BIT2053 FUNDAMENTAL OF MODERN DATA

FINAL PROJECT - GROUP
(40%)

ALL SECTIONS

FACULTY OF INFORMATION TECHNOLOGY
CITY UNIVERSITY MALAYSIA
CYBERJAYA CAMPUS

PREPARED BY
SIR NAZMIRUL IZZAD BIN NASSIR

INSTRUCTIONS TO STUDENTS:

1. This **project brings 40%** from overall course assessment.
2. You are allowed to refer to the learning material to produce a quality assignment output within a specified time.
3. It is a **group assignment and full marks is 40%.**
4. Each group must consist of **4-5 members.**
5. Please **form your group** using the Google Docs Link given.
6. **This assignment submission can be done by a single representative of a group.**
7. The submission of the assignment must include your members'
    a. **FULL NAME,**
    b. **STUDENT ID,**
    c. **CLASS SECTION,**
    d. **PROGRAM,**
    e. **NRIC/PASSPORT NUMBER**.
8. Please upload the completed GitHub repository link via assignment link submission in the Google Classroom of BIT2053 Fundamental of Modern Data

Prepared by:

NO	Name	ID	NRIC	PROGRAM
1	Kenneth Chaw Yi Jie	202409010519	050903120587	BCSSE
2	Mohamad Hazrul Ekhwan Bin Moruin @ Abdul Hamid	202409010547	030703121333	BCSSE
3	Alexandro Elvin Adrian	202409010544	050806121027	BIT

Modern Data Exploration with BI Tools

1. **Objective:**

   Apply your understanding of modern data concepts by analyzing a real-world dataset using Business Intelligence (BI) tools. This project aims to simulate a data-driven decision-making process through a business scenario.

Table of Content

1.0 Introduction

This Final Project of Fundamental of Modern Data is to shows the understanding of modern data concepts by analyzing a real-world dataset from we found in Kaggle (Terron, 2024), using a Business Intelligence (BI) tools. Furthermore, this group consists of 3 membesr, which is Kenneth Chaw Yi Jie, Alexandro Elvin Adrian, and Mohadmad Hazrul Ekhwan. Kenneth as our project coordinator, Alexandro as our BI visualization specialist and last but not least, Ekhwan as our Data Analyst. All member contribute to completing this final project.

First of all, E-commerce dataset we choose from Kaggle is named as “Retail Salas Data Dashboard” . It rely on data-driven decision-making to dominate the overall market. As a construct, this project is to analyzes retail sales data chosen from Kaggle’s “Retail Sales Data Dashboard” dataset to evaluate category and regional performance. Other than that, by using the BI tools can help us clearly understand the impact of discounts on profitability, and further recommend strategies for revenue growth.

Below are the business questions shown to let us solve it eventually through the project:

Q1. Sales Trends : How do sales and quantity sold fluctuate over time (monthly/quarterly) to identify seasonal patterns?

Q2. Product Performance: Which product categories (Electronics, Clothing, Beauty) generate the highest revenue, and what are the best-selling products?

Q3. Customer Analysis: How does spending behavior differ by gender and age group? Which demographic is the most valuable?

2.0 Buiness Understanding

Business Objective: To leverage data-driven insights to optimize inventory management, reallocate marketing resources to high-performing categories, and tailor customer engagement strategies to the most valuable demographics.

Stakeholders: Marketing Director, Head of Sales, Inventory Manager.

Business Question Explanation:

Q1. The Inventory Manager should anticipate the peak sales months to avoid stock-outs and minimize the overstocking in slow periods.

Q2. The Head of Sales must identify the winning and under-performing categories in order to adjust procurement strategies and sales focus.

Q3. The Marketing Director requires a clear profile of the highest-spending demographic in order to improve the customer acquisition and retention ROI.

3.0 Data Understanding

In order to carry out a comprehensive business analysis, it is important to first develop a clear understanding of the dataset. The dataset selected for this project is the "Retail Sales Data Dashboard" obtained from Kaggle. It contains a total of **1,000 transaction records**, each representing an individual retail purchase. Every record includes important details about the transaction, such as the transaction ID, the date of purchase, customer information, product details, and the financial value of the purchase.

The attributes in this dataset cover a mixture of categorical and numerical data. The categorical fields include customer demographics such as **gender**, as well as product-related attributes such as **product category**. Meanwhile, the numerical attributes include **quantity purchased**, **price per unit**, and **total amount spent** in each transaction. The dataset also includes a **date** column, which provides a time dimension that is essential for trend and time-series analysis.

From an initial examination, I use Google Colab as the python tool to determine the dataset is relatively clean, with no missing values or duplicated records detected. However, the date column was originally stored as plain text, which required conversion into a proper datetime format before it could be used effectively in analysis. Overall, the dataset is well-suited for this project as it provides sufficient information to explore **sales performance, customer behaviour, and product demand patterns**, which are central to the business scenario chosen by our group.

```
[8] # Cell 1 - Upload CSV file into Colab
    from google.colab import files
    import pandas as pd
    import numpy as np

    # Upload file manually
    uploaded = files.upload()

    # After upload, the file will be in /content/
    file_path = list(uploaded.keys())[0]   # get uploaded filename
    print("File uploaded:", file_path)

    # Load dataset
    df = pd.read_csv(file_path)

    # Show first 5 rows
    df.head()
```

Choose Files | Copy of Ret... Project.csv

- Copy of Retail Sales Data Project.csv(text/csv) - 50605 bytes, last modified: 9/2/2025 - 100% done

```
Saving Copy of Retail Sales Data Project.csv to Copy of Retail Sales Data Project.csv
File uploaded: Copy of Retail Sales Data Project.csv
```

	Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount
0	1	11/24/2023	CUST001	Male	34	Beauty	3	50	150
1	2	2/27/2023	CUST002	Female	26	Clothing	2	500	1000
2	3	1/13/2023	CUST003	Male	50	Electronics	1	30	30
3	4	5/21/2023	CUST004	Male	37	Clothing	1	500	500
4	5	5/6/2023	CUST005	Male	30	Beauty	2	50	100

Figure 1: Code And Output Sample of Raw Retail Dataset Loaded in Python

```
# Cell 2 - Data Understanding (basic info)

# Shape of dataset
print("Shape of dataset:", df.shape)

# Column names
print("\nColumns:", df.columns.tolist())

# Info
print("\nDataset info:")
print(df.info())

# Missing values
print("\nMissing values per column:")
print(df.isnull().sum())

# Basic statistics
df.describe(include="all")
```

Figure 2: Code Dataset Structure and Data Types

```
Columns: ['Transaction ID', 'Date', 'Customer ID', 'Gender', 'Age', 'Product Category', 'Quantity', 'Price per Unit', 'Total Amount']

Dataset info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 9 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Transaction ID    1000 non-null   int64
 1   Date              1000 non-null   object
 2   Customer ID       1000 non-null   object
 3   Gender            1000 non-null   object
 4   Age               1000 non-null   int64
 5   Product Category  1000 non-null   object
 6   Quantity          1000 non-null   int64
 7   Price per Unit    1000 non-null   int64
 8   Total Amount      1000 non-null   int64
dtypes: int64(5), object(4)
memory usage: 70.4+ KB
None

Missing values per column:
Transaction ID      0
Date                0
Customer ID         0
Gender              0
Age                 0
Product Category    0
Quantity            0
Price per Unit      0
Total Amount        0
dtype: int64
```

	Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount
count	1000.000000	1000	1000	1000	1000.00000	1000	1000.000000	1000.000000	1000.000000
unique	NaN	345	1000	2	NaN	3	NaN	NaN	NaN
top	NaN	5/16/2023	CUST1000	Female	NaN	Clothing	NaN	NaN	NaN
freq	NaN	11	1	510	NaN	351	NaN	NaN	NaN
mean	500.500000	NaN	NaN	NaN	41.39200	NaN	2.514000	179.890000	456.000000
std	288.819436	NaN	NaN	NaN	13.68143	NaN	1.132734	189.681356	559.997632
min	1.000000	NaN	NaN	NaN	18.00000	NaN	1.000000	25.000000	25.000000
25%	250.750000	NaN	NaN	NaN	29.00000	NaN	1.000000	30.000000	60.000000
50%	500.500000	NaN	NaN	NaN	42.00000	NaN	3.000000	50.000000	135.000000
75%	750.250000	NaN	NaN	NaN	53.00000	NaN	4.000000	300.000000	900.000000
max	1000.000000	NaN	NaN	NaN	64.00000	NaN	4.000000	500.000000	2000.000000

Figure 3: Output Dataset Structure and Data Types

4.0 Data Preparation

Before analysis on going, raw data should be transformed into a processed and duly formatted form. It is inevitable that multiple pre-processing procedures would have to be applied to the dataset in order to make it consistent and ready for use by the analyst.

The first procedure was to reconcile the column names. With the raw dataset, the column names contained inconsistent description with spaces and capitalisation, which is problematic when applying programming languages such as Python. The column names were reformatted into lowercase, snake_case (for example, Product Category became product_category). Thereby, consistency was established and the probability of making errors during the coding was greatly reduced.

Next, the date column was converted into a datetime format which was important because that allowed us to derive different time-based features including year, month, day, quarter, month_year, etc. We created these features to enable more flexibility in our trend analyses and to allow the data to be aggregated at various time intervals.

After converting the date column into datetime format, the numeric columns (quantity, price_per_unit, total_amount) were examined to ensure they were correctly imported as numbers to allow aggregating, summations and statistical calculations on these variables. And would now also create a new column called sales_per_unit which is total_amount divided by the quantity sold, this derived metric is an important metric for us to evaluate pricing and profitability by unit.

Once these preprocessing steps were completed, the dataset was clean, consistent, and structured in a way that made it ready for meaningful analysis. The final prepared dataset was saved as clean_retail_data.csv, ensuring that it could be reused both for Python-based analysis and for integration into Business Intelligence (BI) tools such as Google Looker Studio.

```python
# Cell 3 - Data Preparation (cleaning)

# Rename columns to lowercase and snake_case
df.columns = df.columns.str.strip().str.lower().str.replace(" ", "_")

# Convert date column to datetime
df['date'] = pd.to_datetime(df['date'], errors='coerce')

# Ensure numeric columns are numeric
numeric_cols = ['quantity', 'price_per_unit', 'total_amount']
for col in numeric_cols:
    df[col] = pd.to_numeric(df[col], errors='coerce')

# Check again
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 9 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   transaction_id    1000 non-null   int64
 1   date              1000 non-null   datetime64[ns]
 2   customer_id       1000 non-null   object
 3   gender            1000 non-null   object
 4   age               1000 non-null   int64
 5   product_category  1000 non-null   object
 6   quantity          1000 non-null   int64
 7   price_per_unit    1000 non-null   int64
 8   total_amount      1000 non-null   int64
dtypes: datetime64[ns](1), int64(5), object(3)
memory usage: 70.4+ KB
```

Figure 4: Data Types After Cleaning and Conversion

```python
# Cell 4 - Feature Engineering

# Extract year, month, day, quarter
df['year'] = df['date'].dt.year
df['month'] = df['date'].dt.month
df['day'] = df['date'].dt.day
df['quarter'] = df['date'].dt.to_period('Q').astype(str)
df['month_year'] = df['date'].dt.to_period('M').astype(str)

# Sales per unit
df['sales_per_unit'] = df['total_amount'] / df['quantity']

df.head()
```

	transaction_id	date	customer_id	gender	age	product_category	quantity	price_per_unit	total_amount	year	month	day	quarter	month_year	sales_per_unit
0	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150	2023	11	24	2023Q4	2023-11	50.0
1	2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000	2023	2	27	2023Q1	2023-02	500.0
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30	2023	1	13	2023Q1	2023-01	30.0
3	4	2023-05-21	CUST004	Male	37	Clothing	1	500	500	2023	5	21	2023Q2	2023-05	500.0
4	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100	2023	5	6	2023Q2	2023-05	50.0

Figure 5: Dataset Sample After Feature Engineering

5.0 Analytics & Methods

The analytic phase of this project aimed to understand patterns, trends and insights that might come from the retail data. The approach we took was to focus on descriptive analytics which we interpreted as condensing the data into meaningful metrics that represent business performance. This analysis was undertaken using Python's pandas library and we put the results into separate summary tables to aid in the visual presentation when we subsequently use Business Intelligence tools.

First of all, to get an overall picture, I created an overall summary, including measured statistics, such as total sales, total quantity selling, unique customers, number of transactions and average sale for each transaction.

```
# Cell 5 - Overall Summary

overall_summary = {
    "total_rows": len(df),
    "total_sales": df['total_amount'].sum(),
    "total_quantity": df['quantity'].sum(),
    "num_customers": df['customer_id'].nunique(),
    "num_transactions": df['transaction_id'].nunique(),
    "avg_sales_per_transaction": df['total_amount'].mean()
}

overall_df = pd.DataFrame([overall_summary])
overall_df
```

	total_rows	total_sales	total_quantity	num_customers	num_transactions	avg_sales_per_transaction
0	1000	456000	2514	1000	1000	456.0

Figure 6: Overall Sales Summary

To enhance understanding of performance through time, a monthly summary was produced by aggregating transactions by the month_year field. The monthly summary demonstrated total monthly sales, total quantities sold, monthly active customers, and average sales per transaction. Hence, the summary is useful in identifying seasonal trends, sales growth or declines on a month over month basis requiring further scrutiny for the business.

```
# Cell 6 - Monthly Summary

monthly_summary = df.groupby('month_year').agg(
    total_sales=('total_amount', 'sum'),
    total_quantity=('quantity', 'sum'),
    avg_sales=('total_amount', 'mean'),
    num_customers=('customer_id', 'nunique')
).reset_index()

monthly_summary.head()
```

	month_year	total_sales	total_quantity	avg_sales	num_customers
0	2023-01	35450	195	466.447368	76
1	2023-02	44060	214	518.352941	85
2	2023-03	28990	194	397.123288	73
3	2023-04	33870	214	393.837209	86
4	2023-05	53150	259	506.190476	105

Figure 7: Monthly Sales and Customer Trends

At the product level we were able to generate a summary by category. This summary indicated which categories of products contributed the most to sales and which ones were performing at a lower level. Evaluating total sales, total quantities sold, and average unit price by product category allows a business to identify strong performing categories to evaluate if they can add to the sales effort in those categories. We also identified which product categories contribute to the bottom line and which one are under-performing and required promotions or some other intervention in order to drive sales.

```
# Cell 7 - Category Summary

category_summary = df.groupby('product_category').agg(
    total_sales=('total_amount', 'sum'),
    total_quantity=('quantity', 'sum'),
    avg_price=('price_per_unit', 'mean'),
    num_customers=('customer_id', 'nunique')
).reset_index()

category_summary.head()
```

	product_category	total_sales	total_quantity	avg_price	num_customers
0	Beauty	143515	771	184.055375	307
1	Clothing	155580	894	174.287749	351
2	Electronics	156905	849	181.900585	342

Figure 8: Product Category Performance Summary

A summary for customers was produced on the customer side. This was done by summarizing the data based on customer id, to be able to measure the total spend for each customer, the total quantity purchased, average order value, and number of transactions. This type of customer analysis is important for targeting high-value customers, getting insight into their purchase behaviour and ultimately developing strategies for customer loyalty.

```
# Cell 8 - Customer Summary

customer_summary = df.groupby('customer_id').agg(
    total_sales=('total_amount', 'sum'),
    total_quantity=('quantity', 'sum'),
    avg_order_value=('total_amount', 'mean'),
    num_transactions=('transaction_id', 'nunique')
).reset_index()

customer_summary.head()
```

	customer_id	total_sales	total_quantity	avg_order_value	num_transactions
0	CUST001	150	3	150.0	1
1	CUST002	1000	2	1000.0	1
2	CUST003	30	1	30.0	1
3	CUST004	500	1	500.0	1
4	CUST005	100	2	100.0	1

Figure 9: Customer Spending and Transaction Summary

Finally, a daily summary was produced, providing transaction level information on a daily basis. This allowed investigation into day-to-day variations in sales, short term peaks in demand, and identifying potential anomalies.

```
# Cell 10 - Daily Summary (for time-series charts)

daily_summary = df.groupby('date').agg(\

    total_sales=('total_amount', 'sum'),
    total_quantity=('quantity', 'sum'),
    num_customers=('customer_id', 'nunique'),
    num_transactions=('transaction_id', 'nunique')
).reset_index()

# Save daily summary
daily_summary.to_csv("summary_daily.csv", index=False)

daily_summary.head()
```

	date	total_sales	total_quantity	num_customers	num_transactions
0	2023-01-01	3600	10	3	3
1	2023-01-02	1765	10	4	4
2	2023-01-03	600	2	1	1
3	2023-01-04	1240	8	3	3
4	2023-01-05	1100	5	3	3

Figure 10: Daily Sales Performance Summary

These summary outputs were saved into CSV files and will later be imported into Google Looker Studio, where they will be transformed into interactive dashboards with charts, filters, and KPI indicators. Together, the analytics and visualization will enable decision-makers to explore the data more intuitively and support data-driven business strategies.

6.0 BI Dashboards

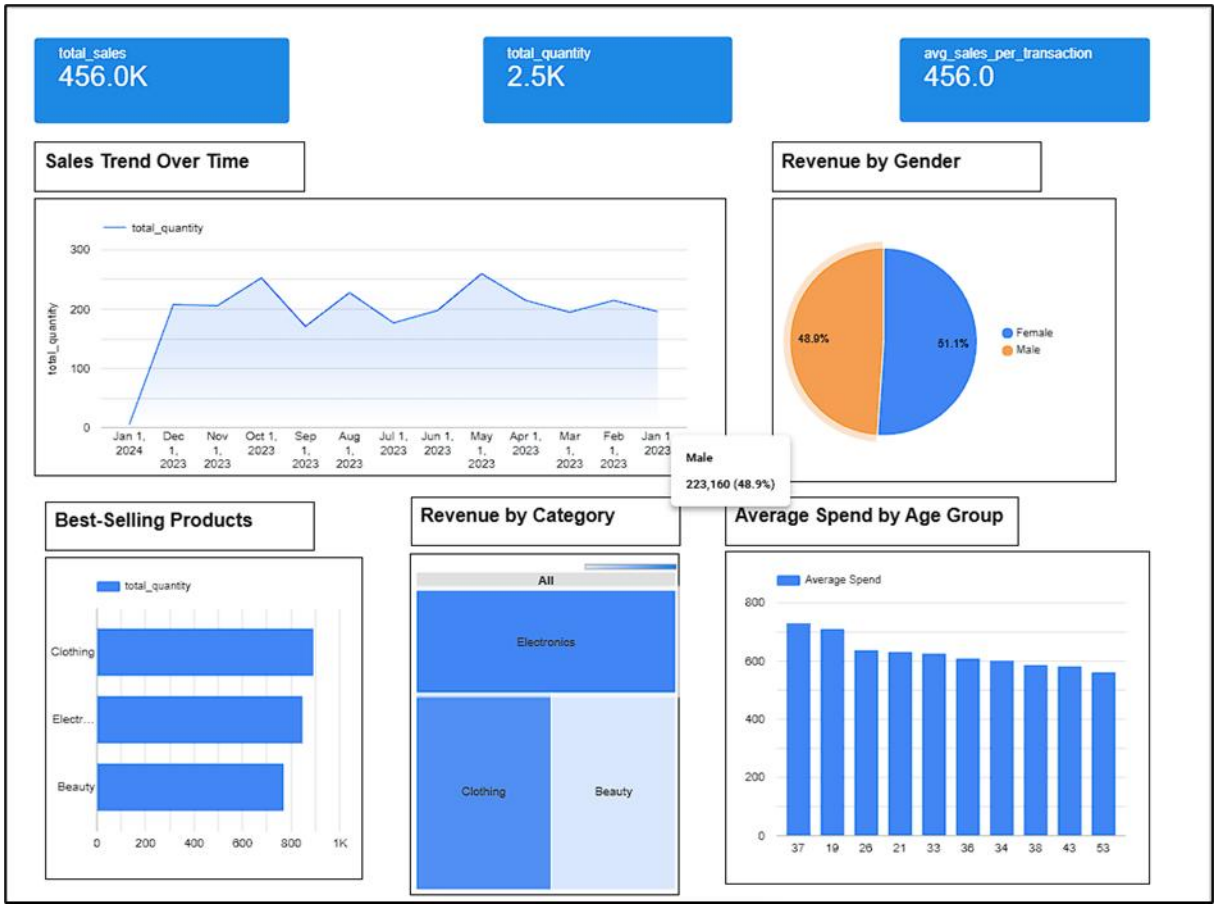

Figure 11: Retail Sales Data Dashboard

Key Metrics:

- Total sales of 456.0K, 2.5K items sold, with average transaction value of 456.0 showing solid business performance.

Sales Trend Over Time:

- Line chart reveals fluctuating sales quantities from 2023-2024, with notable peaks in May and September 2023.

Revenue by Gender:

- Pie chart shows nearly balanced gender distribution - females contribute 51.1% and males 48.9% of total revenue.

Best-Selling Products:

- Bar chart displays Clothing leading in quantity sold, followed by Electronics and Beauty products in descending order.

Revenue by Category:

- Treemap visualization indicates Electronics dominates revenue share despite lower sales quantities compared to other product categories.

Average Spend by Age Group:

- Bar chart shows spending patterns across ages 19-53, with peak spending occurring around age 37.

7.0 Findings & Insights

Strong Transaction Economics:

- High average transaction value of 456.0 suggests premium positioning and effective upselling strategies driving profitable customer interactions.

Balanced Customer Demographics:

- Equal gender split and broad age appeal from 19-53 indicates successful market penetration across diverse customer segments.

Product Mix Optimization Needed:

- Electronics generate higher revenue per unit than Clothing despite lower volumes, suggesting pricing and margin optimization opportunities.

Seasonal Volatility Management:

- Significant sales fluctuations indicate need for improved demand forecasting and inventory planning to smooth operational challenges.

Target Demographics Identified:

- Age 37 represents peak spending customers, suggesting focused marketing campaigns could maximize revenue from this high-value segment.

Category Growth Potential:

- Beauty products show underperformance in both quantity and revenue, presenting clear expansion opportunities for business growth.

8.0 Recommendations & Impact

As a construct, the comprehensive analysis of the sales trends, product performance, and customer demographic, we propose the following data-driven strategies that able to increase the overall profitability and also market share.

Retail store can expand the electronics category assortment and feature it. This recommendation can allocate a larger portion of the marketing budget to promote the high-margin electronic products through the targeted online ads and homepage featuring on the company website.

Other than that, retial store can try to develop a personalized marketing campaigns for the 34-43 age demographic by creating tailored email marketing and loyalty programs that able to bundle the population of Electronics and Beauty products to appeal this high-value segment's preferences. As impact, this able to maximize revenue generation from the most profitable segment.

Retail store can also initiate a "Pre-Peak" inventory review in October to secure the stock for Electronics and other best-selling products, preventing stock-outs during critical high-demand periods. These recommendations can be able to capture early holiday shoppers and ensure smooth out demand, in order to reduce the logistical pressure. These measures will minimize the lost sale from stock-outs during the peak season, reduce the cost associated with emergency shipping, and potentially improve peak season revenue.

Last but not least, retail store can do several investigations about the underperformance in the Beauty category. This analysis should include a competitor pricing review, assessment of product quality/perception. In addition, we can create a product bundle that pair Beauty items with best-selling Electronics. This strategy can increase sales for the Beauty category by leveraging the strength of Electronics lineup. This strategy is designed to stimulate growth within the Beauty category in order to improve overall brand appeal.

9.0 Conclusion

As a result, this project successfully demonstrates the BI analytics to transform retail data into actionable strategy. Analysis revealed high transaction values and a nicely balanced customer base, with Electronics contributing high-value revenue versus Clothing's volume. The Beauty category was identified as a priority growth opportunity. Demographically, the 37-year-old segment was most valuable, though the brand enjoys strong multi-generational appeal.

These results have directly informed our recommendations: capitalizing on high-margin Electronics, promotional targeting of peak-spending cohorts, inventory optimization for seasonal purchases, and reinventing the Beauty category through strategic bundling. In total, this project presents a brief data-driven roadmap to enhanced profitability and competitive advantage, showing that modern data concepts are extremely useful for strategic decision-making in the retail sector.