Gencafe

Sales performance analysis (01 - 07.2024)

TABLE OF CONTENT

OBJECTIVES

- **Sales Trend**
  - Analyze daily, weekly, and monthly sales trend to identify peak sales periods
  - Determine which products and regions contribute most to revenue generation.

- **Evaluate promotion effectiveness**
  - Assess the impact of different promotional strategies (e.g., discounts, bundles) on sales performance.
  - Identify the return on investment (ROI) for various promotional campaigns.

- **Analyze Customer Behavior**
  - Examine the distribution of sales channels (e.g., online, in-store, delivery) and customer preferences.
  - Identify patterns in customer purchases based on time of day, region, or store location.

#Sales data

```python
import pandas as pd

# Load the CSV file
df = pd.read_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\sales_data.csv")

# Check info for the DataFrame
print("Data Info:")
print(df.info())

print("Data Description")
print(df.describe())

# Check for duplicate rows
duplicates = df[df.duplicated()]
print("Duplicate Rows:")
print(duplicates)

# Remove duplicates and keep the first occurrence
df_cleaned = df.drop_duplicates()

# Save the cleaned data to a new file (use .csv for CSV format)
df_cleaned.to_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\sales_data.csv", index=False)
```

Results

```
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Date                5000 non-null   object
 1   Product             5000 non-null   object
 2   Quantity Sold       5000 non-null   int64
 3   Sale Price          5000 non-null   float64
 4   Region              5000 non-null   object
 5   Store Location      5000 non-null   object
 6   Promotions Applied  5000 non-null   object
 7   Sales Channel       5000 non-null   object
 8   Customer Count      5000 non-null   int64
```

- Import **pandas** and use **df** for dataset
- Use **df.info** and **df.describe** for overall check of the data
- To ensure no duplicated values, use **df.duplicated** and the **df.drop_duplicate**s

#Promotion data

```python
import pandas as pd

# Load the CSV file
df = pd.read_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\promotion.csv")

# Check info for the DataFrame
print("Data Info:")
print(df.info())

print("Data Description")
print(df.describe())

# Check for duplicate rows
duplicates = df[df.duplicated()]
print("Duplicate Rows:")
print(duplicates)

# Remove duplicates and keep the first occurrence
df_cleaned = df.drop_duplicates()

# Save the cleaned data to a new file (use .csv for CSV format)
df_cleaned.to_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\promotion.csv", index=False)
```

Results

```
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Promotion ID      5000 non-null   object
 1   Product           5000 non-null   object
 2   Promotion Type    5000 non-null   object
 3   Start Date        5000 non-null   object
 4   End Date          5000 non-null   object
 5   Regions Targeted  5000 non-null   object
 6   Success Rate (%)  5000 non-null   float64
```

- Import **pandas** and use **df** for dataset
- Use **df.info** and **df.describe** for overall check of the data
- To ensure no duplicated values, use **df.duplicated** and the **df.drop_duplicate**s

#Inventory data

```python
import pandas as pd

# Load the CSV file
df = pd.read_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\inventory_data.csv")

# Check info for the DataFrame
print("Data Info:")
print(df.info())

print("Data Description")
print(df.describe())

# Check for duplicate rows
duplicates = df[df.duplicated()]
print("Duplicate Rows:")
print(duplicates)

# Remove duplicates and keep the first occurrence
df_cleaned = df.drop_duplicates()

# Save the cleaned data to a new file (use .csv for CSV format)
df_cleaned.to_csv(r"C:\Users\MY PC\OneDrive\文档\Data Analyst\PROJECT\Gencafe\sales_dataset\inventory_data.csv", index=False)
```

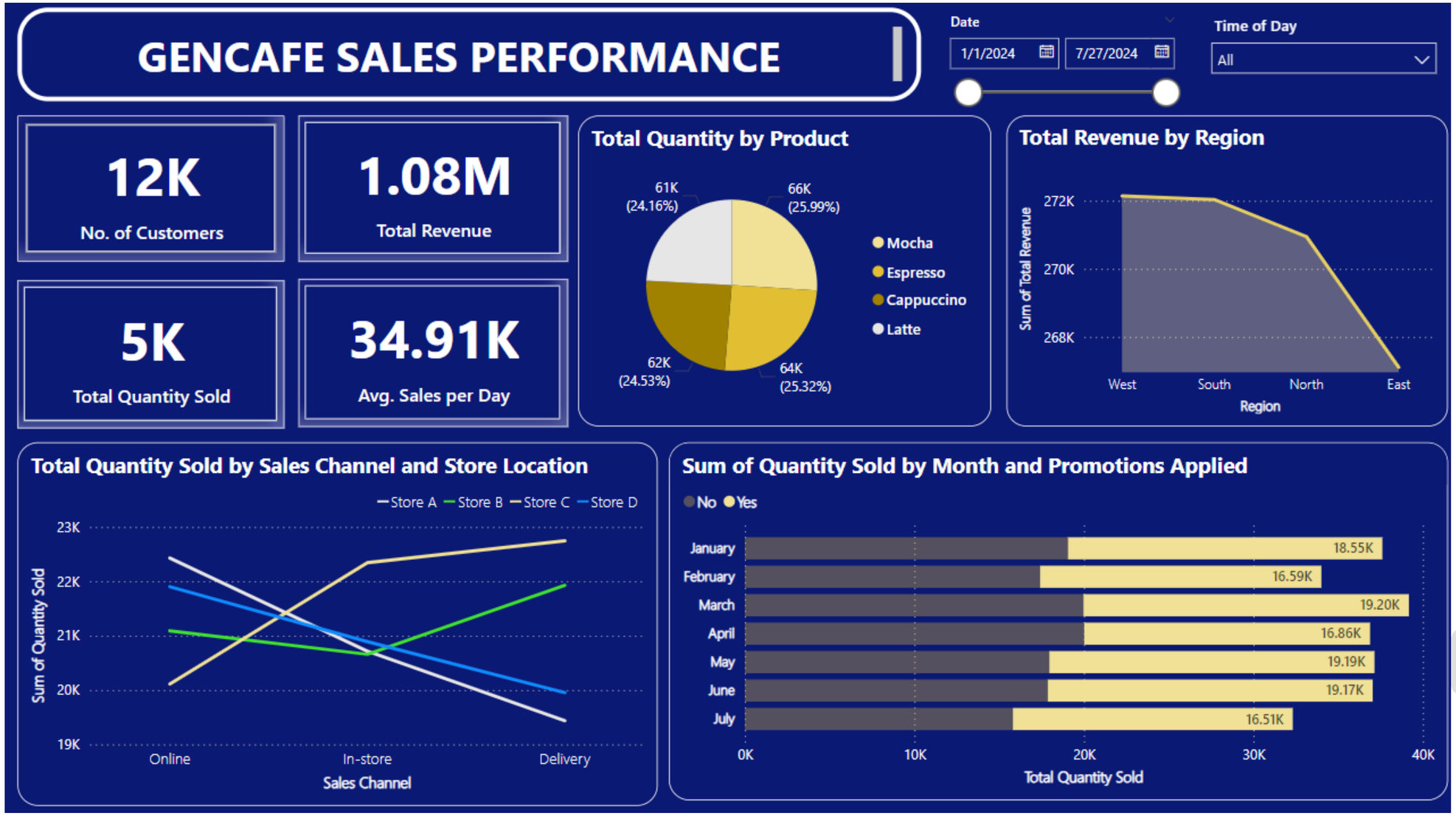
Results

```
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Date               5000 non-null   object
 1   Product            5000 non-null   object
 2   Region             5000 non-null   object
 3   Store Location     5000 non-null   object
 4   Initial Inventory  5000 non-null   int64
 5   Restocks           5000 non-null   int64
 6   Sales Impact       5000 non-null   int64
 7   Current Inventory  5000 non-null   int64
```

- Import **pandas** and use **df** for dataset
- Use **df.info** and **df.describe** for overall check of the data
- To ensure no duplicated values, use **df.duplicated** and the **df.drop_duplicate**s

Using Power Bi and MySQL Workbench for data mining and visualizing

#Overview display ***Sales performance***



UNDERSTANDING

Overall:

- Total Revenue: **1.08M**
- **5,000 products** were sold
- Appeared in 4 regions: **East, West, South, North**, however, it records that West has the most quantity sold contributed to total revenue, after that South, North, East
- We have 4 default names of store (A,B,C,D) with 3 different sales channels (online, in-store, delivery)
    - Store A: has the least quantity sold, in contrast, the most quantity sold in store and delivery. Store B,C,D - almost has the same quantity in store.

```sql
WITH monthlysales AS (
    SELECT
        DATE_FORMAT(`Date`, '%Y-%m') AS month,
        `Product`,
        avg (`Quantity Sold` * `Sale Price`) AS month_sales
    FROM sales_data
    GROUP BY DATE_FORMAT(`Date`, '%Y-%m'), `Product`
),
monthlysales_rankvalue AS (
    SELECT
        month,
        Product,
        month_sales,
        RANK() OVER (PARTITION BY month ORDER BY month_sales DESC) AS rankvalue
    FROM monthlysales
)
SELECT *
FROM monthlysales_rankvalue
WHERE rankvalue = 1
ORDER BY month;
```

Results

month	Product	month_sales	rankvalue
NULL	Mocha	224.49900477706998	1

- Product with the **most average monthly sales: Mocha (~224.5 USD)**
- Execute the same syntax to figure out the product with rank 2,3,4 is Expresso (217.7 USD), Cappuccino (212.1 USD), Latte (211.3) respectively

=> Not much difference among 4 products in terms of sales during the half year of 2024. The most favorable drink is Mocha

```sql
-- the relationship between sales performance and time of Day
select
    `Time of Day`,
    Sum(`Quantity Sold`*`Sale Price`) as total_revenue,
    count(*) as transaction_count,
    avg(`Quantity Sold`*`Sale Price`) as avg_revenue
from sales_data
group by `Time of Day`
order by field(`Time of Day`, 'Morning', 'Afternoon', 'Evening', 'Night');
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Time of Day	total_revenue	transaction_count	avg_revenue
Morning	257893.4499999999	1201	214.73226477935046
Afternoon	268048.77999999997	1268	211.39493690851734
Evening	276502.82999999996	1248	221.55675480769227
Night	279783.1100000004	1283	218.06945440374153

At night, it records the most customers - transactions as well as total revenue.

Using Power Bi and MySQL Workbench for data mining and visualizing

#Overview display ***Promotion Effectiveness***

UNDERSTANDING

- Overall, **the promotion rate is quite high (>50%)** and the quantity sold is high as well.
- We have **4 promotion types (BOGO, Bundle, Free Sample, Discount),** the number of customers is **distributed quite well.**

```sql
select
    `Region`,
    avg(Success_Rate) as avg_success_rate,
    avg (`Quantity Sold`*`Sale Price`) as avg_revenue
from sales_data sd
join promotion p
on
sd.`Region`=p.`Regions Targeted`
group by
    sd.`Region`
order by avg_success_rate desc
limit 4;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

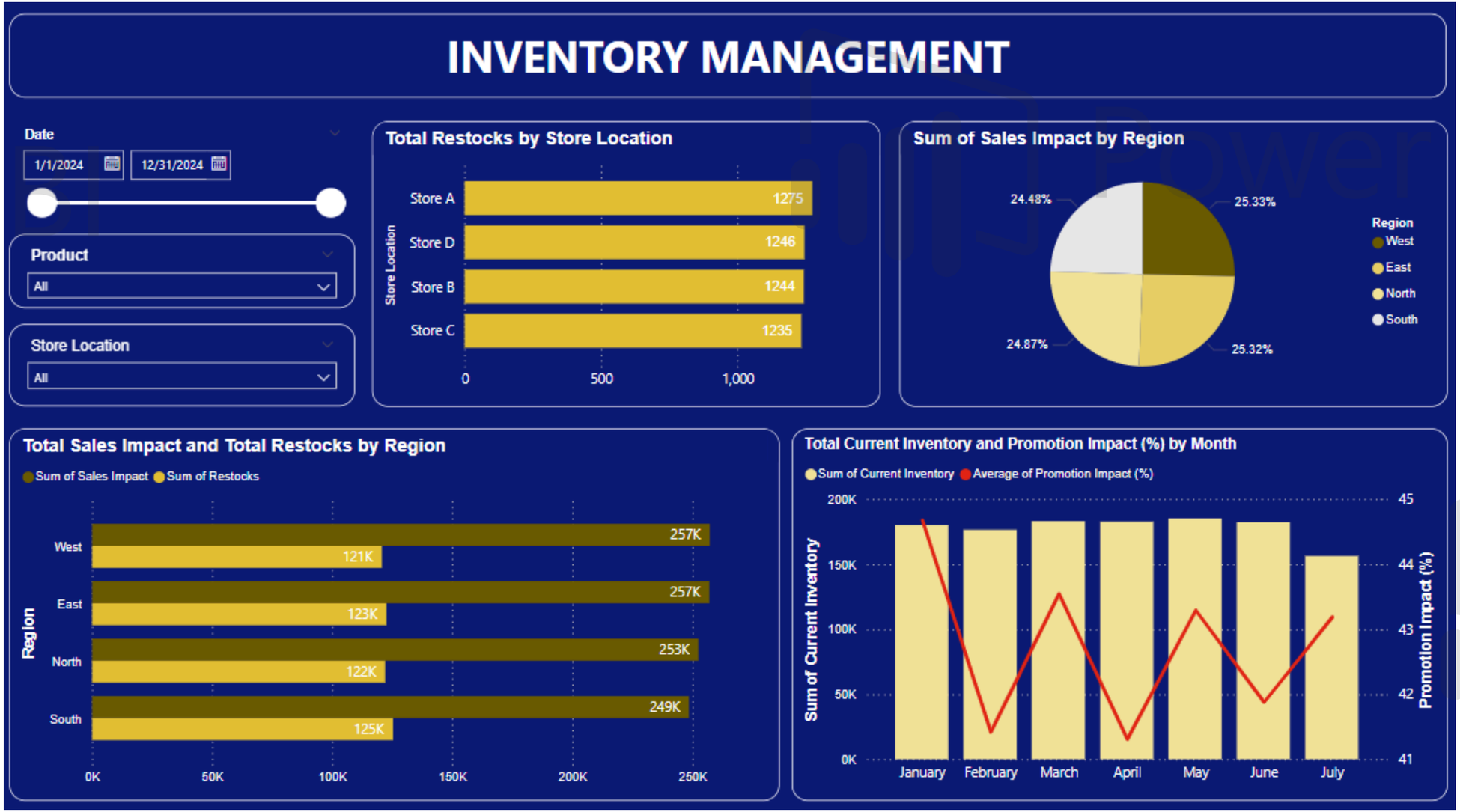
Region	avg_success_rate	avg_revenue
East	50.72575581394726	217.1633170731578
South	50.1211145997446	212.6937294762496
North	49.6694813614574	219.38787044535164
West	49.28306677017266	216.6703742037611

- East has the highest success rate at 50.7%
- Other regions:
  - South: 50.1%
  - North: 49.7
  - West: 49.2%

=> For promotion, East is the region that stores do the most successful with the highest rate though the total revenue contributing is the lowest.
However, West is recorded the most total, its success rate is lowest

Using Power Bi and MySQL Workbench for data mining and visualizing

#Overview display ***Inventory Status***



UNDERSTANDING

*Promotion Impact = [(Initial Inventory - Current Inventory)/Initial Inventory] * 100*

**The higher promotion impact, the more effective*

- **January has the highest promotion impact**
  -> most successful promotion campaign in the new year
- The products are distributed in a quite balanced manner across 4 regions

RESULT

Sales Performance

- Sales performance is good, the total revenue in each month is not much different

Promotion

- Almost cafes in 4 regions are working well and aligned with the initial objective of gaining more customers
- The average success rate is just above 50%

Inventory Management

- The stocks are running fast, however, in some months like February, April, June, the index of promotion impact is low, meaning that initial inventory is not much enough to serve when launching promotion.

RECOMMENDATION

- Gencafe should **add and make more drinks** to gain **more revenue** because currently, 4 products are working very well and attracting quite many customers

- Promotion campaigns are working effective, however, need **more actions to boost the success rate greater**

- Inventory management is **drastically fluctuated**, so that the team need to carefully follow in order to be ready when running out of stocks

LINKS

Github link (Dataset, PBi, Python, SQL)

https://github.com/holang07/Gencafe_sales-01-07-2024-

CONTACT

- Email: lamnh.forwork@gmail.com
- Phone: +84 386 079 237
- Linkedin: Hoang-Lam Nguyen