Rakamin Academy X KALBE Nutritionals

Final Project

Kalbe Nutritionals Data Scientist Project Based Internship

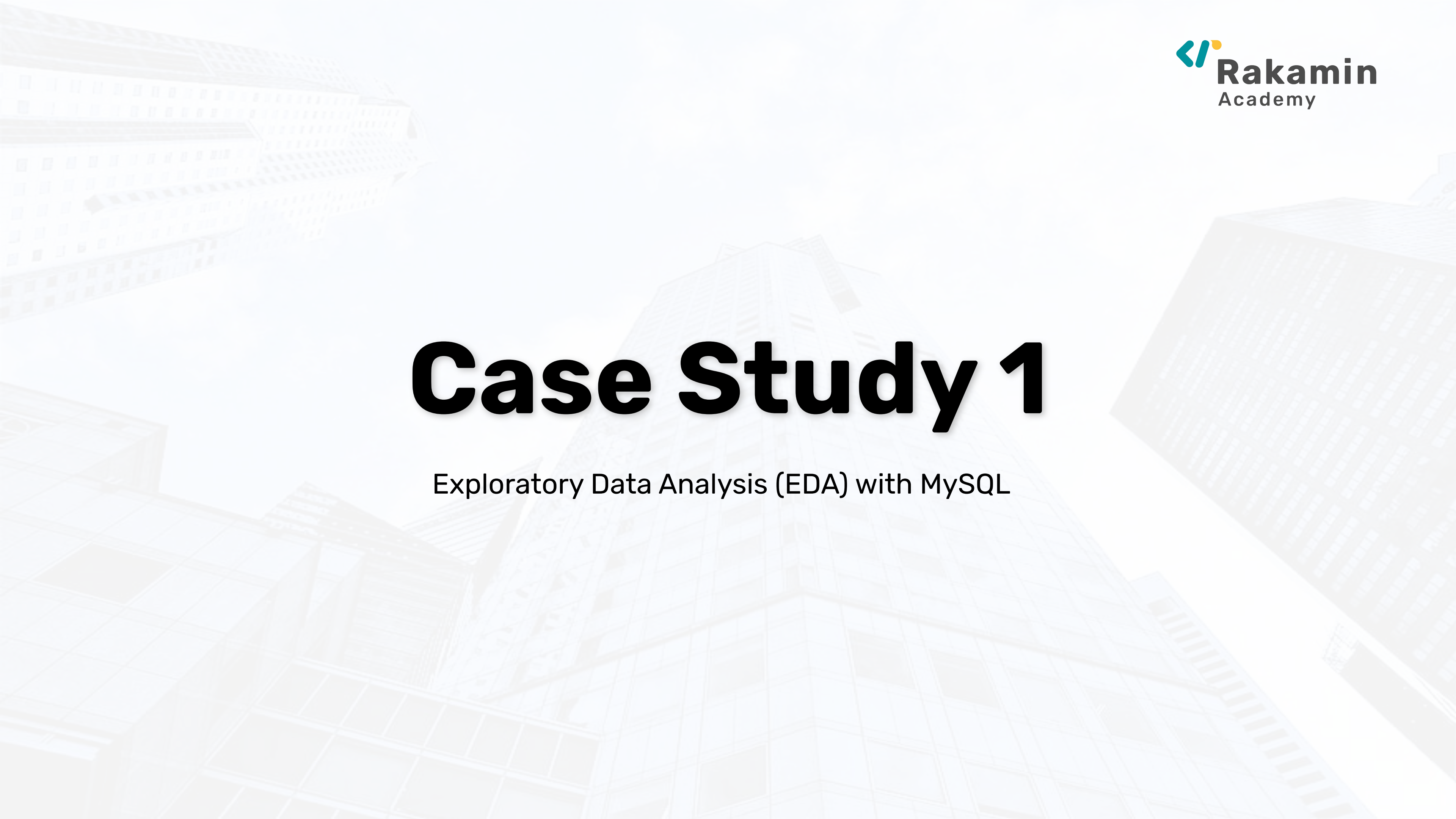
Presented by
Felicia Dina Widyasari

Case Study Description

In this project, as a Data Scientist at Kalbe Nutritionals, I am responsible for addressing two key objectives from the inventory team and marketing team:

1. Daily Sales Prediction
   - Assist the inventory team in forecasting daily sales quantity for all Kalbe products.
   - Goal: Ensure optimal stock availability to meet daily demand efficiently.
2. Customer Segmentation
   - Group customers based on various criteria for the marketing team.
   - Goal: Enhance promotional effectiveness through personalized marketing strategies.

This project leverages Python, Tableau, MySQL for data analysis, predictive modeling, and visualization

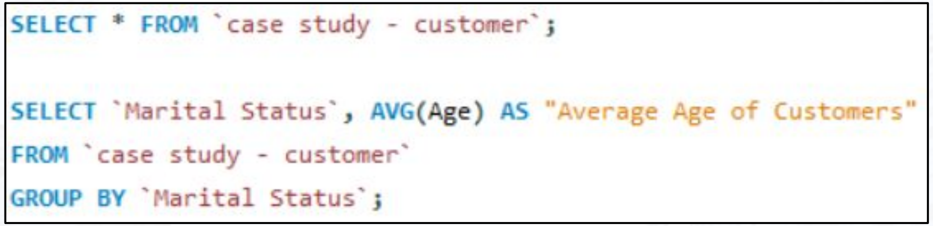


Rakamin Academy

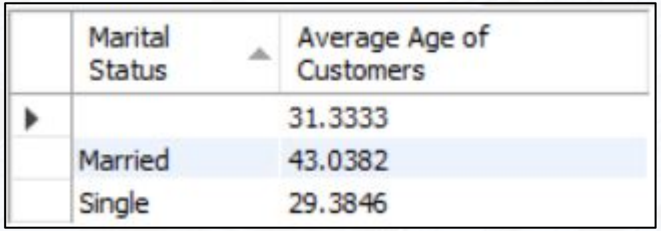
Case Study 1

Exploratory Data Analysis (EDA) with MySQL

Query 1:
Average Age of Customers based on Marital Status

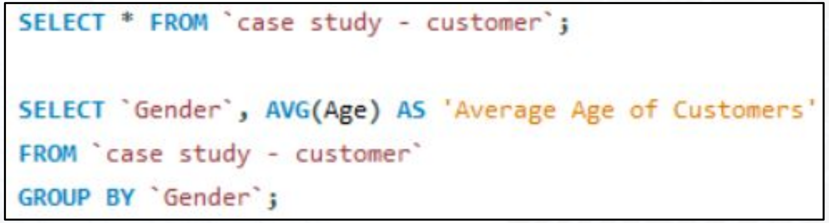


```
SELECT * FROM `case study - customer`;

SELECT `Marital Status`, AVG(Age) AS "Average Age of Customers"
FROM `case study - customer`
GROUP BY `Marital Status`;
```

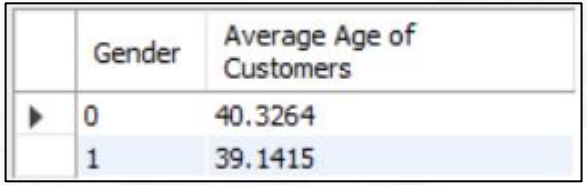


Marital Status	Average Age of Customers
	31.3333
Married	43.0382
Single	29.3846

Query 2:
Average Age of Customers based on Gender

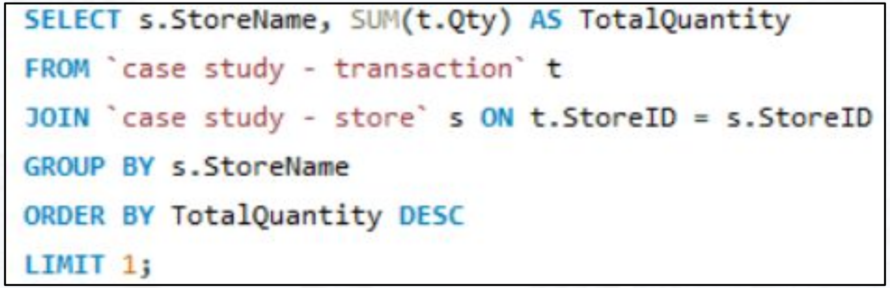


```
SELECT * FROM `case study - customer`;

SELECT `Gender`, AVG(Age) AS 'Average Age of Customers'
FROM `case study - customer`
GROUP BY `Gender`;
```

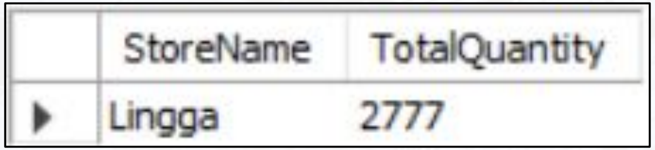


Gender	Average Age of Customers
0	40.3264
1	39.1415

Query 3:
The store with the highest total quantity sold

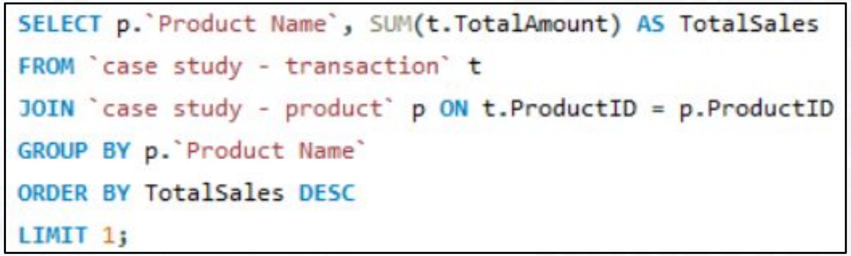


```
SELECT s.StoreName, SUM(t.Qty) AS TotalQuantity
FROM `case study - transaction` t
JOIN `case study - store` s ON t.StoreID = s.StoreID
GROUP BY s.StoreName
ORDER BY TotalQuantity DESC
LIMIT 1;
```

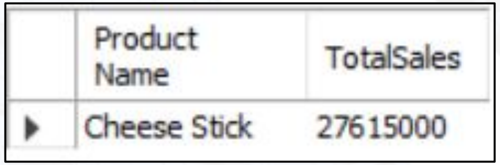


	StoreName	TotalQuantity
▶	Lingga	2777

Query 4:
The best-selling product based on the highest total amount



```
SELECT p.`Product Name`, SUM(t.TotalAmount) AS TotalSales
FROM `case study - transaction` t
JOIN `case study - product` p ON t.ProductID = p.ProductID
GROUP BY p.`Product Name`
ORDER BY TotalSales DESC
LIMIT 1;
```

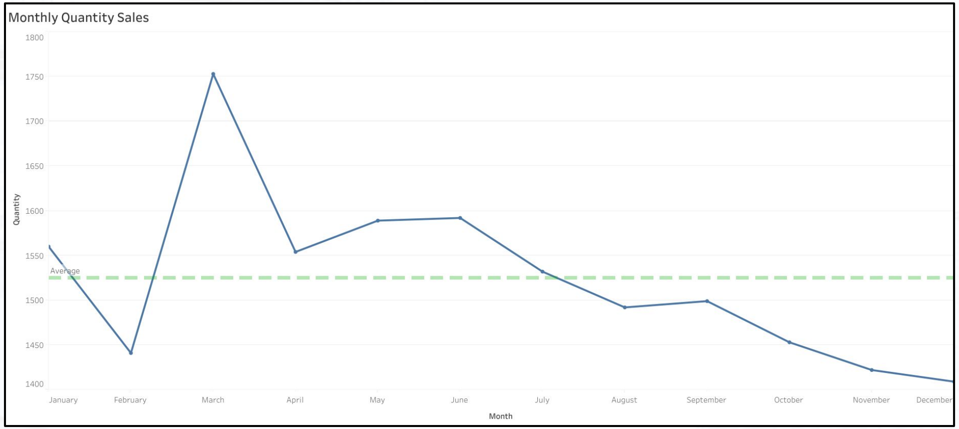


	Product Name	TotalSales
▶	Cheese Stick	27615000

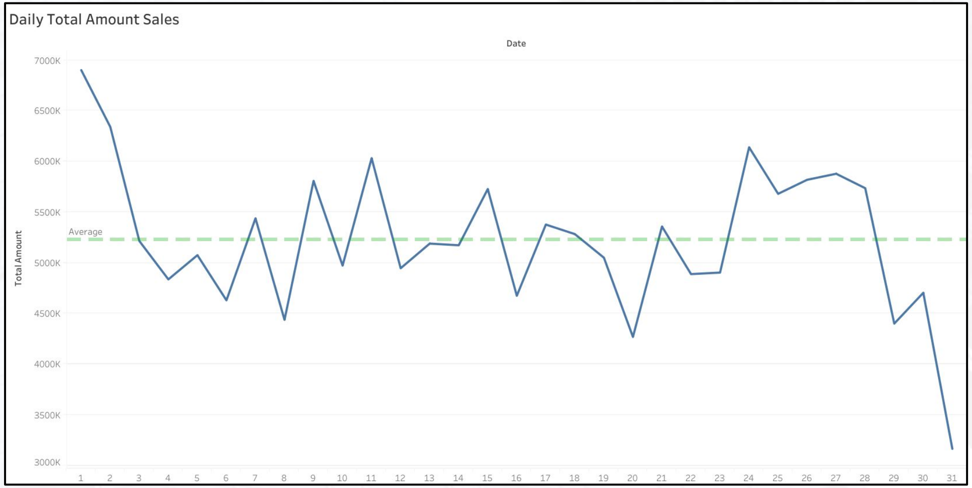
Case Study 2

Data Visualization and Dashboard creation using Tableau

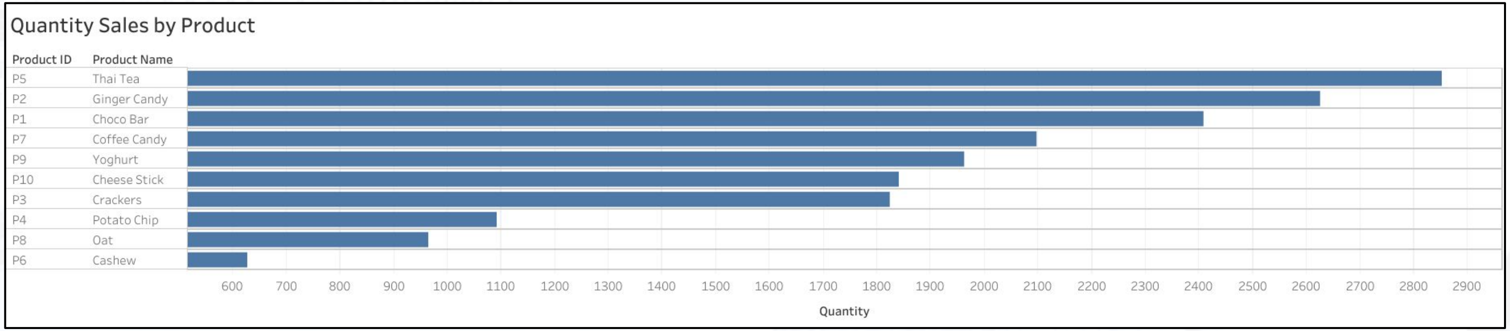
1. Monthly Quantity Sales



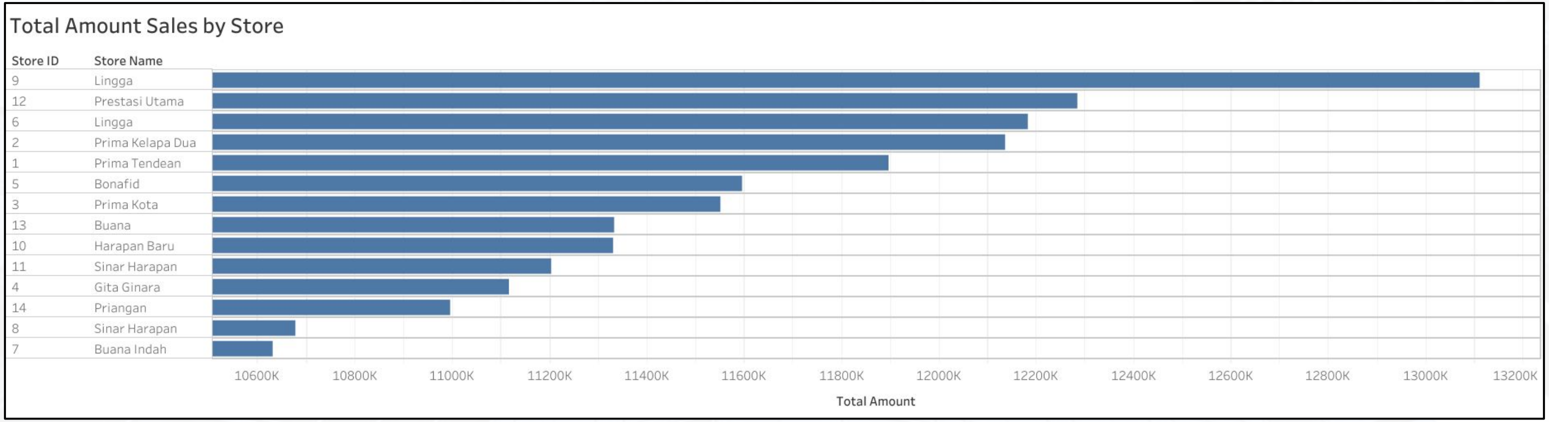
2. Daily Total Amount Sales



3. Total Sales by Product



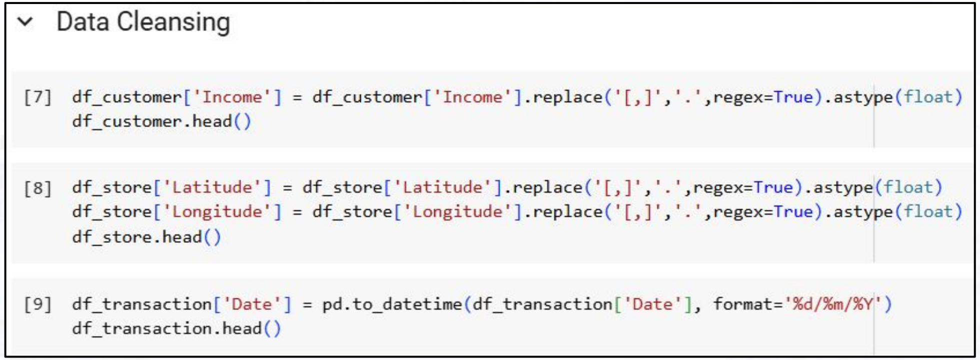
4. Total Amount Sales by Store



Case Study 3

Machine Learning Regression (Time Series) Using ARIMA

The purpose of developing this machine learning model is to predict the daily total quantity of products sold.



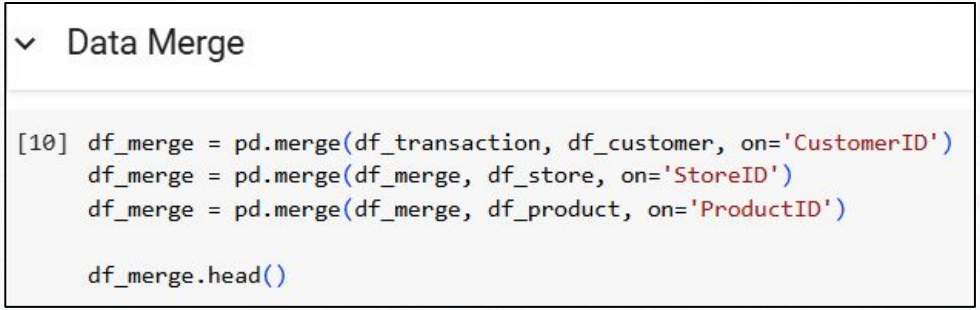
Data Cleansing

```
[7]  df_customer['Income'] = df_customer['Income'].replace('[,]','.',regex=True).astype(float)
     df_customer.head()
```

```
[8]  df_store['Latitude'] = df_store['Latitude'].replace('[,]','.',regex=True).astype(float)
     df_store['Longitude'] = df_store['Longitude'].replace('[,]','.',regex=True).astype(float)
     df_store.head()
```

```
[9]  df_transaction['Date'] = pd.to_datetime(df_transaction['Date'], format='%d/%m/%Y')
     df_transaction.head()
```

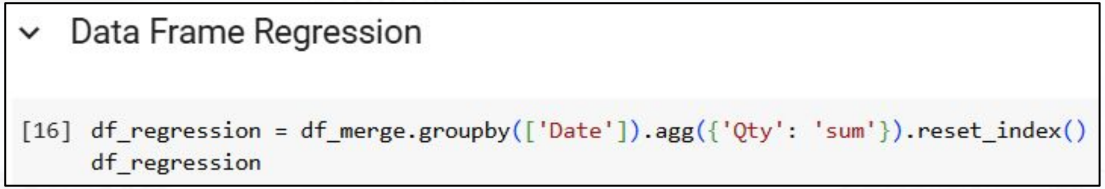
Data cleansing first, then adjust data types accordingly.



Data Merge

```
[10] df_merge = pd.merge(df_transaction, df_customer, on='CustomerID')
     df_merge = pd.merge(df_merge, df_store, on='StoreID')
     df_merge = pd.merge(df_merge, df_product, on='ProductID')

     df_merge.head()
```

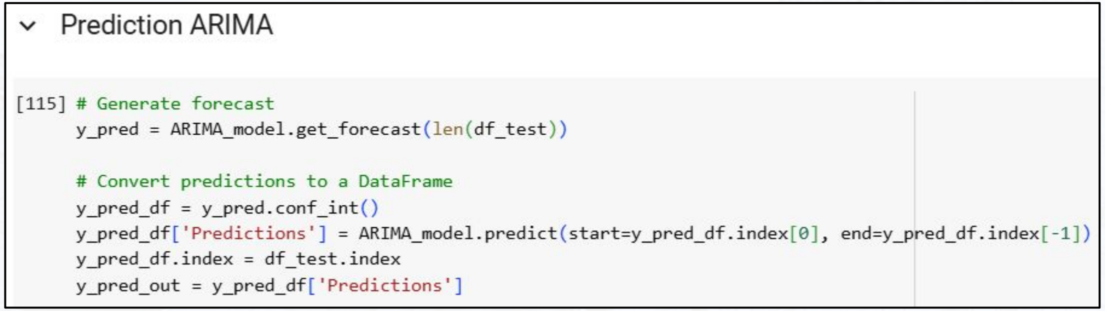
Data merging to combine all datasets



Data Frame Regression

```
[16] df_regression = df_merge.groupby(['Date']).agg({'Qty': 'sum'}).reset_index()
     df_regression
```

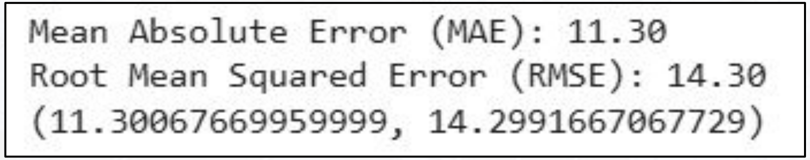
- Create a new dataset for regression by grouping data by date and aggregating the quantity using sum.
- The final dataset will have approximately 365 rows.



Prediction ARIMA

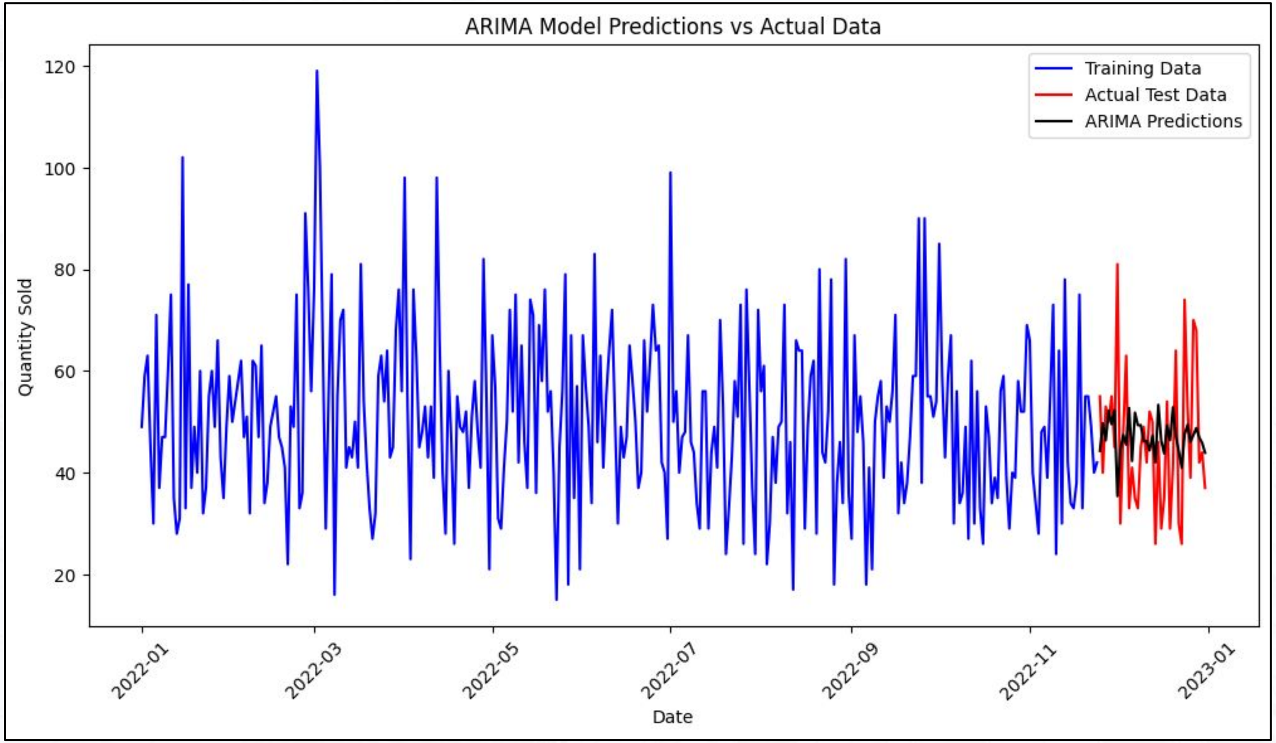
```
[115] # Generate forecast
      y_pred = ARIMA_model.get_forecast(len(df_test))

      # Convert predictions to a DataFrame
      y_pred_df = y_pred.conf_int()
      y_pred_df['Predictions'] = ARIMA_model.predict(start=y_pred_df.index[0], end=y_pred_df.index[-1])
      y_pred_df.index = df_test.index
      y_pred_out = y_pred_df['Predictions']
```

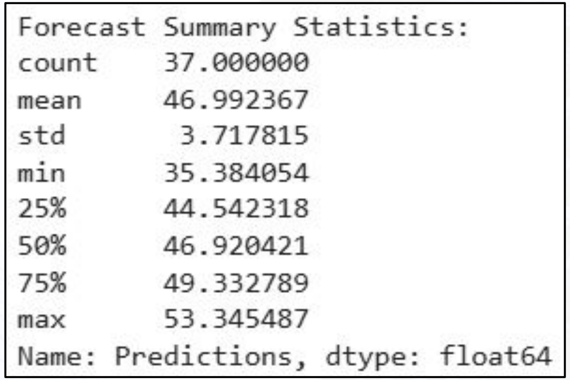
The data is split into training and testing sets based on time, then the ARIMA model is trained using the training data to capture historical patterns, and finally, it is used to predict values on the testing data.



```
Mean Absolute Error (MAE): 11.30
Root Mean Squared Error (RMSE): 14.30
(11.30067669959999, 14.2991667067729)
```

The ARIMA model's performance is evaluated using MAE (11.30) and RMSE (14.30), indicating its accuracy in predicting the test data.





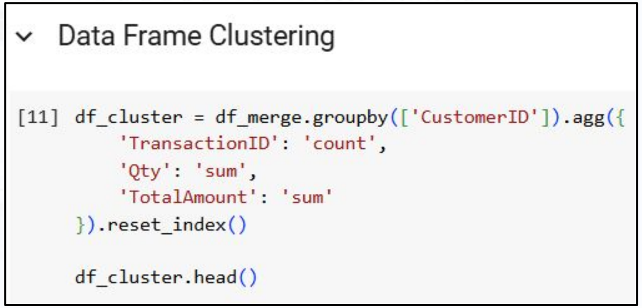
```
Forecast Summary Statistics:
count     37.000000
mean      46.992367
std        3.717815
min       35.384054
25%       44.542318
50%       46.920421
75%       49.332789
max       53.345487
Name: Predictions, dtype: float64
```

The ARIMA model predicts that for the next month (January 1-31, 2023), the inventory team should prepare approximately 47 stocks/day.

Case Study 4

Machine Learning Clustering Using K-Means

The purpose of building this machine learning model is to create clusters of similar customers.

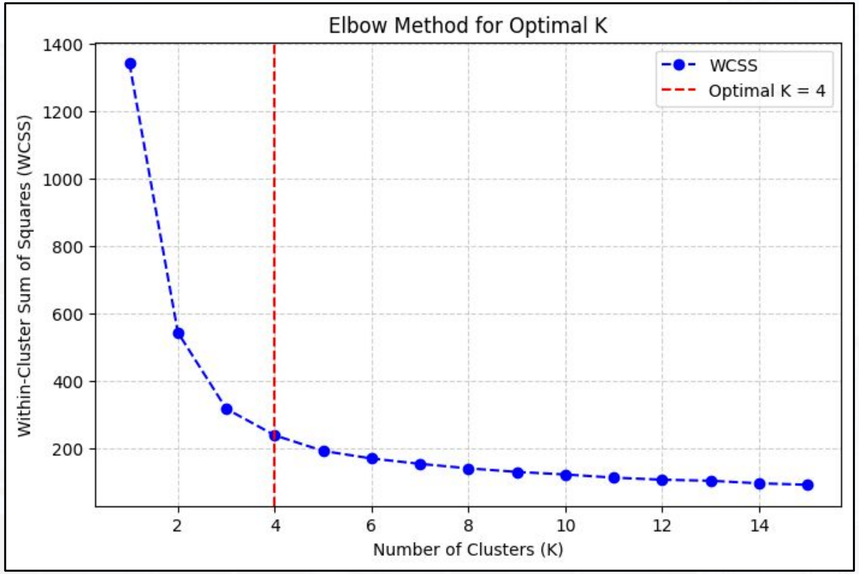


Data Frame Clustering

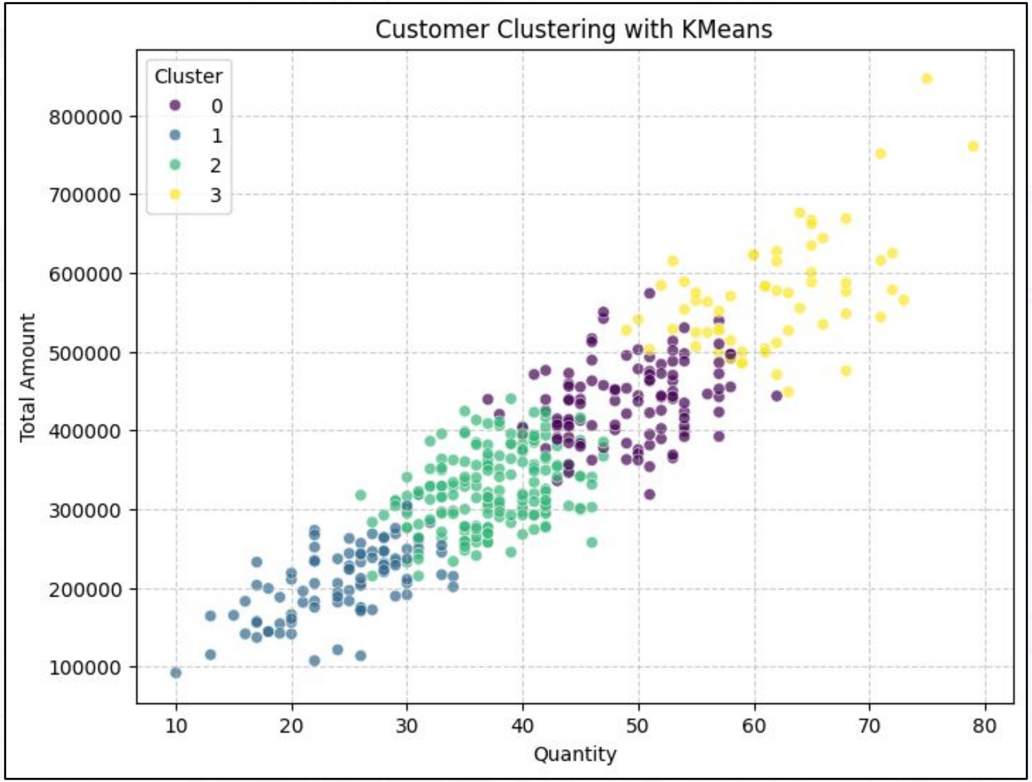
```
[11] df_cluster = df_merge.groupby(['CustomerID']).agg({
         'TransactionID': 'count',
         'Qty': 'sum',
         'TotalAmount': 'sum'
     }).reset_index()

     df_cluster.head()
```

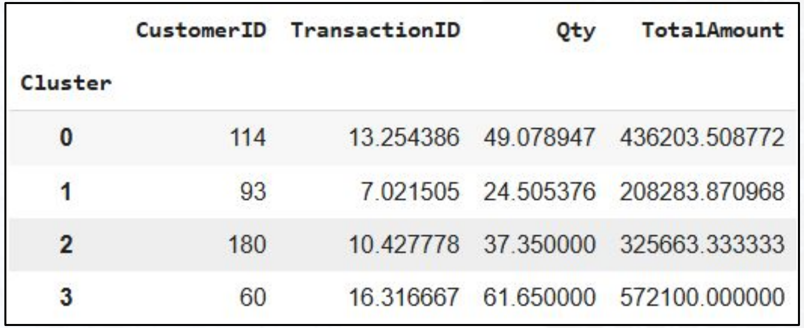
Create a new dataset for clustering by grouping by CustomerID, then aggregate the following:

- TransactionID → Count
- Qty → Sum
- TotalAmount → Sum



The Elbow Method is used to determine the optimal number of clusters (K), and based on the graph, the best value for K is identified as 4.





	CustomerID	TransactionID	Qty	TotalAmount
Cluster				
0	114	13.254386	49.078947	436203.508772
1	93	7.021505	24.505376	208283.870968
2	180	10.427778	37.350000	325663.333333
3	60	16.316667	61.650000	572100.000000

Customer Clustering Insights

🔹 **Cluster 3 – High Value Customers**

- Highest transactions (16.3 times) & largest spending (Rp572,100)
- Potentially loyal customers or wholesale buyers

🔹 **Cluster 0 – Frequent & High Spenders**

- Large customer base (114 people), high transactions (13.25 times)
- Frequently make large purchases (Rp436,203)

🔹 **Cluster 2 – Moderate Customers**

- Largest customer group (180), moderate transactions & spending
- Potential target for loyalty programs

🔹 **Cluster 1 – Low Spenders**

- Lowest transactions & spending (Rp208,283)
- Possibly new or less active customers

Recommended Actions

✔ **Cluster 3** → Offer exclusive deals to retain high-value customers.

✔ **Clusters 0 & 2** → Provide discounts or membership programs to boost loyalty.

✔ **Cluster 1** → Launch special promotions or retargeting campaigns to increase engagement.

Rakamin Academy

Link GitHub

https://github.com/feliciadina/FinalProject-Kalbe-Rakamin

Thank You

Rakamin Academy X KALBE Nutritionals