Data Analyst

Portofolio.

By. Adam Maulana

Hello.

Adam Maulana

Adam Maulana

About Me.

Hi! I'm Adam Maulana
an Data enthusiast with a strong passion for analysis, digital content, and visualization. Experienced in data management and storytelling to support decisions. Adaptable, communicative, and quick to learn new tools.

Education & Experience

Feb - 2023

Graduate From

Universitas Nasional B.Sc. in Agrotechnology

Oct 2021 – Nov 2022

Join

PT. Harcoselaras Sentosajaya as Administrative Staff

Oct 2023 – Des 2024

FREELANCE

Handled

Content creation and management for Instagram platform

Jun 2025

Finished

Full Stack Data Analyst 6 Months Program Batch 26

Certificate click here

Data Analyst.

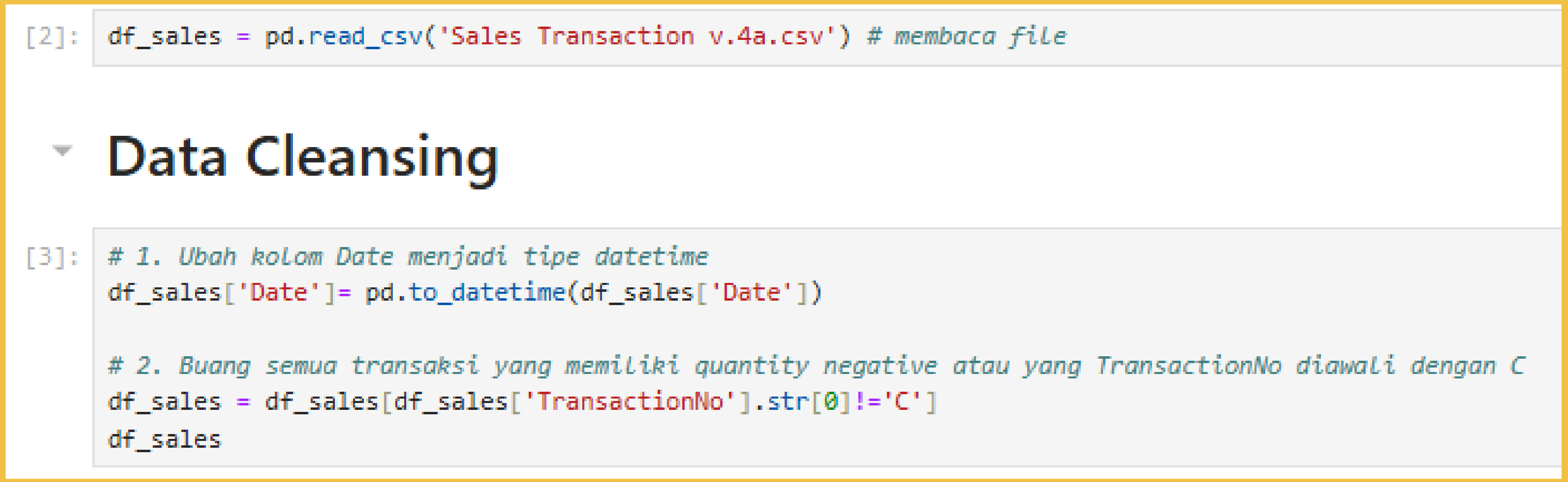
Data Cleansing - Data Segmentation - Retention Cohort Analysis - Time Series Analysis - Basket Size Analysis

Adam Maulana

Click Here To See Full Coding

Data Cleansing

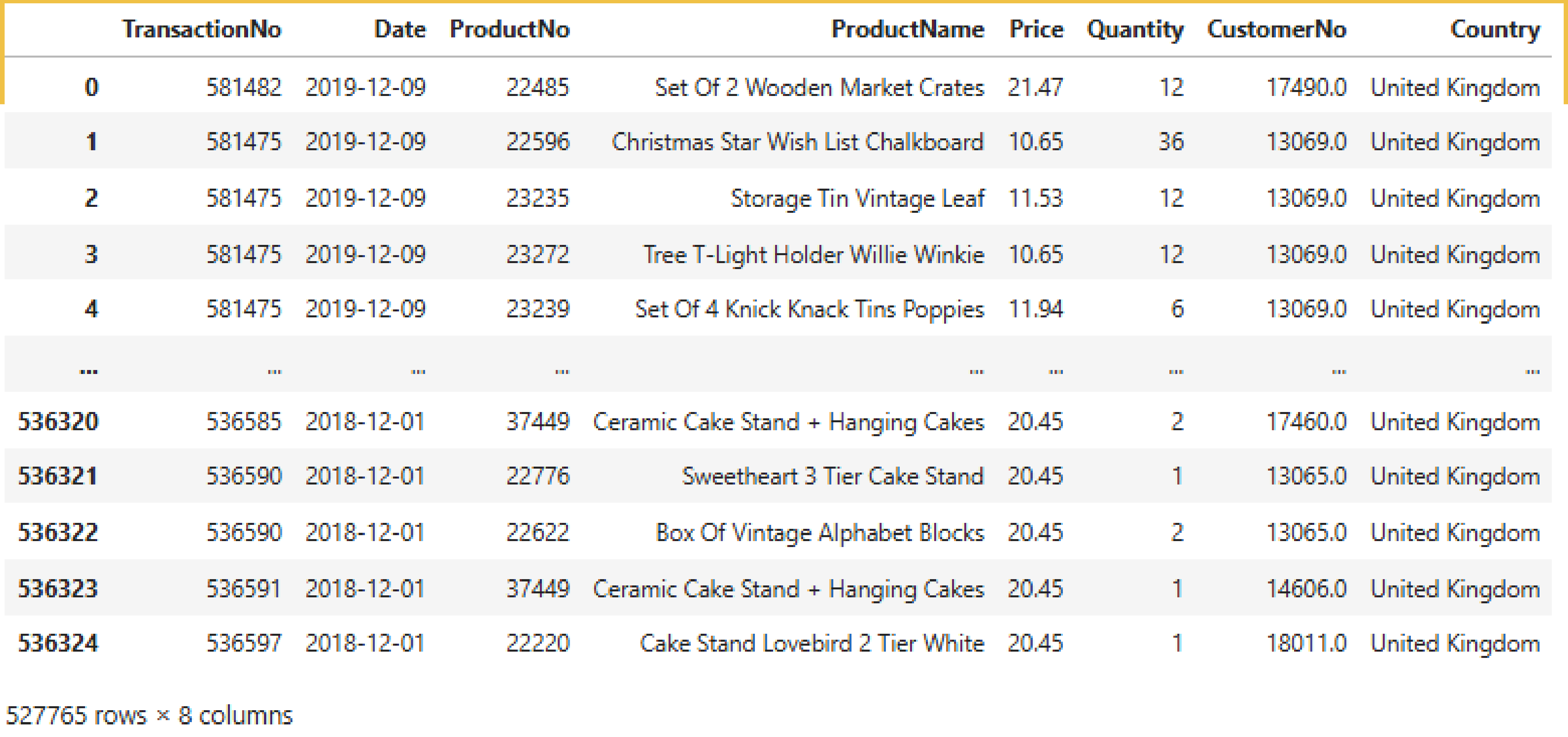
This dataset was utilized during my Intermediate Data Analyst certification program at Growia, where I applied various analytical techniques as part of the final project.



```
[2]: df_sales = pd.read_csv('Sales Transaction v.4a.csv') # membaca file
```

Data Cleansing

```
[3]: # 1. Ubah kolom Date menjadi tipe datetime
     df_sales['Date']= pd.to_datetime(df_sales['Date'])

     # 2. Buang semua transaksi yang memiliki quantity negative atau yang TransactionNo diawali dengan C
     df_sales = df_sales[df_sales['TransactionNo'].str[0]!='C']
     df_sales
```



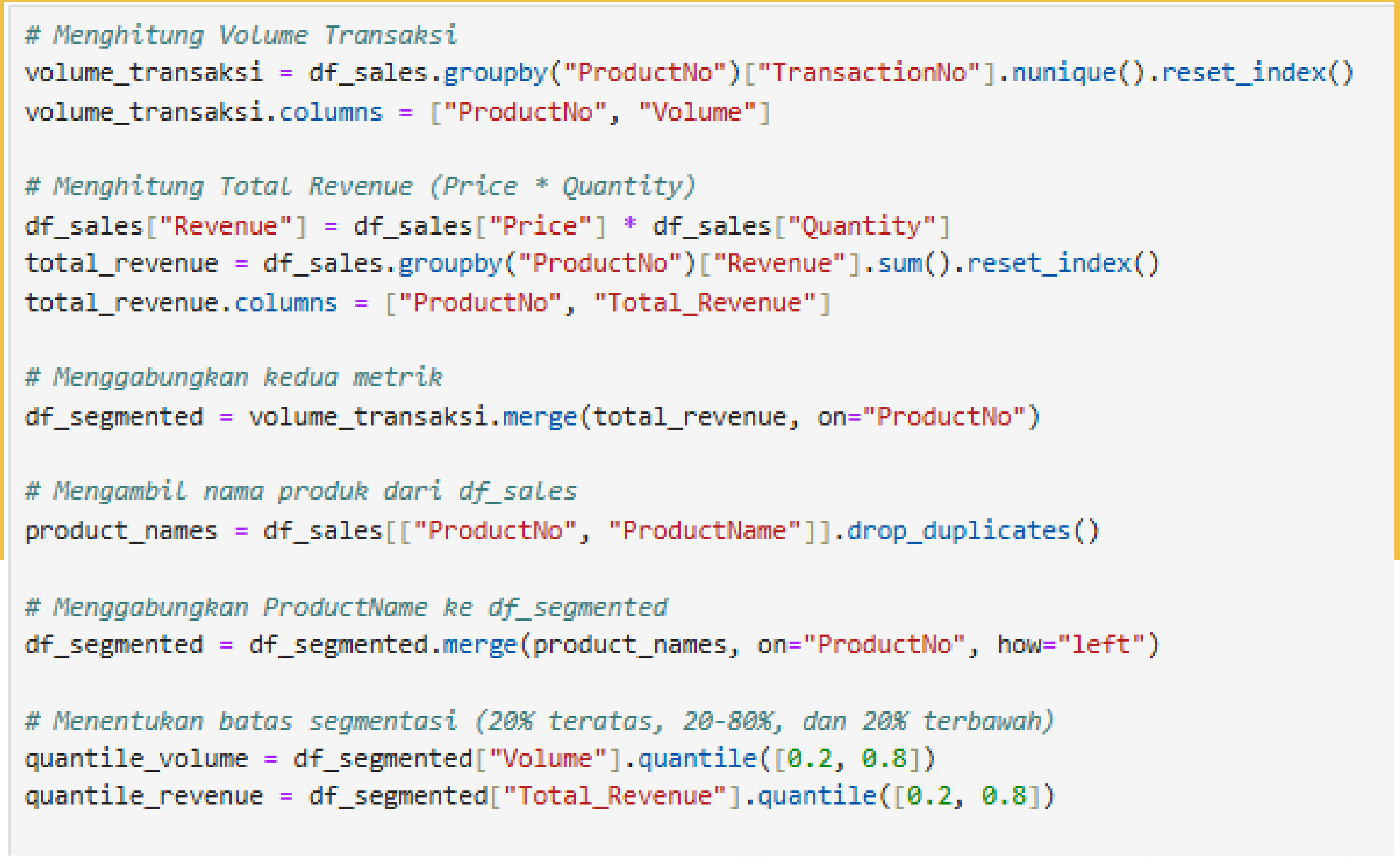
	TransactionNo	Date	ProductNo	ProductName	Price	Quantity	CustomerNo	Country
0	581482	2019-12-09	22485	Set Of 2 Wooden Market Crates	21.47	12	17490.0	United Kingdom
1	581475	2019-12-09	22596	Christmas Star Wish List Chalkboard	10.65	36	13069.0	United Kingdom
2	581475	2019-12-09	23235	Storage Tin Vintage Leaf	11.53	12	13069.0	United Kingdom
3	581475	2019-12-09	23272	Tree T-Light Holder Willie Winkie	10.65	12	13069.0	United Kingdom
4	581475	2019-12-09	23239	Set Of 4 Knick Knack Tins Poppies	11.94	6	13069.0	United Kingdom
...	...	...	...	...	...	...	...	...
536320	536585	2018-12-01	37449	Ceramic Cake Stand + Hanging Cakes	20.45	2	17460.0	United Kingdom
536321	536590	2018-12-01	22776	Sweetheart 3 Tier Cake Stand	20.45	1	13065.0	United Kingdom
536322	536590	2018-12-01	22622	Box Of Vintage Alphabet Blocks	20.45	2	13065.0	United Kingdom
536323	536591	2018-12-01	37449	Ceramic Cake Stand + Hanging Cakes	20.45	1	14606.0	United Kingdom
536324	536597	2018-12-01	22220	Cake Stand Lovebird 2 Tier White	20.45	1	18011.0	United Kingdom

527765 rows × 8 columns

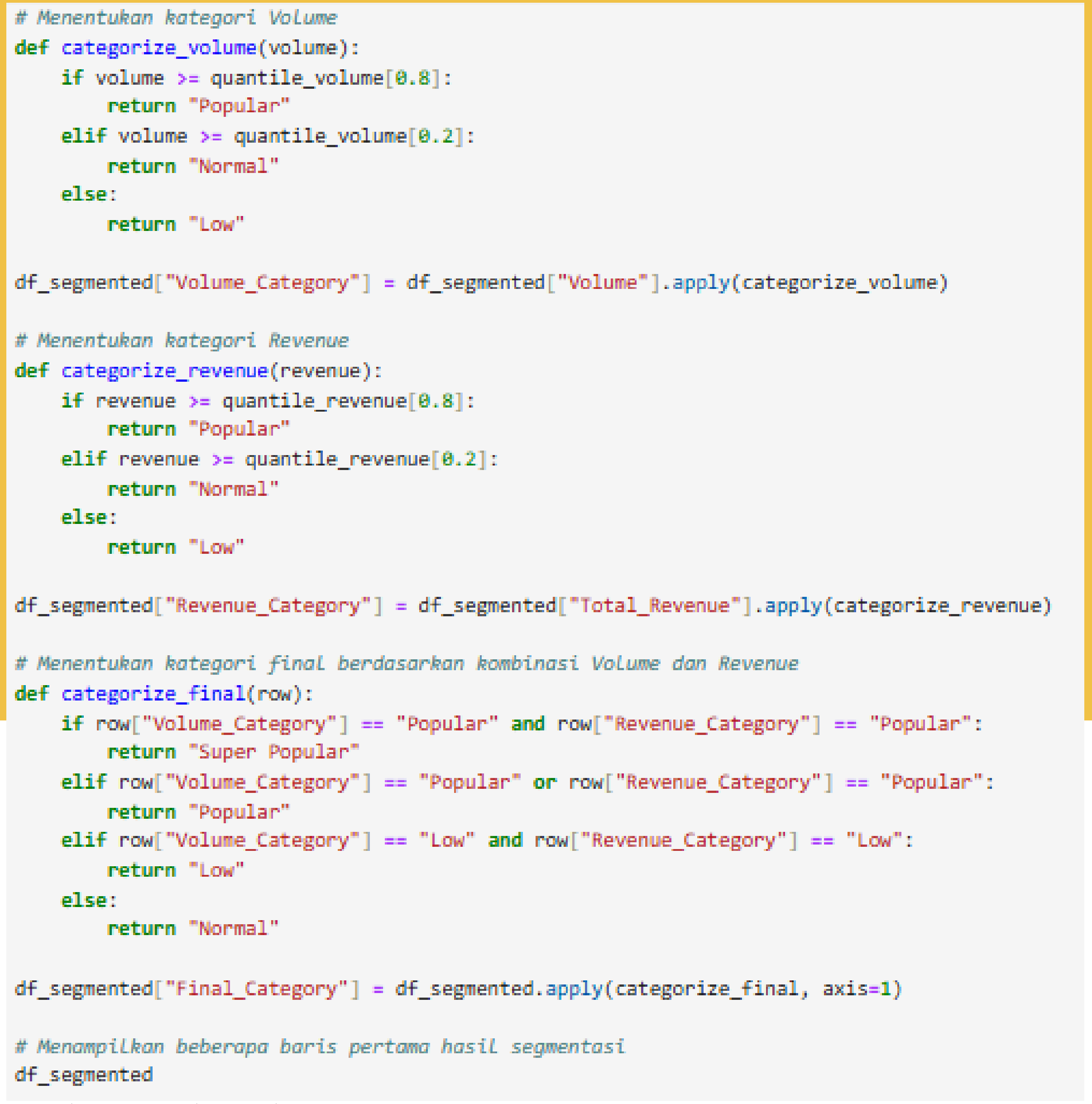
Adam Maulana

Click Here To See Full Coding

Data Segmentation



```python
# Menghitung Volume Transaksi
volume_transaksi = df_sales.groupby("ProductNo")["TransactionNo"].nunique().reset_index()
volume_transaksi.columns = ["ProductNo", "Volume"]

# Menghitung Total Revenue (Price * Quantity)
df_sales["Revenue"] = df_sales["Price"] * df_sales["Quantity"]
total_revenue = df_sales.groupby("ProductNo")["Revenue"].sum().reset_index()
total_revenue.columns = ["ProductNo", "Total_Revenue"]

# Menggabungkan kedua metrik
df_segmented = volume_transaksi.merge(total_revenue, on="ProductNo")

# Mengambil nama produk dari df_sales
product_names = df_sales[["ProductNo", "ProductName"]].drop_duplicates()

# Menggabungkan ProductName ke df_segmented
df_segmented = df_segmented.merge(product_names, on="ProductNo", how="left")

# Menentukan batas segmentasi (20% teratas, 20-80%, dan 20% terbawah)
quantile_volume = df_segmented["Volume"].quantile([0.2, 0.8])
quantile_revenue = df_segmented["Total_Revenue"].quantile([0.2, 0.8])
```



```python
# Menentukan kategori Volume
def categorize_volume(volume):
    if volume >= quantile_volume[0.8]:
        return "Popular"
    elif volume >= quantile_volume[0.2]:
        return "Normal"
    else:
        return "Low"

df_segmented["Volume_Category"] = df_segmented["Volume"].apply(categorize_volume)

# Menentukan kategori Revenue
def categorize_revenue(revenue):
    if revenue >= quantile_revenue[0.8]:
        return "Popular"
    elif revenue >= quantile_revenue[0.2]:
        return "Normal"
    else:
        return "Low"

df_segmented["Revenue_Category"] = df_segmented["Total_Revenue"].apply(categorize_revenue)

# Menentukan kategori final berdasarkan kombinasi Volume dan Revenue
def categorize_final(row):
    if row["Volume_Category"] == "Popular" and row["Revenue_Category"] == "Popular":
        return "Super Popular"
    elif row["Volume_Category"] == "Popular" or row["Revenue_Category"] == "Popular":
        return "Popular"
    elif row["Volume_Category"] == "Low" and row["Revenue_Category"] == "Low":
        return "Low"
    else:
        return "Normal"

df_segmented["Final_Category"] = df_segmented.apply(categorize_final, axis=1)

# Menampilkan beberapa baris pertama hasil segmentasi
df_segmented
```

Adam Maulana

In this project, I performed product segmentation based on:

- Transaction Volume: The number of transactions involving a product, regardless of the quantity purchased per transaction.
- Total Revenue: The total dollar revenue generated from the sales of each product.

The segmentation was done using the following criteria:

- Total revenue is calculated as the product of item price and quantity sold.
- Products were segmented by transaction volume as follows:
  - The top 20% highest in transaction volume were labeled "Popular"
  - The middle 60% (20%-80%) were labeled "Normal"
  - The bottom 20% were labeled "Low"
- A similar rule was applied to total revenue:
  - Top 20% in revenue → "Popular"
  - Middle 60% → "Normal"
  - Bottom 20% → "Low"

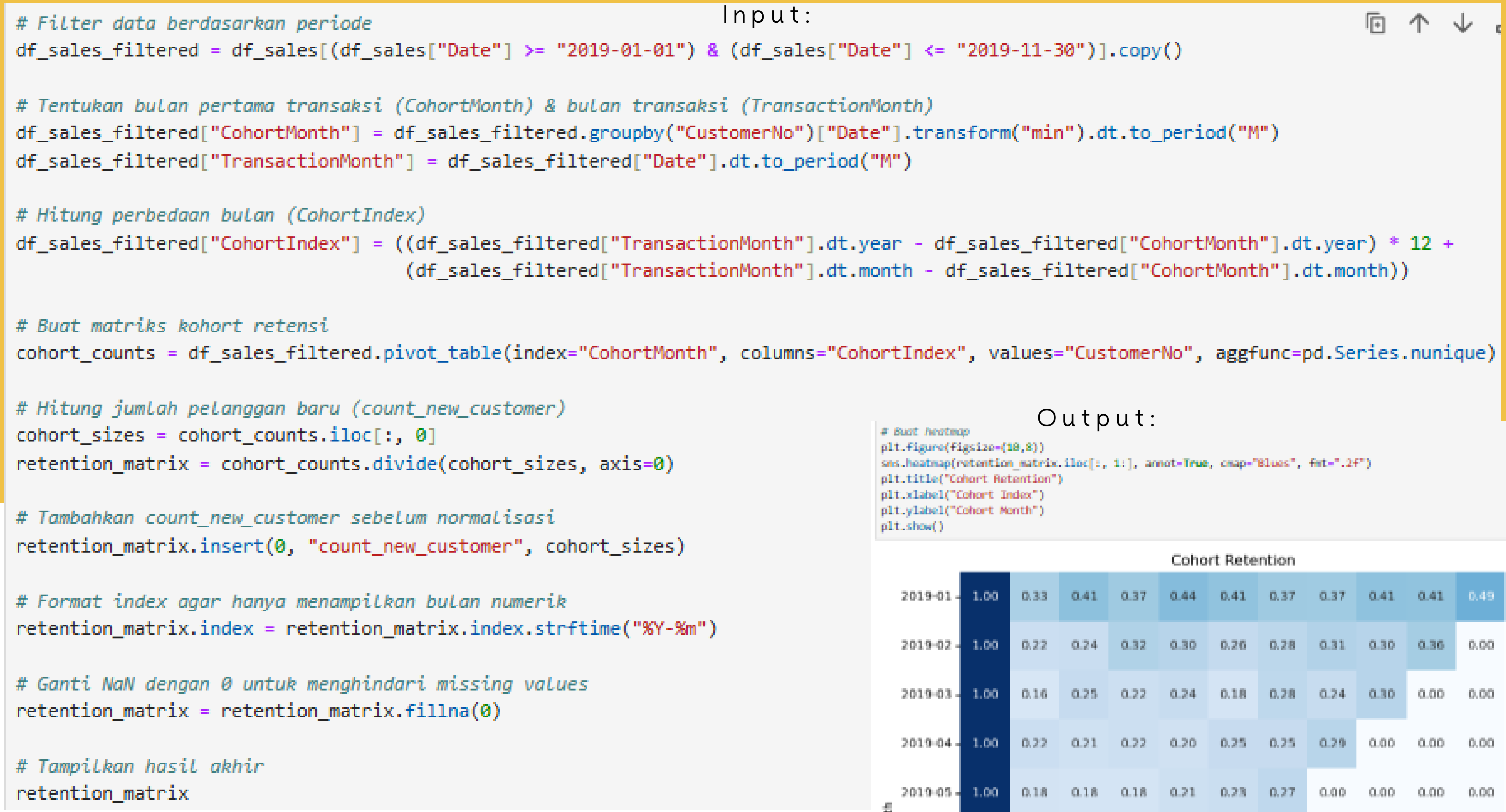
Then, based on the combination of both segmentations:

- Products that were "Popular" in both transaction volume and total revenue were labeled "Super Popular"
- Products that were "Popular" in one metric and "Normal" in the other were labeled "Popular"
- Products that were "Low" in both metrics were labeled "Low"
- All other combinations were categorized as "Normal"

Click Here To See Full Coding

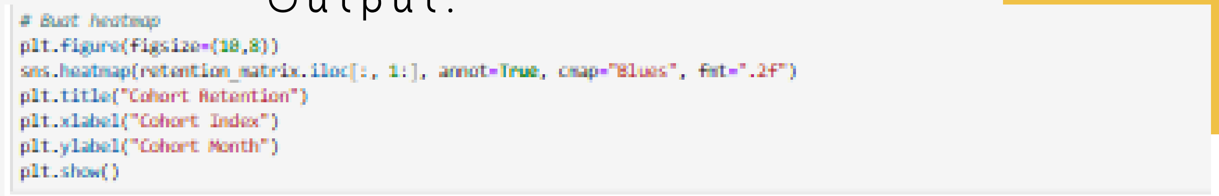
Retention Cohort Analysis

Input:

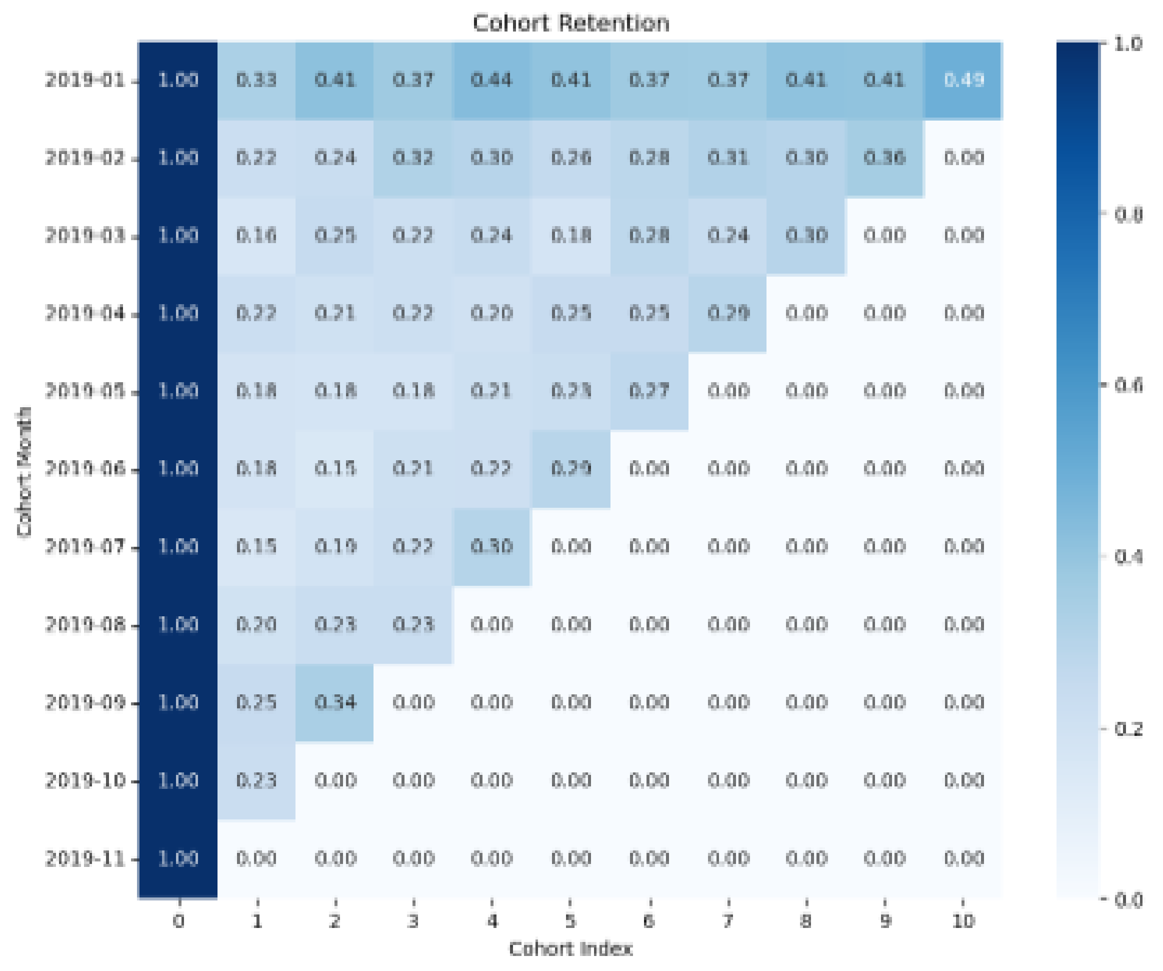


```python
# Filter data berdasarkan periode
df_sales_filtered = df_sales[(df_sales["Date"] >= "2019-01-01") & (df_sales["Date"] <= "2019-11-30")].copy()

# Tentukan bulan pertama transaksi (CohortMonth) & bulan transaksi (TransactionMonth)
df_sales_filtered["CohortMonth"] = df_sales_filtered.groupby("CustomerNo")["Date"].transform("min").dt.to_period("M")
df_sales_filtered["TransactionMonth"] = df_sales_filtered["Date"].dt.to_period("M")

# Hitung perbedaan bulan (CohortIndex)
df_sales_filtered["CohortIndex"] = ((df_sales_filtered["TransactionMonth"].dt.year - df_sales_filtered["CohortMonth"].dt.year) * 12 +
                                    (df_sales_filtered["TransactionMonth"].dt.month - df_sales_filtered["CohortMonth"].dt.month))

# Buat matriks kohort retensi
cohort_counts = df_sales_filtered.pivot_table(index="CohortMonth", columns="CohortIndex", values="CustomerNo", aggfunc=pd.Series.nunique)

# Hitung jumlah pelanggan baru (count_new_customer)
cohort_sizes = cohort_counts.iloc[:, 0]
retention_matrix = cohort_counts.divide(cohort_sizes, axis=0)

# Tambahkan count_new_customer sebelum normalisasi
retention_matrix.insert(0, "count_new_customer", cohort_sizes)

# Format index agar hanya menampilkan bulan numerik
retention_matrix.index = retention_matrix.index.strftime("%Y-%m")

# Ganti NaN dengan 0 untuk menghindari missing values
retention_matrix = retention_matrix.fillna(0)

# Tampilkan hasil akhir
retention_matrix
```

Output:



```python
# Buat heatmap
plt.figure(figsize=(10,8))
sns.heatmap(retention_matrix.iloc[:, 1:], annot=True, cmap="Blues", fmt=".2f")
plt.title("Cohort Retention")
plt.xlabel("Cohort Index")
plt.ylabel("Cohort Month")
plt.show()
```



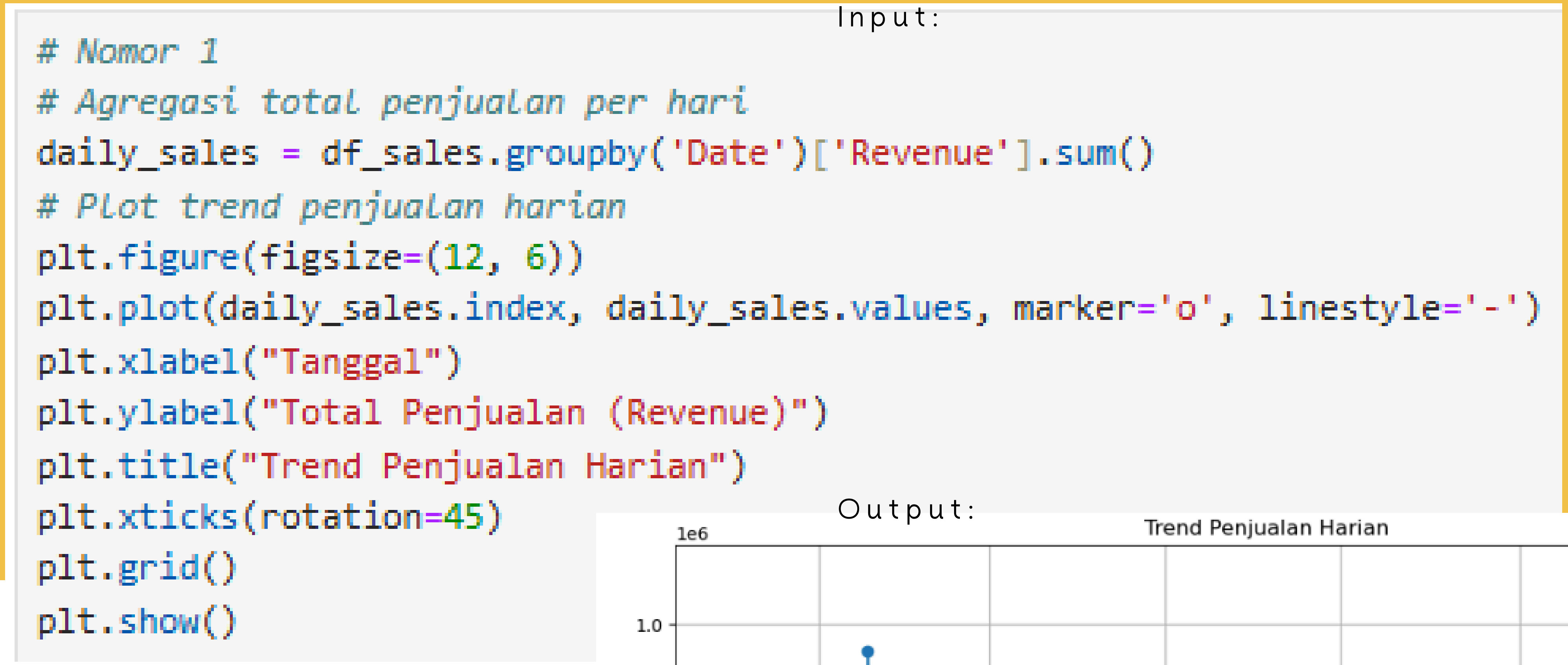
	0	1	2	3	4	5	6	7	8	9	10
2019-01	1.00	0.33	0.41	0.37	0.44	0.41	0.37	0.37	0.41	0.41	0.49
2019-02	1.00	0.22	0.24	0.32	0.30	0.26	0.28	0.31	0.30	0.36	0.00
2019-03	1.00	0.16	0.25	0.22	0.24	0.18	0.28	0.24	0.30	0.00	0.00
2019-04	1.00	0.22	0.21	0.22	0.20	0.25	0.25	0.29	0.00	0.00	0.00
2019-05	1.00	0.18	0.18	0.18	0.21	0.23	0.27	0.00	0.00	0.00	0.00
2019-06	1.00	0.18	0.15	0.21	0.22	0.29	0.00	0.00	0.00	0.00	0.00
2019-07	1.00	0.15	0.19	0.22	0.30	0.00	0.00	0.00	0.00	0.00	0.00
2019-08	1.00	0.20	0.23	0.23	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2019-09	1.00	0.25	0.34	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2019-10	1.00	0.23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2019-11	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Adam Maulana

Click Here To See Full Coding

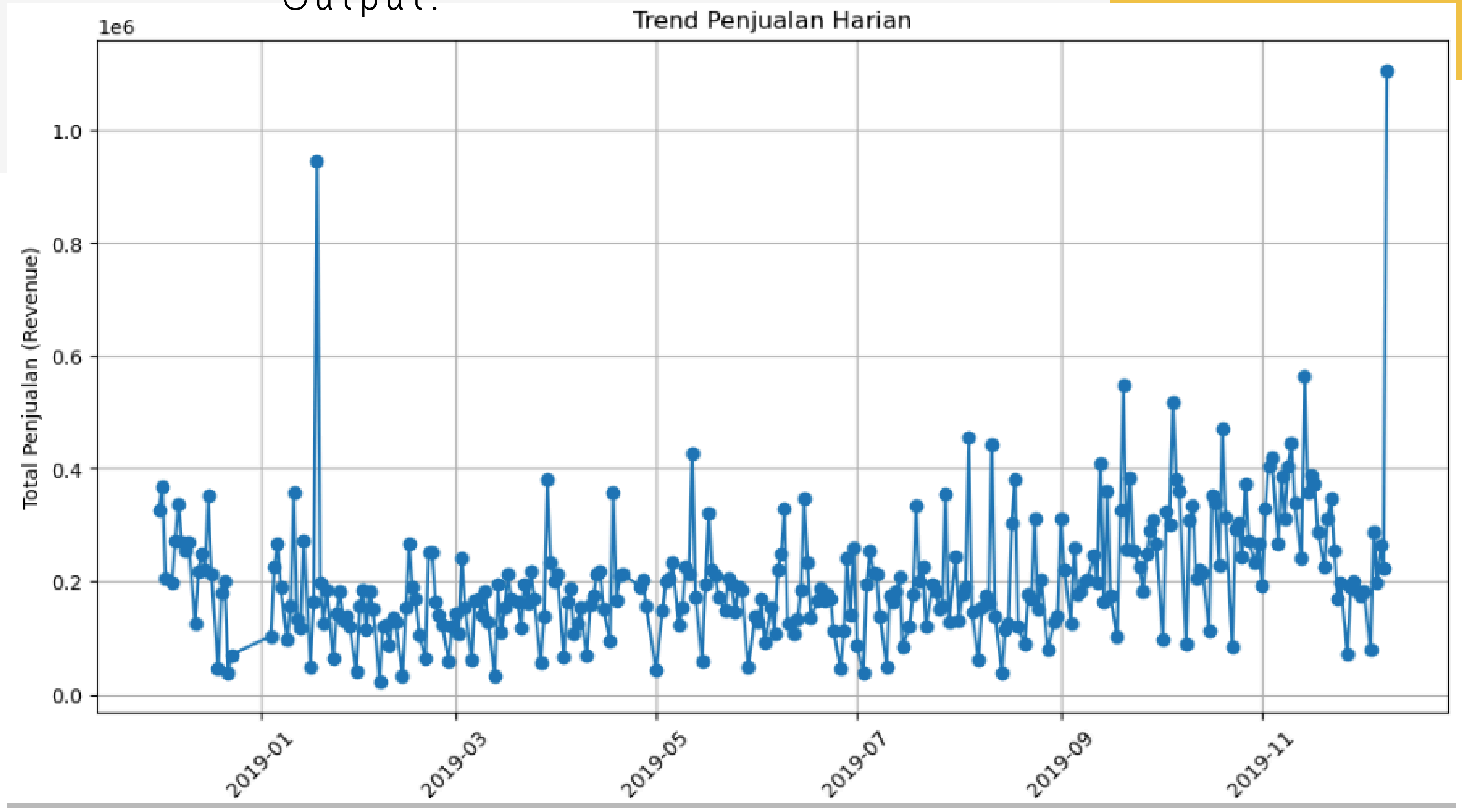
Time Series Analysis

Input:



```
# Nomor 1
# Agregasi total penjualan per hari
daily_sales = df_sales.groupby('Date')['Revenue'].sum()
# Plot trend penjualan harian
plt.figure(figsize=(12, 6))
plt.plot(daily_sales.index, daily_sales.values, marker='o', linestyle='-')
plt.xlabel("Tanggal")
plt.ylabel("Total Penjualan (Revenue)")
plt.title("Trend Penjualan Harian")
plt.xticks(rotation=45)
plt.grid()
plt.show()
```

Output:

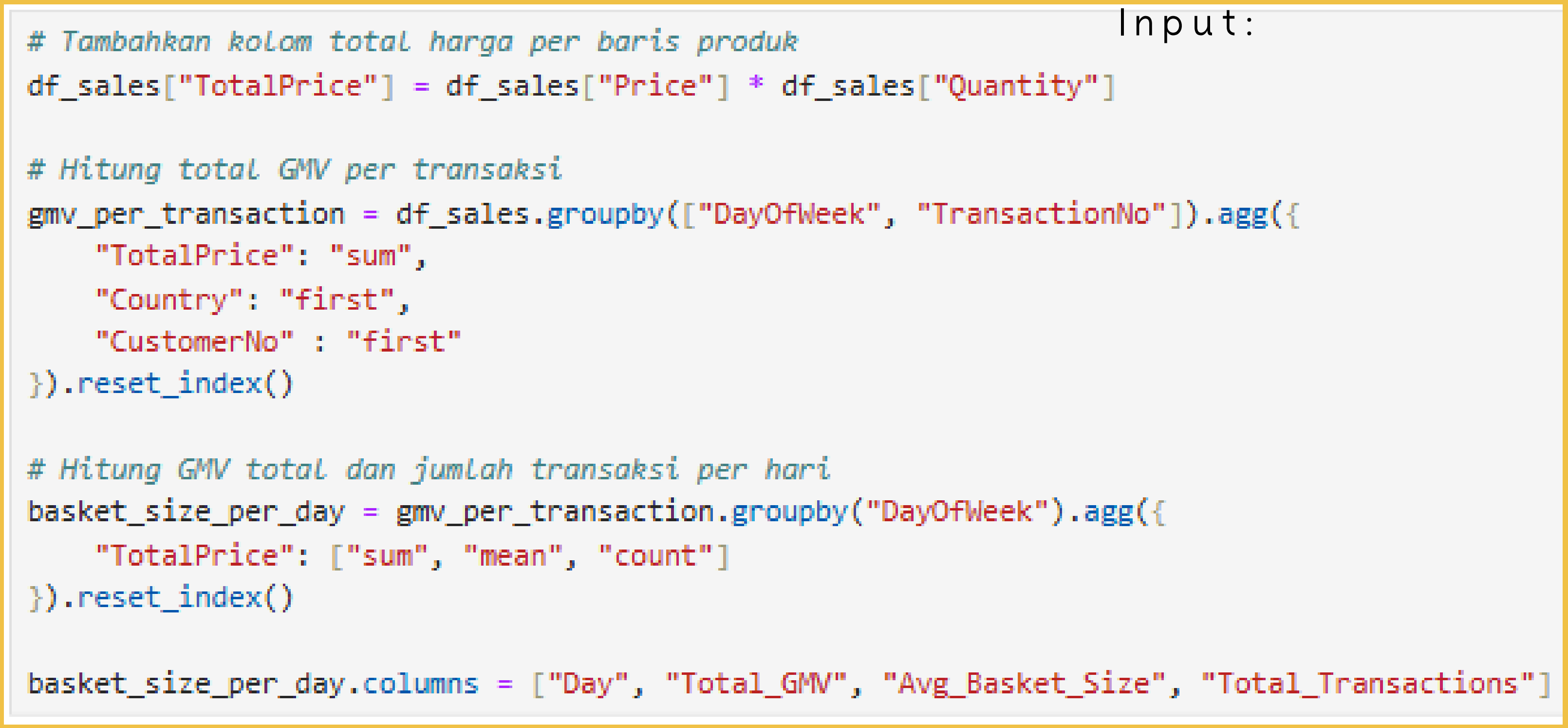


Adam Maulana

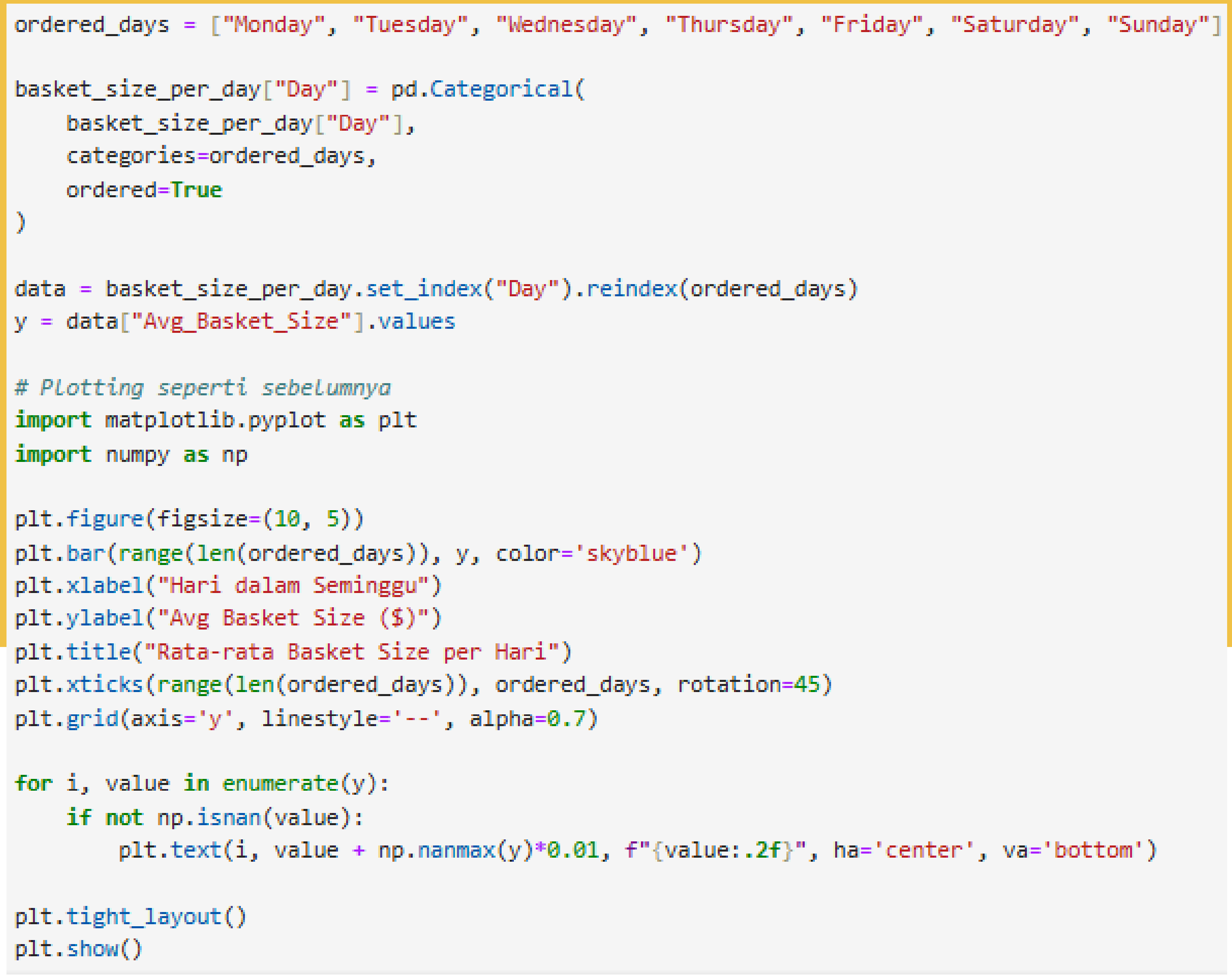
Click Here To See Full Coding

Basket Size Analysis

Input:

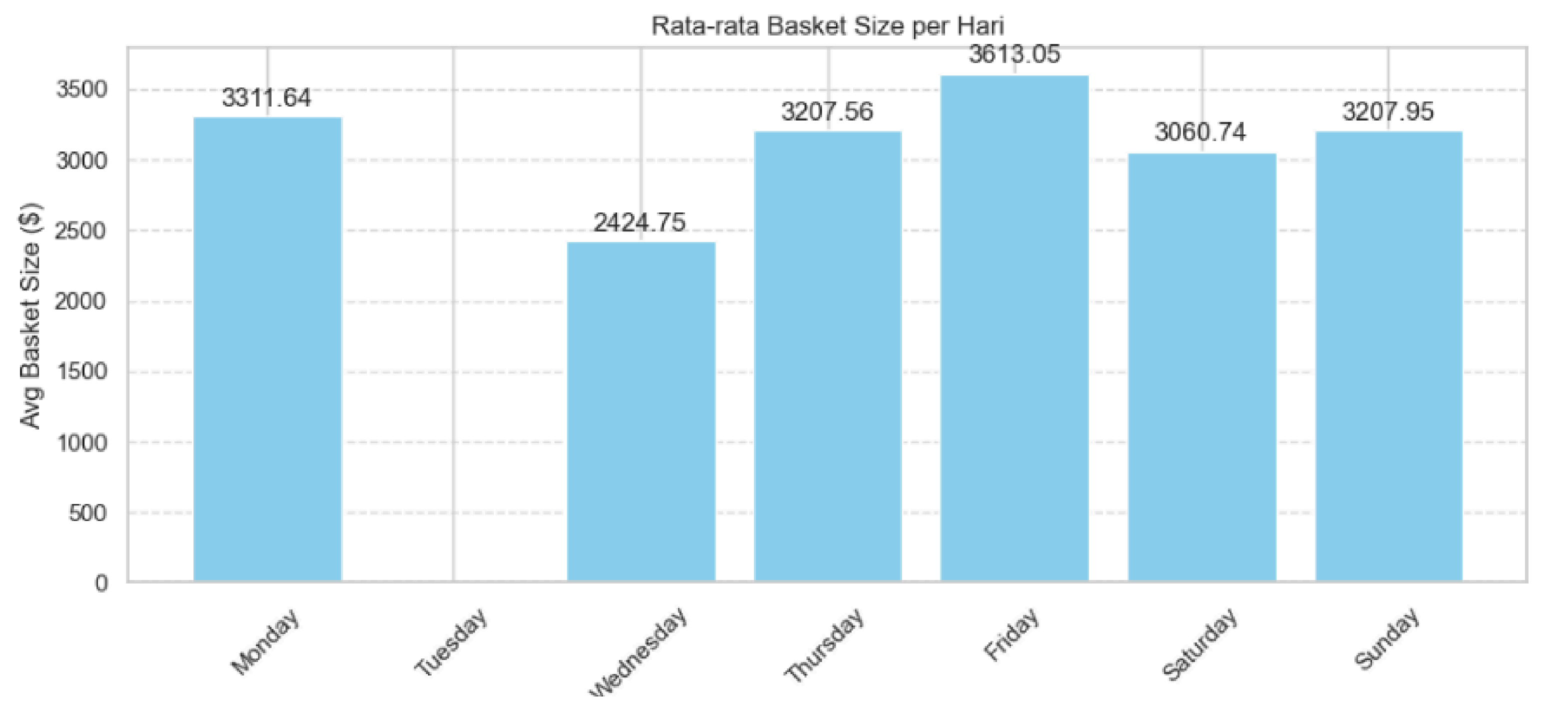


```python
# Tambahkan kolom total harga per baris produk
df_sales["TotalPrice"] = df_sales["Price"] * df_sales["Quantity"]

# Hitung total GMV per transaksi
gmv_per_transaction = df_sales.groupby(["DayOfWeek", "TransactionNo"]).agg({
    "TotalPrice": "sum",
    "Country": "first",
    "CustomerNo" : "first"
}).reset_index()

# Hitung GMV total dan jumlah transaksi per hari
basket_size_per_day = gmv_per_transaction.groupby("DayOfWeek").agg({
    "TotalPrice": ["sum", "mean", "count"]
}).reset_index()

basket_size_per_day.columns = ["Day", "Total_GMV", "Avg_Basket_Size", "Total_Transactions"]
```



```python
ordered_days = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]

basket_size_per_day["Day"] = pd.Categorical(
    basket_size_per_day["Day"],
    categories=ordered_days,
    ordered=True
)

data = basket_size_per_day.set_index("Day").reindex(ordered_days)
y = data["Avg_Basket_Size"].values

# Plotting seperti sebelumnya
import matplotlib.pyplot as plt
import numpy as np

plt.figure(figsize=(10, 5))
plt.bar(range(len(ordered_days)), y, color='skyblue')
plt.xlabel("Hari dalam Seminggu")
plt.ylabel("Avg Basket Size ($)")
plt.title("Rata-rata Basket Size per Hari")
plt.xticks(range(len(ordered_days)), ordered_days, rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)

for i, value in enumerate(y):
    if not np.isnan(value):
        plt.text(i, value + np.nanmax(y)*0.01, f"{value:.2f}", ha='center', va='bottom')

plt.tight_layout()
plt.show()
```

Output:



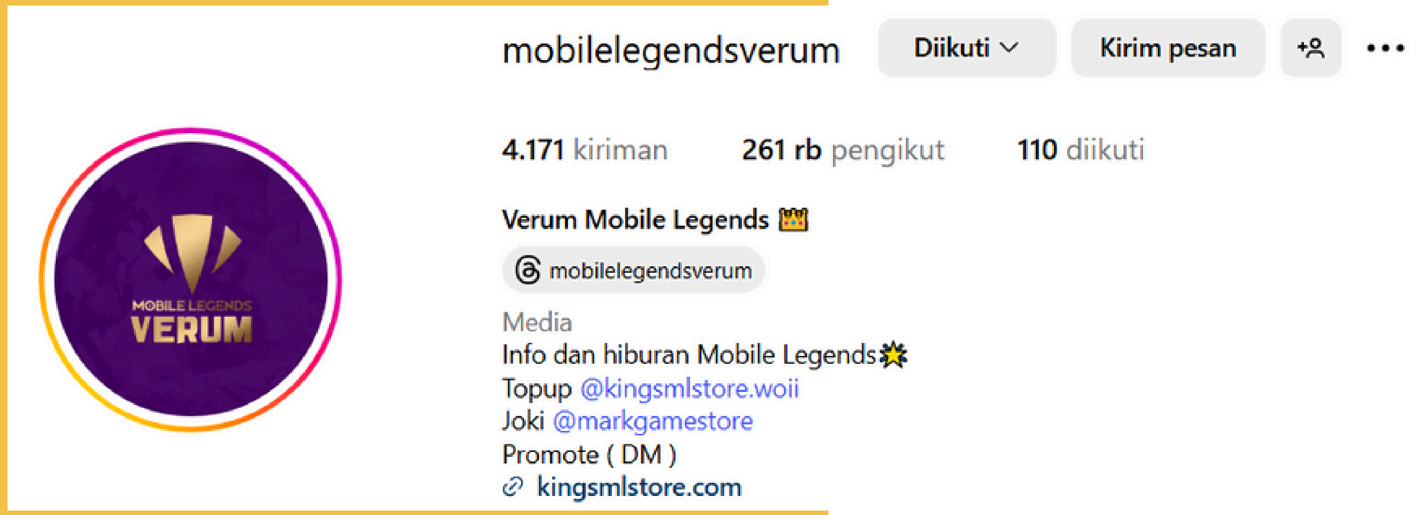
Adam Maulana

Content Creator.

Account - Content

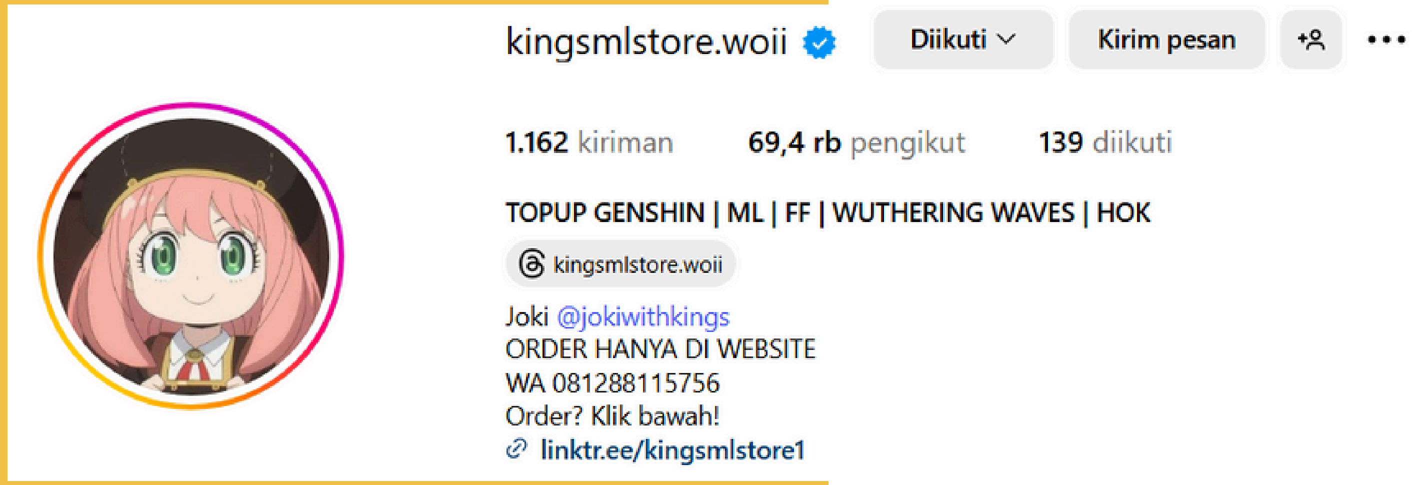
Adam Maulana

Account


mobilelegendsverum Diikuti Kirim pesan

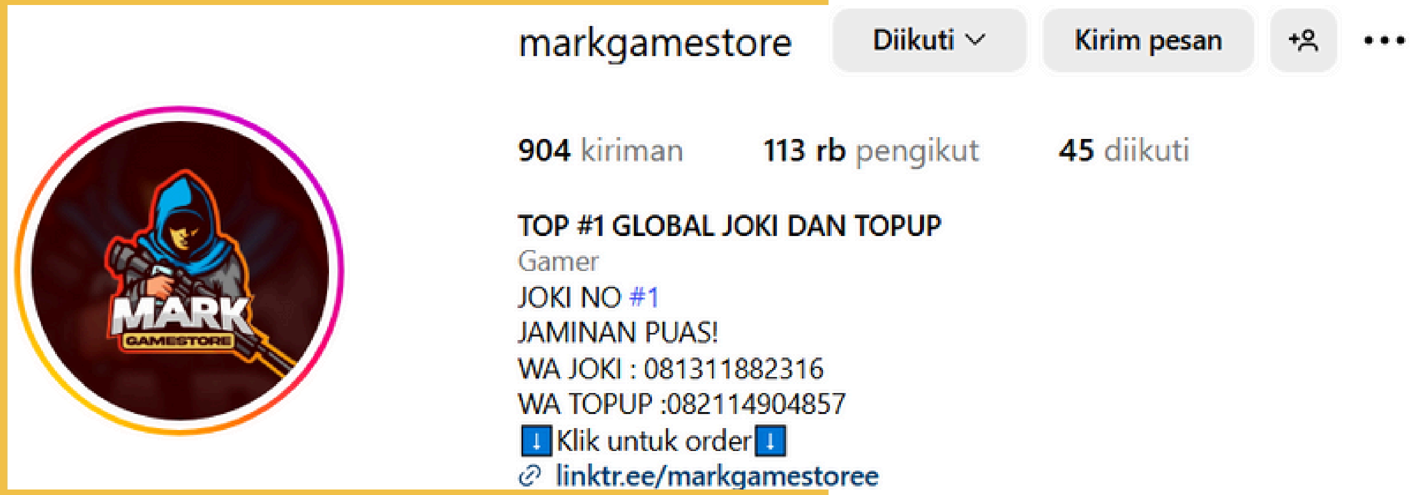
4.171 kiriman 261 rb pengikut 110 diikuti

Verum Mobile Legends 👑
mobilelegendsverum
Media
Info dan hiburan Mobile Legends💥
Topup @kingsmlstore.woii
Joki @markgamestore
Promote (DM)
kingsmlstore.com


kingsmlstore.woii Diikuti Kirim pesan

1.162 kiriman 69,4 rb pengikut 139 diikuti

TOPUP GENSHIN | ML | FF | WUTHERING WAVES | HOK
kingsmlstore.woii
Joki @jokiwithkings
ORDER HANYA DI WEBSITE
WA 081288115756
Order? Klik bawah!
linktr.ee/kingsmlstore1


markgamestore Diikuti Kirim pesan

904 kiriman 113 rb pengikut 45 diikuti

TOP #1 GLOBAL JOKI DAN TOPUP
Gamer
JOKI NO #1
JAMINAN PUAS!
WA JOKI : 081311882316
WA TOPUP :082114904857
⬇️Klik untuk order⬇️
linktr.ee/markgamestoree

Responsible for daily content creation across multiple social media accounts, ensuring consistent engagement and brand alignment on each platform.

Adam Maulana

Content

Adam Maulana

MARK GAMESTORE
MARKGAMESTORE
0821 1490 4857
SC: ig/@wsscommunity
REKT
TAZZ
ALTER EGO
PUNYA TRIO JUARA DUNIA
MARKGAMESTORE.COM

Roster MPL S13 Alter Ego terdapat TRIO yang pernah juara dunia. Berikut adalah 3 Pemain Roster Alter Ego yang akan bertanding pada 8 maret 2024 di MPL S13 nanti!

COMBO CLEAR CEPAT
SC: tt/@irish_caviar
KINGSMLSTORE
KINGSMLSTORE.WOII
0812 8811 5756
KINGSMLSTORE.COM
BCA OVO GOPAY DANA

Baru baru ini, sedang ramai combo baru Claude dengan Luo Yi yang menampilkan gameplay clear cepat dan super aman pada MPL PH.

KINGSDAILY
KINGSMLSTORE
@KINGSMLSTORE.WOII
0812 8811 5756
KINGSMLSTORE.COM

Bukti hero ini juga bisa dipakai untuk roam, bukan hanya sebagai preman EXP Laner saja. Coba cek build dan set up hero berikut ini.

VREN
COACH
@MOBILELEGENDSVERUM

PERMASALAHAN COACH VREN

Social Media sedang digemparkan masalah dari Coach Vren yaitu coach dari RRQ Hoshi yang sekarang ada di zona merah MPL ID S13 kali ini. berikut beberapa rangkuman permasalahan dari mimin. simak slide selanjutnya.

Published over 250 content pieces across multiple social media platforms.

check here to see all content

Thank you.

Contact Me

+62 87-880-925-800
linkedin.com/in/adammaulana100
adammaulana100@gmail.com

Adam Maulana