HELLO,

MY NAME IS YATISH GUPTA AND I HAVE CREATED A SQL PROJECT, FOR A COFFEE SHOP BUSINESS, WHERE I HAVE UTILIZED MULTIPLE SQL QEURIES TO EXTRACT DATA FOR VARIOUS SCENARIOS. EVENTUALLY USING THE SAME DATA TO ALSO CREATE POWER BI REPORT AND DASHBOARD.

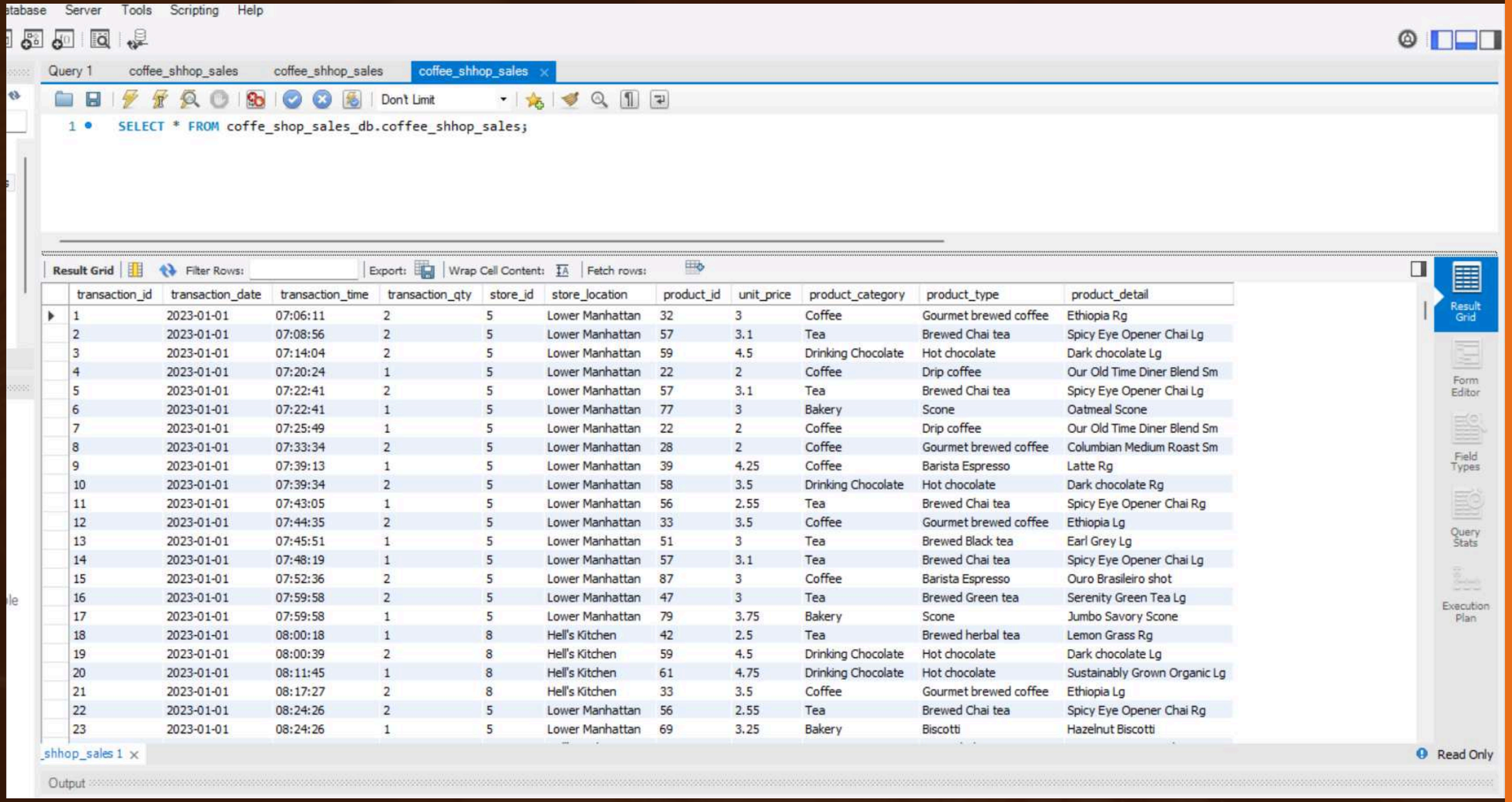
STEPS FOR THE PROCESS

- Data Walkthrough
- Raw data file preparation
- Creating Database
- Importing File
- Cleaning Imported File
- Changing Data Types
- Firing SQL Queries for Business Requirements
- Storing Results
- Preparing SQL Documents

FUNCTIONALITITES USED IN THE PROCESS

- STR_TO_DATE
- ROUND
- SUM
- COUNT
- AVG
- LAG
- MONTH
- DAY
- DAYOFWEEK
- SELECT
- ALIAS
- MAX/ MIN
- HOUR
- ALTER TABLE
- UPDATE TABLE
- CHANGE COLUMN
- WHERE
- GROUP BY
- CASE
- ORDER BY
- LIMIT
- WINDOW FUNCTIONS
- JOINS
- SUBQUERIES

DATABASE OVERVIEW



transaction_id	transaction_date	transaction_time	transaction_qty	store_id	store_location	product_id	unit_price	product_category	product_type	product_detail
1	2023-01-01	07:06:11	2	5	Lower Manhattan	32	3	Coffee	Gourmet brewed coffee	Ethiopia Rg
2	2023-01-01	07:08:56	2	5	Lower Manhattan	57	3.1	Tea	Brewed Chai tea	Spicy Eye Opener Chai Lg
3	2023-01-01	07:14:04	2	5	Lower Manhattan	59	4.5	Drinking Chocolate	Hot chocolate	Dark chocolate Lg
4	2023-01-01	07:20:24	1	5	Lower Manhattan	22	2	Coffee	Drip coffee	Our Old Time Diner Blend Sm
5	2023-01-01	07:22:41	2	5	Lower Manhattan	57	3.1	Tea	Brewed Chai tea	Spicy Eye Opener Chai Lg
6	2023-01-01	07:22:41	1	5	Lower Manhattan	77	3	Bakery	Scone	Oatmeal Scone
7	2023-01-01	07:25:49	1	5	Lower Manhattan	22	2	Coffee	Drip coffee	Our Old Time Diner Blend Sm
8	2023-01-01	07:33:34	2	5	Lower Manhattan	28	2	Coffee	Gourmet brewed coffee	Columbian Medium Roast Sm
9	2023-01-01	07:39:13	1	5	Lower Manhattan	39	4.25	Coffee	Barista Espresso	Latte Rg
10	2023-01-01	07:39:34	2	5	Lower Manhattan	58	3.5	Drinking Chocolate	Hot chocolate	Dark chocolate Rg
11	2023-01-01	07:43:05	1	5	Lower Manhattan	56	2.55	Tea	Brewed Chai tea	Spicy Eye Opener Chai Rg
12	2023-01-01	07:44:35	2	5	Lower Manhattan	33	3.5	Coffee	Gourmet brewed coffee	Ethiopia Lg
13	2023-01-01	07:45:51	1	5	Lower Manhattan	51	3	Tea	Brewed Black tea	Earl Grey Lg
14	2023-01-01	07:48:19	1	5	Lower Manhattan	57	3.1	Tea	Brewed Chai tea	Spicy Eye Opener Chai Lg
15	2023-01-01	07:52:36	2	5	Lower Manhattan	87	3	Coffee	Barista Espresso	Ouro Brasileiro shot
16	2023-01-01	07:59:58	2	5	Lower Manhattan	47	3	Tea	Brewed Green tea	Serenity Green Tea Lg
17	2023-01-01	07:59:58	1	5	Lower Manhattan	79	3.75	Bakery	Scone	Jumbo Savory Scone
18	2023-01-01	08:00:18	1	8	Hell's Kitchen	42	2.5	Tea	Brewed herbal tea	Lemon Grass Rg
19	2023-01-01	08:00:39	2	8	Hell's Kitchen	59	4.5	Drinking Chocolate	Hot chocolate	Dark chocolate Lg
20	2023-01-01	08:11:45	1	8	Hell's Kitchen	61	4.75	Drinking Chocolate	Hot chocolate	Sustainably Grown Organic Lg
21	2023-01-01	08:17:27	2	8	Hell's Kitchen	33	3.5	Coffee	Gourmet brewed coffee	Ethiopia Lg
22	2023-01-01	08:24:26	2	5	Lower Manhattan	56	2.55	Tea	Brewed Chai tea	Spicy Eye Opener Chai Rg
23	2023-01-01	08:24:26	1	5	Lower Manhattan	69	3.25	Bakery	Biscotti	Hazelnut Biscotti

PART - 1

WRITING QUERIES FOR THE FOLLOWING KPI REQUIREMENTS --

1. **Total Sales Analysis:**
   - Calculate the total sales for each respective month.
   - Determine the month-on-month increase or decrease in sales.
   - Calculate the difference in sales between the selected month and the previous month.
2. **Total Orders Analysis:**
   - Calculate the total number of orders for each respective month.
   - Determine the month-on-month increase or decrease in the number of orders.
   - Calculate the difference in the number of orders between the selected month and the previous month.
3. **Total Quantity Sold Analysis:**
   - Calculate the total quantity sold for each respective month.
   - Determine the month-on-month increase or decrease in the total quantity sold.
   - Calculate the difference in the total quantity sold between the selected month and the previous month.

1 - TOTAL SALES ANALYSIS

```sql
-- ----------------------------------TOTAL SALES KPI - MOM DIFFERENCE AND MOM GROWTH


WITH monthly_sales AS (
    SELECT
        MONTH(transaction_date) AS month,
        ROUND(SUM(unit_price * transaction_qty)) AS total_sales
    FROM
        coffee_shhop_sales
    WHERE
        Month(transaction_date) in (1,2,3,4,5)
    GROUP BY
        MONTH(transaction_date)
)

SELECT
    month,
    total_sales,
    ROUND(
        (total_sales - LAG(total_sales) OVER (ORDER BY month)) * 100.0 /
        NULLIF(LAG(total_sales) OVER (ORDER BY month), 0), 2
    ) AS mom_increase_percentage
FROM
    monthly_sales
ORDER BY
    month;
```

2- TOTAL ORDERS ANALYSIS - MOM DIFF AND MOM GROWTH

```sql
-- ---------------------------------------------------------------------- TOTAL ORDERS KPI -

WITH monthly_orders AS (
    SELECT
        MONTH(transaction_date) AS month,
        ROUND(COUNT(transaction_id)) AS total_orders
    FROM
        coffee_shhop_sales
    WHERE
        MONTH(transaction_date) IN (4, 5) -- April and May
    GROUP BY
        MONTH(transaction_date)
)

SELECT
    month,
    total_orders,
    ROUND(
        (total_orders - LAG(total_orders) OVER (ORDER BY month)) * 100.0 /
        NULLIF(LAG(total_orders) OVER (ORDER BY month), 0), 2
    ) AS mom_increase_percentage
FROM
    monthly_orders
ORDER BY
    month;
```

3- TOTAL QUANTITY SOLD ANALYSIS- MOM DIFF AND GROWTH

```sql
WITH monthly_orders AS (
    SELECT
        MONTH(transaction_date) AS month,
        ROUND(COUNT(transaction_id)) AS total_orders
    FROM
        coffee_shhop_sales
    WHERE
        MONTH(transaction_date) IN (4, 5) -- April and May
    GROUP BY
        MONTH(transaction_date)
)

SELECT
    month,
    total_orders,
    ROUND(
        (total_orders - LAG(total_orders) OVER (ORDER BY month)) * 100.0 /
        NULLIF(LAG(total_orders) OVER (ORDER BY month), 0), 2
    ) AS mom_increase_percentage
FROM
    monthly_orders
ORDER BY
    month;
```

PART - 2

WRITING QUERIES FOR THE FOLLOWING CHART REQUIREMENTS --

1. **Calendar Heat Map:**
   - Implement a calendar heat map that dynamically adjusts based on the selected month from a slicer.
   - Each day on the calendar will be color-coded to represent sales volume, with darker shades indicating higher sales.
   - Implement tooltips to display detailed metrics (Sales, Orders, Quantity) when hovering over a specific day.
2. **Sales Analysis by Weekdays and Weekends:**
   - Segment sales data into weekdays and weekends to analyze performance variations.
   - Provide insights into whether sales patterns differ significantly between weekdays and weekends.
3. **Sales Analysis by Store Location:**
   - Visualize sales data by different store locations.
   - Include month-over-month (MoM) difference metrics based on the selected month in the slicer.
   - Highlight MoM sales increase or decrease for each store location to identify trends.

4. **Daily Sales Analysis with Average Line:**
   - Display daily sales for the selected month with a line chart.
   - Incorporate an average line on the chart to represent the average daily sales.
   - Highlight bars exceeding or falling below the average sales to identify exceptional sales days.
5. **Sales Analysis by Product Category:**
   - Analyze sales performance across different product categories.
   - Provide insights into which product categories contribute the most to overall sales.
6. **Top 10 Products by Sales:**
   - Identify and display the top 10 products based on sales volume.
   - Allow users to quickly visualize the best-performing products in terms of sales.
7. **Sales Analysis by Days and Hours:**
   - Utilize a heat map to visualize sales patterns by days and hours.
   - Implement tooltips to display detailed metrics (Sales, Orders, Quantity) when hovering over a specific day-hour.

CALEDNDAR TABLE - DAILY SALES ,ORDERS AND QUANTITIES

```sql
-- ----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_shhop_sales
WHERE
    transaction_date = '2023-05-18';      -- ---for 18th may


    -- ---------If we want to get exact Rounded off values
SELECT
    CONCAT(ROUND(SUM(unit_price * transaction_qty) / 1000, 1),'K') AS total_sales,
    CONCAT(ROUND(COUNT(transaction_id) / 1000, 1),'K') AS total_orders,
    CONCAT(ROUND(SUM(transaction_qty) / 1000, 1),'K') AS total_quantity_sold
FROM
    coffee_shhop_sales
WHERE
    transaction_date = '2023-05-18'; -- -------------For 18 May 2023
```

DAILY SALES VS AVERAGE SALES

```sql
-- ------------------------------------------------------------------
-- ------------------------COMPARING DAILY SALES WITH AVERAGE SALES
SELECT
    day_of_month,
    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_shhop_sales
    WHERE
        MONTH(transaction_date) = 5  -- Filter for May
    GROUP BY
        DAY(transaction_date)
) AS sales_data
ORDER BY
    day_of_month;
```

SALES BY WEEKDAY / WEEKEND

```sql
-- ----------------------------------------------------------------
-- ------------------------------------ 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_shhop_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 -- LOCATION, PRODUCT CATEGORY, PRODUCTS

```sql
-- ------------------------------------------------------------------------------
-- ----------------------------------------------------------SALES BY STORE LOCATION
SELECT
    store_location,
    SUM(unit_price * transaction_qty) as Total_Sales
FROM coffee_shhop_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_shhop_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

```sql
-- ---------------------------------------------------------------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_shhop_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)
```

SALES FROM MON - SUN FOR A SPECIFIC MONTH

```sql
-- --TO GET SALES FROM MONDAY TO SUNDAY FOR MONTH OF 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
    coffee_shhop_sales
WHERE
    MONTH(transaction_date) = 5 -- Filter for May (month number 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;
```

COFFEE SHOP SALES

Sales Report

FILTER PANEL

Month: Jan 2023

Jan 2023

Mon	Tue	Wed	Thu	Fri	Sat	Sun
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Hover On this visual to see the details

Total Sales
$82K
▲ +inf vs LM

Total orders
17314
▲ +inf vs LM

Total Quantity Sold
24870
▲ +100.0% | +24.9k vs LM

Sales trend over the period

Sales by weekday / weekend

Sales by store_location

- Hell's Kitchen | $27.82K — ▲ +inf vs LM
- Astoria | $27.31K — ▲ +inf vs LM
- Lower Manhattan | $26.54K — ▲ +inf vs LM

Sales by product category

- Coffee | $31.26K — ▲ +inf
- Tea | $22.62K — ▲ +inf
- Bakery | $9.54K — ▲ +inf
- Drinking Chocolate | $8.34K — ▲ +inf
- Coffee beans | $5.25K — ▲ +...
- Branded | $1.89K
- Loose Tea | $1.29K
- Flavours | $0.97K
- Packaged Chocolate | $0.52K

Sales by product category

- Barista Espresso | $10.46K — ▲ +inf
- Brewed Chai tea | $8.76K — ▲ +inf
- Hot chocolate | $8.34K — ▲ +inf
- Gourmet brewed coffee | $8.07K — ▲ +inf
- Brewed Black tea | $5.54K — ▲ +inf
- Brewed herbal tea | $5.44K — ▲ +inf
- Organic brewed coffee | $4.61K — ▲ +inf
- Premium brewed coffee | $4.53K — ▲ +inf
- Scone | $4.35K — ▲ +inf
- Drip coffee | $3.58K — ▲ +inf

Sales by Days | Hours

THANK YOU