

Bright Coffee Shop Methodology

SQL Version

Busile Ndhlovu

Table of Contents

Topic	Page Number
<i>Objective</i>	3
<i>Data Overview</i>	3
<i>Data Flow & Architecture Diagram</i>	3
<i>Tools and Techniques</i>	4
<i>Understanding the dataset</i>	4
<i>Best-selling Products</i>	7
<i>Peak Sale Times</i>	8
<i>Sale Trend Across Time Interval</i>	9
<i>Recommendations for Growth</i>	10
<i>Initiative for Growth</i>	12
<i>Conclusion</i>	13

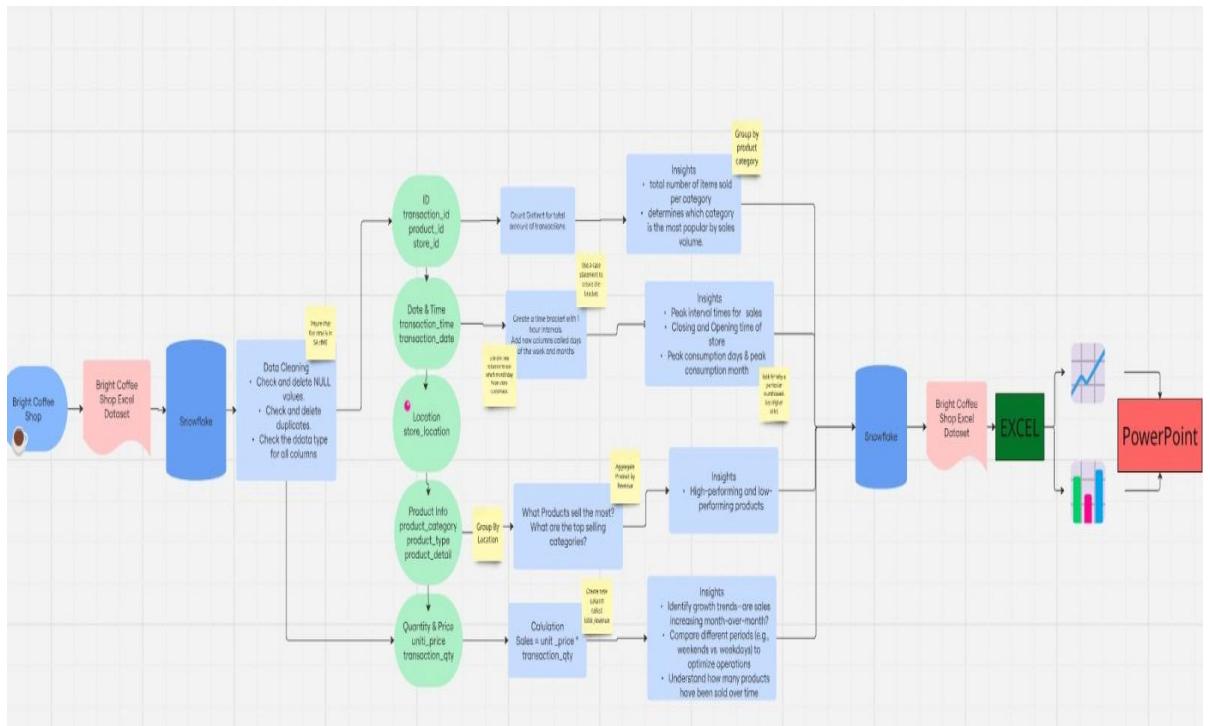
Objective

Extract actionable insights from historical data to help grow the company's revenue and improve product performance.

Data Overview

- The dataset from Bright Coffee Shop contain information about the shops transactions and product details.
- The dataset was from the months January, February, March, April, May and June from the year 2023.
- There was a total of 149116 transactions.
- The dataset provided also revealed that the store opens at 6:00 am and closes at 21:00 pm.

Data Flow & Architecture Diagram



This Data Flow & Architecture Diagram outlines the steps that will be taken to achieve our goal which is the completion of this project on PowerPoint as a presentation.

Tools and Techniques

- Planning: Miro for Data Flow & Architecture Diagram.
- Data Analysis Tools: SQL on Snowflake and MS Excel for statistical analysis and visualization.
- Visualization Tools: PowerPoint for visualization.

Understanding the dataset

1.Handle Missing Values:

Code used to find missing values in all columns of the user profile table.

Missing values in this case are considered as NULL values.

```
--DATA CLEANING
--missing values/nulls

SELECT
    SUM( CASE WHEN transaction_id IS NULL THEN 1 ELSE 0 END) AS transaction_id_missing,
    SUM( CASE WHEN transaction_date IS NULL THEN 1 ELSE 0 END) AS transaction_id_missing,
    SUM( CASE WHEN transaction_time IS NULL THEN 1 ELSE 0 END) AS transaction_id_missing,
    SUM( CASE WHEN transaction_qty IS NULL THEN 1 ELSE 0 END) AS transaction_qty_missing,
    SUM( CASE WHEN store_id IS NULL THEN 1 ELSE 0 END) AS store_id_missing,
    SUM( CASE WHEN store_location IS NULL THEN 1 ELSE 0 END) AS store_location_missing,
    SUM( CASE WHEN product_id IS NULL THEN 1 ELSE 0 END) AS pro_id_missing,
    SUM( CASE WHEN unit_price IS NULL THEN 1 ELSE 0 END) AS unit_price_missing,
    SUM( CASE WHEN product_category IS NULL THEN 1 ELSE 0 END) AS pro_cate_missing,
    SUM( CASE WHEN product_type IS NULL THEN 1 ELSE 0 END) AS pro_type_missing,
    SUM( CASE WHEN product_detail IS NULL THEN 1 ELSE 0 END) AS pro_detail_missing,
FROM "BUSI"."PUBLIC"."BRIGHT_COFFEE_SHOP";
```

Returned Output:

TRANSACTION_ID_MISSING	TRANSACTION_ID_MISSING	TRANSACTION_ID_MISSING	TRANSACTION_QTY_MISSING	STORE_ID_MISSING
1	0	0	0	0

The code returned 0 missing values in all columns therefore no NULL values.

2.Checking for duplicates

```
--checking for duplicates
SELECT *,
       COUNT(*) AS duplicate_count
FROM "BUSI"."PUBLIC"."BRIGHT_COFFEE_SHOP"
GROUP BY transaction_id,
transaction_date,transaction_time,transaction_qty,store_id,store_location,product_id,unit_price,product_category,product_type,product_detail
HAVING COUNT(*) > 1;
```

Found no duplicates.

3. Changing the format of the digit in unit price from , to .

This is done for easy data processing using a comma instead might cause errors or misinterpretations during calculations.

```
35 --changing the format of the digit in unit price , to .
36
37 UPDATE bright_coffee_shop
38 SET unit_price = CAST(REPLACE(unit_price, ',', '.') AS DECIMAL(10,2));
39
```

↳ Results ↵ Chart

# number of rows updated	# number of multi-joined rows updated
1	149116

4. Calculating the total amount

```
--calculating the total amount and creating a new column for it

ALTER TABLE bright_coffee_shop
ADD COLUMN total_amount DECIMAL(10,2); --creating a new column total amount
|
UPDATE bright_coffee_shop
SET total_amount = unit_price * transaction_qty; --populating the new column
```

This code alters the table to add a new column called total amount, the second part of the code populates the column we created.

5. Checking what the total number of transaction is.

```
61 --Count total number of transactions
62
63 SELECT COUNT(transaction_id) AS total_transactions
64 FROM bright_coffee_shop;
65
```

↳ Results ↵ Chart

TOTAL_TRANSACTIONS
1 149116

6. Checking the min and max transaction_time

This will help with developing an accurate time bucket, this also helps see what time the store opens and closes.

```
54 --time bracket
55
56
57 SELECT MIN(transaction_time),
58      MAX(transaction_time)
59 FROM "BUSI"."PUBLIC"."BRIGHT_COFFEE_SHOP" ;
60
```

↳ Results ↵ Chart

⌚ MIN(TRANSACTION_TIME)	⌚ MAX(TRANSACTION_TIME)
1 06:00:00	20:59:32

7.Creating a new column transaction_time

```
9
10    SELECT transaction_date,
11          TO_DATE(transaction_date, 'YYYY/MM/DD') AS T_Date,
12          DAYNAME(T_Date) AS Days_of_the_week,
13          MONTHNAME(T_Date) AS Month_Name,
14          COUNT(transaction_id) AS number_of_sales,
15          COUNT(product_id) AS unique_products_sold,
16          SUM(transaction_qty * TO_NUMBER(REPLACE(unit_price, ',', '.'))) AS total_amount,
17          product_category,
18          product_type,
19          store_location,
20          product_detail,
21          CASE
22              WHEN DATE_PART('hour', transaction_time) BETWEEN 6 AND 9 THEN 'Early Morning'
23              WHEN DATE_PART('hour', transaction_time) BETWEEN 10 AND 13 THEN 'Mid-Morning'
24              WHEN DATE_PART('hour', transaction_time) BETWEEN 14 AND 17 THEN 'Afternoon'
25              WHEN DATE_PART('hour', transaction_time) BETWEEN 18 AND 20 THEN 'Evening'
26              ELSE 'Unknown'
27          END AS time_bucket
28      FROM "BUSI"."PUBLIC"."COFFEE_SHOP"
29  GROUP BY ALL;
```

Created a new column called transaction_time_bucket .

Grouping people this way helps capture behaviours and needs unique to each time frame.

Also Created an time bracket to group individuals into meaningful, manageable categories for analysis and decision-making. Specifically, here's how I created an time bracket with differences like 4 hours years between categories:

Time Bracket	Category
6:00 – 9:00 am	Early Morning
10:00 – 13:00 pm	Mid-Morning
14:00 : 17:00 pm	Afternoon
18:00 -21:00 pm	Evening

In the above code I also added two new columns months and days of the week.

Returned Output:

	A TRANSACTION_DATE	C T_DATE	A DAYS_OF_THE_WEEK	A MONTH_NAME	# NUMBER_OF_SALES	# UNIQUE_PRODUCTS_
1	2023/01/01	2023-01-01	Sun	Jan	1	
2	2023/01/01	2023-01-01	Sun	Jan	3	
3	2023/01/01	2023-01-01	Sun	Jan	1	
4	2023/01/01	2023-01-01	Sun	Jan	2	
5	2023/01/01	2023-01-01	Sun	Jan	2	

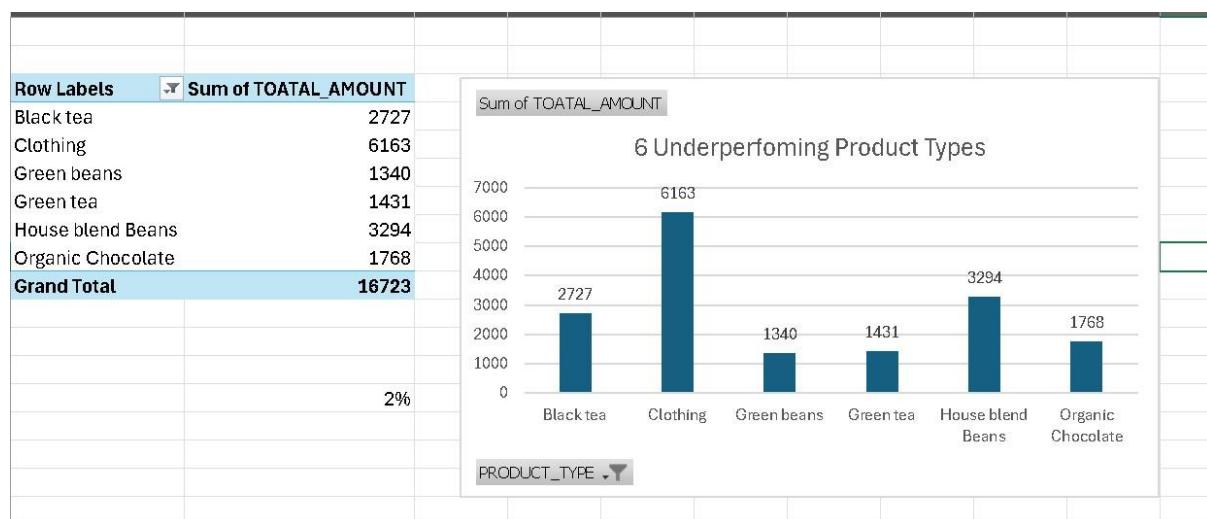
Best Selling Products

Which products generate the most revenue

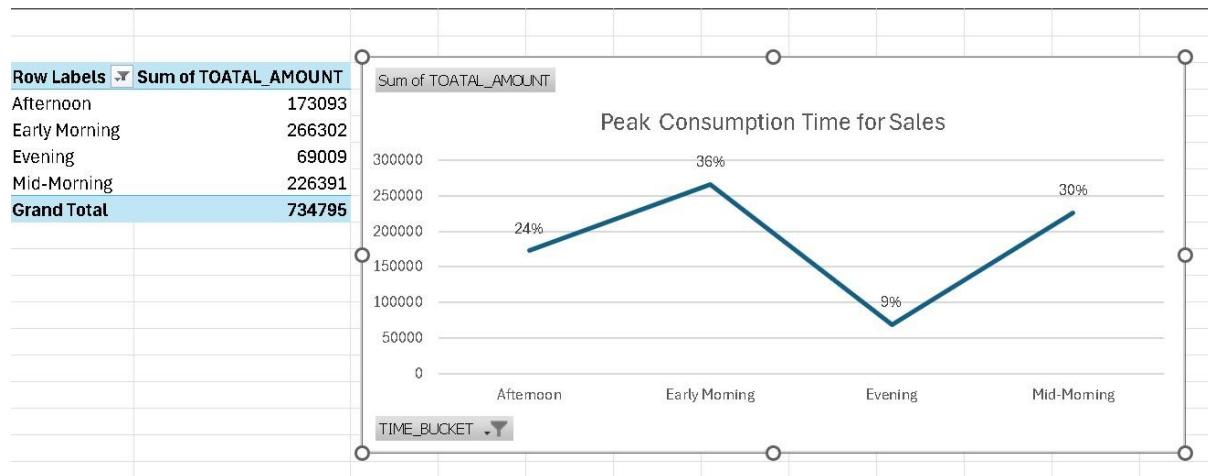
Filtered the table to show the top 6 best-selling product type these 6 products make up 60% of our total revenue.



The 6 products that generate the least towards the total revenue this 6 products make up 2 % of our total revenue.



Peak Sale Time

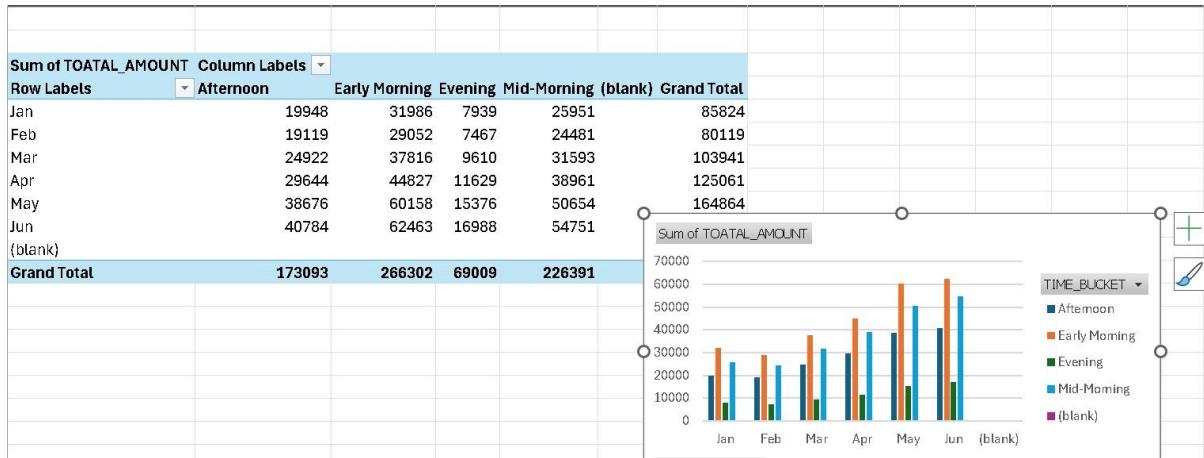


The peak sale time is from 6:00 – 9:00 am

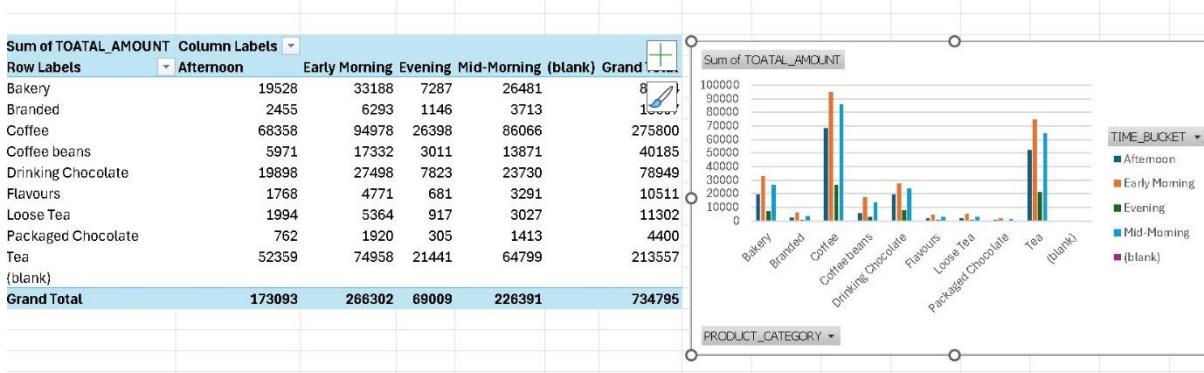
Time Bracket	Category
6:00 – 9:00 am	Early Morning
10:00 – 13:00 pm	Mid-Morning
14:00 : 17:00 pm	Afternoon
18:00 -21:00 pm	Evening

Sale Trend Across Time Interval

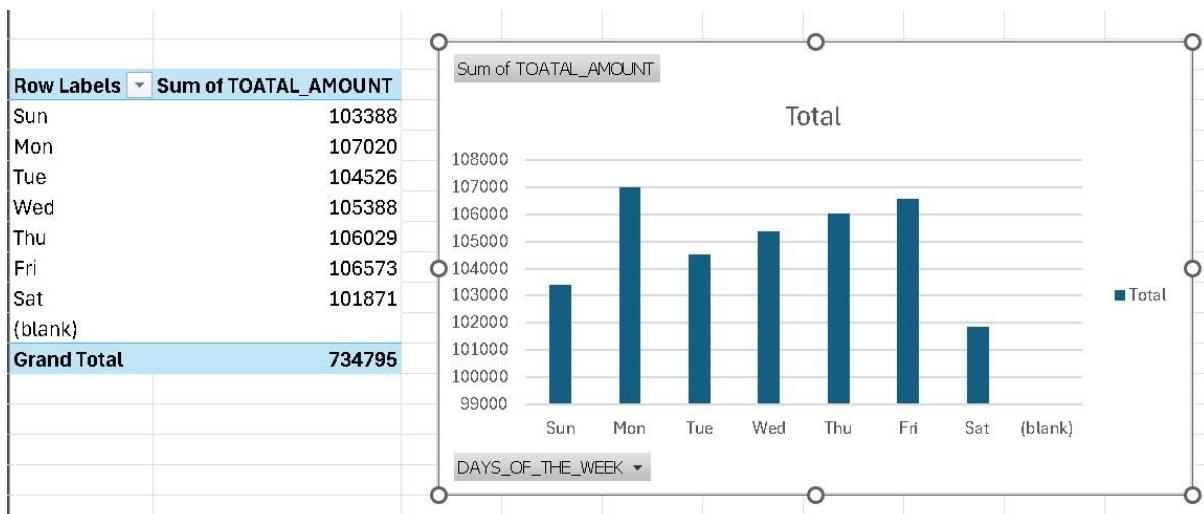
The below graph shows the months Bright Coffee Shop generates the most revenue at what time of the day. From this graph we are able to see which seasons the store makes the most sales which would be winter as there is an increase in sales from the month of May to June.



The below graph shows the categories that generate the most revenue at what time of the day in all stores.



The bar graph below shows the days of the week where generates the most revenue



Recommendations for Growth

Revenue Growth Strategies

1. Enhance Best-Selling Products

- Introduce **premium versions** or seasonal variations of the top 6 products to boost sales.
- Offer **subscription plans** for regular customers to ensure repeat purchases.

2. Revamp Underperforming Items

- **Bundle slower-selling items** with popular products to increase visibility.
- Launch **limited-time promotions** to test demand before deciding whether to phase them out.

3. Optimize Peak Hours Sales

- Introduce **breakfast bundles** during **Early Morning** (6–9 AM) and **Mid-Morning** (10 AM–1 PM).
- Implement **express service options** during peak hours to reduce waiting times.

4. Customer Engagement Initiatives

Loyalty Program & Subscription Model

- Reward frequent customers with **discounts, freebies, or exclusive early access** to new menu items.
- Introduce a **coffee subscription** for regular customers (e.g., unlimited coffee for a fixed monthly fee).

5. Personalized Marketing

- Use **customer purchase history** to send targeted promotions (e.g., a discount on a customer's favourite drink).
- Leverage **SMS or app notifications** to remind customers about special offers based on peak sales times.
-

Store & Sales Optimization

1. Expand Popular Store Locations

- Consider **opening new branches** in high-performing areas.
- Optimize **staffing levels** to ensure enough employees during peak hours.

2. Weekend Promotional Campaigns

- Since weekdays perform better than weekends, introduce **weekend-exclusive promotions** (e.g., "Saturday Family Coffee Discount").
- Organize **community events or coffee-tasting experiences** to draw in more foot traffic.

3. Seasonal and Monthly Promotions

- Based on monthly sales data, create **seasonal offers** aligned with high-demand months.
- Offer **holiday-themed products** to drive sales during festive seasons.

Initiatives for Growth

1. Product Strategy

- Enhance best-selling items → Introduce limited-edition flavors or seasonal versions.
- Revamp slow-selling products → Adjust pricing, reposition or bundle them with popular items.

2. Peak Time Optimization

- Improve early morning efficiency → Train staff to handle high-demand hours.
- Create express menu options → Faster service during peak rush to increase sales volume.

3. Marketing & Promotions

- Target weekday customers → Special promotions for high-traffic days (e.g., Monday deals).
- Weekend sales growth initiatives → Events, discount campaigns, or exclusive weekend products.
- Loyalty program introduction → Reward frequent buyers to encourage repeat purchases.

4. Operational Improvements

- Adjust staffing during peak hours → Ensure high sales periods have sufficient employee coverage.
- Optimize inventory stocking → Align product availability with high-demand periods.

5. Data-Driven Decision Making

- Continuously monitor trends → Implement ongoing data analysis to refine strategy.
- Leverage customer preferences → Use purchase history insights for targeted marketing campaigns.

Conclusion

The analysis of Bright Coffee Shop's sales data has provided valuable insights into the business's revenue drivers, customer behaviour, and product performance. By identifying **best-selling products**, **peak consumption hours**, and **high-performing days**, we can strategically refine operations and marketing efforts to maximize growth.

A few key takeaways:

- **Best-selling products** contribute **60%** of total sales, requiring further optimization for maximum revenue.
- **Underperforming products** account for only **2%**, needing adjustments in pricing, marketing, or removal.
- **Weekday traffic surpasses weekends**, with **Monday leading sales**, followed by Wednesday.
- **Early morning (6–9 AM) and mid-morning (10 AM–1 PM)** drive the highest revenue, highlighting prime business hours.
- **Seasonal patterns** suggest targeted promotions should align with high-performing months.

By implementing **strategic growth initiatives**, Bright Coffee Shop can **increase revenue**, **enhance customer engagement**, and **boost overall business performance**.

This analysis equips **Bright Coffee Shop** with a clear path for **revenue growth and product performance improvements**. By implementing **data-driven strategies**, the company can strengthen its competitive position and build long-term customer loyalty.