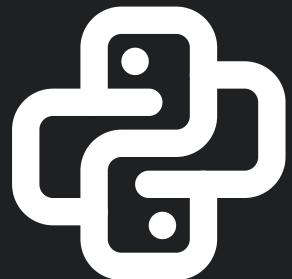


FINAL PROJECT PYTHON

NAMA ANGGOTA KELOMPOK:

- Ayubi Alfasiri
- Adrian Hafiz David
- Ahmad Suhaebi
- Arifin Hanafi
- Diana Ayu Lestari
- Fanny Oscar B.D.K.R.
- Fatimah Azahra
- Hana Priska Sihite
- Latifah Fauziah
- Mayka Raflisiansyah
- Muhammad Saikhudin
- Reza Difaputra
- Slamet Indra Nurcahya

Final Project



MENTOR : KAK ERIKA

Kelompok B4

Background

Perusahaan **PT. Tokopedia (bukan nama aslinya)** merupakan perusahaan yang bergerak di bidang E-Commerce yang menjual product-product dengan total jumlah **15 kategori product** yang disediakan. Berjalannya waktu perusahaan dihadapkan pada permasalahan-permasalahan yang bersifat pada **penjualan yang rendah pada suatu kategori, stock gudang yang menumpuk, pelanggan yang stagnan dan ketidak sesuaian strategi pemasaran yang diterapakan**, menjadikan kami sebagai tim data analyst mengharuskan membreakdown beberapa data yang dibutuhkan untuk mengetahui apa yang sebenarnya terjadi pada permasalahan-permasalahan tersebut. Dengan demikian harapan perusahaan pada kami sangat tinggi terkait keberlangsungan bisnis perusahaan ke depannya.

Semoga penyajian data ini dapat membantu perusahaan.

Data Source

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from pandas.tseries.offsets import BDay
```

Dataset

Data yang digunakan adalah data yang berasal dari Tokopedia (*bukan data sesungguhnya*). Mengenai penjelasan dataset adalah sebagai berikut:

variable	class	description
order_detail:		
id	object	angka unik dari order / id_order
customer_id	object	angka unik dari pelanggan
order_date	object	tanggal saat dilakukan transaksi
sku_id	object	angka unik dari produk (sku adalah stock keeping unit)
price	int64	harga yang tertera pada tagging harga
qty_ordered	int64	Jumlah barang yang dibeli oleh pelanggan
before_discount	float64	Nilai harga total dari produk (price * qty_ordered)
discount_amount float64 nilai diskon produk total		
after_discount	float64	Nilai harga total produk ketika sudah dikurangi dengan diskon
is_gross	int64	Menunjukkan pelanggan belum membayar pesanan
is_valid	int64	Menunjukkan pelanggan sudah melakukan pembayaran
is_net	int64	Menunjukkan transaksi sudah selesai
payment_id	int64	Angka unik dari metode pembayaran
sku_detail:		
id	object	Angka unik dari produk (dapat digunakan untuk key saat join)
sku_name	object	Nama dari produk
base_price	float64	Harga barang yang tertera pada tagging harga / price
cogs	int64	Cost of goods sold / Total biaya untuk menjual 1 produk
category	object	Kategori produk
customer_detail:		
id	object	Angka unik dari pelanggan
registered_date	object	Tanggal pelanggan mulai mendaftarkan diri sebagai anggota
payment_detail:		
id	int64	Angka unik dari metode pembayaran
payment_method	object	Metode pembayaran yang digunakan



Data Source

Menggabungkan dataframe :

```
1 #Menjalankan SQL di Colab
2 from sqlite3 import connect
3 conn = connect(':memory:')
4 df_od.to_sql('order_detail',conn, index=False, if_exists='replace')
5 df_pd.to_sql('payment_detail', conn, index=False, if_exists='replace')
6 df_sd.to_sql('sku_detail', conn, index=False, if_exists='replace')
7 df_cd.to_sql('customer_detail', conn, index=False, if_exists='replace')
```

3998

```
1 #Query SQL untuk menggabungkan data
2 df = pd.read_sql("""
3     SELECT
4         order_detail.*,
5         payment_detail.payment_method,
6         sku_detail.sku_name,
7         sku_detail.base_price,
8         sku_detail.cogs,
9         sku_detail.category,
10        customer_detail.registered_date
11    FROM order_detail
12    LEFT JOIN payment_detail
13        on payment_detail.id = order_detail.payment_id
14    LEFT JOIN sku_detail
15        on sku_detail.id = order_detail.sku_id
16    LEFT JOIN customer_detail
17        on customer_detail.id = order_detail.customer_id
18     "", conn)
```

Data Source

Hasil Penggabungan :

index	id	customer_id	order_date	sku_id	price	qty_order ed	before_di scount	discount_ amount	after_disc ount	is_gross	is_valid	is_net	payment_i d	payment_ method	sku_name	base_pric e	cogs	category	registered _date
0	ODR9939707760w	C713589L	11/19...	P858068	26100	200	5220000	2610000	2610000	1	1	0	5	jazzwallet	RB_Dettol Germ Busting	26100	18270	Others	7/7/2...
1	ODR7448356649d	C551551L	11/19...	P886455	1971942	5	9859710	2464927.5	7394782.5	1	0	0	5	jazzwallet	PS4_Slim -500GB	1971942	1321182	Entertain ment	11/20...
2	ODR4011281866z	C685596L	11/25...	P678648	7482000	1	7482000	2065344.62	5416655.38	1	0	0	4	Payaxis	Changhon g Ruba 55 Inches	7482000	5162580	Entertain ment	11/19...
3	ODR3378927994s	C830683L	11/22...	P540013	3593680	1	3593680	1455440.4	2138239.6	1	1	1	5	jazzwallet	Dawlance _Inverter 30	3593680	3054628	Appliance s	11/3/...
4	ODR4904430099k	C191766L	11/21...	P491032	4413220	1	4413220	1059172.8	3354047.2	1	1	1	4	Payaxis	Dawlance _Inverter- 45 2.0 ton	4413220	3177472	Appliance s	7/5/2...

Data Source

Ubah tipe data untuk pengolahan lebih lanjut :

	0
id	object
customer_id	object
order_date	object
sku_id	object
price	int64
qty_ordered	int64
before_discount	float64
discount_amount	float64
after_discount	float64
is_gross	int64
is_valid	int64

is_net	int64
payment_id	int64
payment_method	object
sku_name	object
base_price	float64
cogs	int64
category	object
registered_date	object
dtype: object	

```
1 #Mengubah tipe data agar mudah dilakukan pengolahan data
2 df = df.astype({"before_discount":'int', "discount_amount":'int', "after_discount":'int','base_price':'int'})
3 df.dtypes

1 #Mengubah tipe kolom Date menjadi Datetime
2 df['order_date']= pd.to_datetime(df['order_date'])
3 df['registered_date']= pd.to_datetime(df['registered_date'])
4 df.dtypes
```



Client Request 1

Dear Data Analyst,

Akhir tahun ini, perusahaan akan memberikan hadiah bagi pelanggan yang memenangkan kompetisi Festival Akhir Tahun. Tim Marketing membutuhkan bantuan untuk menentukan perkiraan hadiah yang akan diberikan pada pemenang kompetisi nantinya. Hadiah tersebut akan diambil dari TOP 5 Produk dari Kategori Mobiles & Tablets selama tahun 2022, dengan jumlah kuantitas penjualan (valid = 1) paling tinggi.

Mohon bantuan, untuk mengirimkan data tersebut sebelum akhir bulan ini ke Tim Marketing. Atas bantuan yang diberikan, kami mengucapkan terima kasih.

Regards
Tim Marketing

Jawaban No 1

Query + Penjelasannya :

```
1 top_5_products = (
2     df
3         .query("category == 'Mobiles & Tablets' and is_valid == 1") #Filter kolom category based on Mobiles & Tablets dan is_valid = 1
4         .assign(order_year=df['order_date'].dt.year) #Membuat kolom baru dan mengextract tahun dari data order_date
5         .query("order_year == 2022") #Filter tahun hanya 2022 saja
6         .groupby('sku_name', as_index=False)['qty_ordered'] #Grouping dan menjumlahkan qty_ordered based on sku_name
7         .sum()
8         .sort_values('qty_ordered', ascending=False) #Mengurutkan data qty_ordered secara descending (besar ke kecil)
9         .head(5) #Menampilkan hanya 5 data teratas saja
10        .rename(columns={
11            'sku_name': 'Nama Produk', #Merubah nama kolom sku_name ke Nama Produk
12            'qty_ordered': 'Total Kuantitas Terjual' #Merubah Nama kolom qty_ordered ke Total Kuantitas Terjual
13        })
14        .reset_index(drop=True) #memastikan indeks kolom lama tidak ditambahkan sebagai kolom baru di
15    )
16
17 # Nomor urut
18 top_5_products.index = top_5_products.index + 1 #Menampilkan index baru dengan penomoran dimulai dari 0+1
19
20 top_5_products #Menampilkan hasil query top_5_products
```

Jawaban No 1

Result:

	Nama Produk	Total Kuantitas Terjual
1	IDROID_BALRX7-Gold	1000
2	IDROID_BALRX7-Jet black	31
3	Infinix Hot 4-Gold	15
4	samsung_Grand Prime Plus-Black	11
5	infinix_Zero 4-Grey	10

insight :

- IDROID-BALRX7-Gold terjual 1.000 unit dan jauh lebih tinggi dibanding produk lainnya. Sehingga, produk ini merupakan produk yang **paling diminati pelanggan dan paling populer**.
- Minat pasar di kategori Mobiles & Tablets tidak merata. Hal ini dapat dilihat dari perbedaan kuantitas terjual antar produk yang sangat jauh.
- Warna/varian berpengaruh pada penjualan. Varian Gold muncul dua kali dalam daftar top 5, yaitu masing-masing pada merek IDROID dan Infinix. Hal ini mengindikasikan bahwa **warna gold lebih disukai oleh konsumen**.

Rekomendasi:

- IDROID-BALRX7-Gold dapat dijadikan sebagai hadiah utama dalam kompetisi Festival Akhir Tahun karena merupakan produk yang paling banyak terjual dibandingkan produk lainnya. Dengan memilih produk yang sudah terbukti laku dipasaran, perusahaan dapat memastikan bahwa hadiah yang diberikan benar-benar menarik bagi pemenang.
- Produk yang berada di peringkat kedua hingga kelima tetap dapat dimanfaatkan sebagai hadiah pendamping, misalnya untuk juara dua dan juara tiga.
- Perbedaan penjualan yang cukup jauh antar produk menunjukkan perlunya **evaluasi terhadap produk dengan penjualan rendah** untuk mengetahui apakah rendahnya penjualan disebabkan oleh kurangnya promosi, harga yang kurang kompetitif, atau minat pasar yang rendah terhadap produk tersebut.

Client Request 2

Dear Data Analyst,

Menindaklanjuti meeting gabungan Tim Werehouse dan Tim Marketing, kami menemukan bahwa ketersediaan **stock produk dengan Kategori Others pada akhir 2022 kemarin masih banyak.**

1. Kami mohon bantuan untuk melakukan pengecekan **data penjualan kategori tersebut dengan tahun 2021** secara kuantitas penjualan. Dugaan sementara kami, telah terjadi **penurunan kuantitas penjualan pada 2022 dibandingkan 2021. (Mohon juga menampilkan data ke-15 kategori)**.
2. Apabila memang terjadi penurunan kuantitas penjualan pada kategori Others, kami mohon bantuan untuk menyediakan data **TOP 20 nama produk yang mengalami penurunan paling tinggi pada 2022 jika dibanding dengan 2021.** Hal ini kami gunakan sebagai bahan diskusi pada meeting selanjutnya.

Mohon bantuan untuk mengirimkan data tersebut paling lambat 4 hari dari hari ini. Atas bantuan yang diberikan, kami mengucapkan terima kasih.

Regards
Tim Werehouse

Jawaban No 2.1

Penjelasan Query:

```
1 # komparasi penjualan setiap kategori
2 # To compare 2021 and 2022, it's best to filter data for each year separately.
3 filter_category_2021 = df[
4     (df['is_valid'] == 1) &           #memfilter data dengan pembelian min 1 kali
5     (df['order_date'].dt.year == 2021) #memfilter data hanya pada tahun 2021
6 ].copy()
7
8 filter_category_2022 = df[
9     (df['is_valid'] == 1) &           #memfilter data dengan pembelian min 1 kali
10    (df['order_date'].dt.year == 2022) #memfilter data hanya pada tahun 2022
11 ].copy()
12
13 #langkah 4
14 #grouping based on sku_name dan tahun 2021
15 Product_itemall_2021 = (
16     filter_category_2021
17     .groupby(['category'], as_index=False)  #grouping based on category, sku_name (object type)
18     .agg(qty_ordered_2021=('qty_ordered', 'sum'))      #fungsi aggregat sum terhadap qty_ordered dan merubah nama kolom menjadi qty_order
19     .sort_values(by='qty_ordered_2021', ascending=False) #mengurutkan data based on qty_ordered_2021 secara descending (dari besar ke ke
20 )
21 #langkah 5
22 #grouping based on sku_name dan tahun 2022
23 Product_itemall_2022 = (
24     filter_category_2022
25     .groupby(['category'], as_index=False) #grouping based on category, sku_name (object type)
26     .agg(qty_ordered_2022=('qty_ordered', 'sum')) # Corrected agg syntax and renamed directly
27     .sort_values(by='qty_ordered_2022', ascending=False) #mengurutkan data based on qty_ordered_2022 secara descending (dari besar ke ke
28
29 )
```

Jawaban No 2.1

Query + Penjelasannya :

```
31 #langkah 6
32 #Merge data grouping penjualan 2021 dan 2022
33 merged_data = pd.merge(Product_itemall_2021, Product_itemall_2022, on='category', how='outer')
34 merged_data = merged_data.fillna(0).sort_values(by='qty_ordered_2021', ascending=False)
35
36 #Langkah 7
37 #lakukan pengurangan qty 2021 terhadap qty 2022
38 merged_data['selisih_penjualan_2021vs2022'] = merged_data['qty_ordered_2022'] - merged_data['qty_ordered_2021']
39
40 #Langkah 8
41 #Buat dummy varibale flag category untuk memastikan join product 2021 vs 2022 keduanya berada pada kategori masing-masing
42 merged_data['is_others_flag'] = (merged_data['category'] == 'Others').astype(int)
43
44 #Langkah 9
45 #Sort: Menampilkan urutan data selisih penjualan product dari penjualan yang mengalami penurunan paling besar
46 result_comparison2 = merged_data.sort_values(by=['selisih_penjualan_2021vs2022', 'is_others_flag', 'category'], ascending=[True, True, True])
47
48 #Langkah 10
49 #menghapus kolom dummy "is_others_flag" karena hanya digunakan untuk membantu menampilkan category di urutan pertama
50 result_comparison2 = result_comparison2.drop(columns=['is_others_flag']).reset_index(drop=True)
51
52 #Langkah 11
53 #Setting tampilan
54 result_comparison2_display = result_comparison2[['category', 'qty_ordered_2021', 'qty_ordered_2022', 'selisih_penjualan_2021vs2022']]
55
56 pd.set_option('display.max_columns', None)          #memberitahu panda untuk menampilkan semua kolom dataframe
57 pd.set_option('display.width', 1000)               #syarat menampilkan karakter pada row 1000 karakter, supaya tidak tertutup
58 pd.set_option('display.max_colwidth', None)         #mengatur lebar kolom sesuai isi karakter
59 pd.set_option('display.expand_frame_repr', False)  #mencegah data yang tampil melintang melewati beberapa baris
```

Jawaban No 2.1

Query + Penjelasannya :

```
61 # Merapihkan tampilan category rata kiri
62 styled_df = result_comparison2_display.style.set_properties(
63     subset=['category'],
64     **{'text-align': 'left'}
65 )
66
67 #Langkah 12
68 #Menampilkan hasil
69 print("Perbandingan data penjualan semua kategori tahun 2021 vs 2022:")
70 styled_df
```

```
1 #Langkah 13
2 #Menyimpan file ke CSV
3 # index=False mencegah Pandas menyimpan nomor indeks (0, 1, 2...) sebagai kolom terpisah
4 result_comparison2.to_csv('data_penjualan_Allcategory_2021vs2022.csv', index=False)
```

Jawaban No 2.1

Result:

Perbandingan data penjualan semua kategori tahun 2021 vs 2022:

	category	qty_ordered_2021	qty_ordered_2022	selisih_penjualan_2021vs2022
0	Others	426	263	-163
1	Soghaat	759	612	-147
2	Men Fashion	237	175	-62
3	Beauty & Grooming	168	153	-15
4	Appliances	124	148	24
5	Books	171	195	24
6	Health & Sports	173	200	27
7	Computing	109	153	44
8	School & Education	184	237	53
9	Home & Living	193	250	57
10	Kids & Baby	170	227	57
11	Entertainment	77	150	73
12	Superstore	327	536	209
13	Women Fashion	140	489	349
14	Mobiles & Tablets	107	1154	1047

Insight:

- Dugaan tim warehouse mengenai ketersediaan stock category orders yang masih banyak ada korelasinya dengan penurunan qty penjualan memang benar adanya.
- Dari data di samping di dapatkan bahwa product dengan category others mengalami penurunan penjualan yang sangat signifikan dalam sampai di angka tertinggi -163 pcs (2021 (426) vs 2022 (263))
- Selain category others, didapati 3 kategori lain yang mengalami penurunan penjualan yang signifikan berupa product dari kategori Soghaat (-147 pcs), Men Fashion (-62 pcs), Beauty & Grooming (-15 pcs).
- Terjadi pergeseran peningkatan penjualan yang signifikan pada segmen category lain berupa Superstore (209 pcs), Women Fashion (349 pcs), dan Mobiles & Tablets (1047 pcs).

Rekomendasi:

- Perlu dilakukan breakdown item product pada category others dan soghaat untuk melihat item product yang mana yang mengalami penurunan penjualan yang paling ekstrim
- Lakukan riset pasar mengenai item product yang mengalami penurunan penjualan tersebut, apakah ada perubahan behaviour pada konsumen item-item tersebut berupa harga yang kompetitif dengan pesaing atau model yang ketinggalan trend atau perubahan fungsi produk, dan yang semisal.
- Tentukan solusi dan inovasi yang tepat setelah data-data riset terkumpul

Jawaban No 2.2

Query + Penjelasannya :

```
1 #langkah 1 mengimport pandas as pd dan numpy as np sudah dilakukan di atas (kotak dialog no 6)
2 #langkah 2 mengubah format kolom order_date bertipe datetime sudah dilakukan di atas (kotak dialog no 17)
3
4 #langkah 3 Filter data dengan valid = 1 dan kategori Others
5 # To compare 2021 and 2022, it's best to filter data for each year separately.
6 filter_others_2021 = df[
7     (df['category'] == 'Others') &          #memfilter data dengan kategori Others
8     (df['is_valid'] == 1) &                  #memfilter data dengan pembelian min 1 kali
9     (df['order_date'].dt.year == 2021)        #memfilter data hanya pada tahun 2021
10 ].copy()
11
12 filter_others_2022 = df[
13     (df['category'] == 'Others') &          #memfilter data dengan kategori Others
14     (df['is_valid'] == 1) &                  #memfilter data dengan pembelian min 1 kali
15     (df['order_date'].dt.year == 2022)        #memfilter data hanya pada tahun 2022
16 ].copy()
```

Jawaban No 2.2

Query + Penjelasannya :

```
18 #langkah 4
19 #grouping based on sku_name dan tahun 2021
20 Product_item_2021 = (
21     filter_others_2021
22     .groupby(['category', 'sku_name'], as_index=False) #grouping based on category, sku_name (object type)
23     .agg(qty_ordered_2021=('qty_ordered', 'sum')) #fungsi aggregat sum terhadap qty_ordered dan merubah nama kolom menjadi qty_order
24     .sort_values(by='qty_ordered_2021', ascending=False) #mengurutkan data based on qty_ordered_2021 secara descending (dari besar ke kecil)
25     .rename(columns={'sku_name': 'Others_product_item'}) #mengubah nama kolom dataframe
26 )
27
28 #langkah 5
29 #grouping based on sku_name dan tahun 2022
30 Product_item_2022 = (
31     filter_others_2022
32     .groupby(['category', 'sku_name'], as_index=False) #grouping based on category, sku_name (object type)
33     .agg(qty_ordered_2022=('qty_ordered', 'sum')) # Corrected agg syntax and renamed directly
34     .sort_values(by='qty_ordered_2022', ascending=False) #mengurutkan data based on qty_ordered_2022 secara descending (dari besar ke kecil)
35     .rename(columns={'sku_name': 'Others_product_item'}) #mengubah nama kolom dataframe
36 )
37
38 #langkah 6
39 #Merge data grouping penjualan 2021 dan 2022
40 merged_data = pd.merge(Product_item_2021, Product_item_2022, on='Others_product_item', how='outer')
41 merged_data = merged_data.fillna(0).sort_values(by='qty_ordered_2021', ascending=False)
42
43 #Langkah 7
44 #lakukan pengurangan qty 2021 terhadap qty 2022
45 merged_data['Selisih_penjualan_others'] = merged_data['qty_ordered_2022'] - merged_data['qty_ordered_2021']
```

Jawaban No 2.2

Query + Penjelasannya :

```
47 #Langkah 8
48 #Buat dummy varibale flag Others untuk memastikan join product 2021 vs 2022 keduanya berada pada kategori Others
49 merged_data['is_Others'] = ((merged_data['category_x'] == 'Others') | (merged_data['category_y'] == 'Others')).astype(int)
50
51 #Langkah 9
52 #Sort: Menampilkan urutan data selisih penjualan product dari penjualan yang mengalami penurunan paling besar
53 result_comparison = merged_data.sort_values(by=['is_Others', 'Selisih_penjualan_others'], ascending=[False, True])
54
55 #Langkah 10
56 #menghapus kolom dummy "is_other" karena hanya digunakan untuk membantu menampilkan category others di urutan pertama
57 result_comparison = result_comparison.drop(columns=['is_Others']).reset_index(drop=True)
58
59 #menampilkan selisih penjualan yang minus saja
60 result_comparison = result_comparison[result_comparison['Selisih_penjualan_others'] < 0]
61
62 # Batasi ke 20 baris teratas setelah pemfilteran
63 result_comparison = result_comparison.head(20)
64
65 #Langkah 11
66 #Setting tampilan
67 pd.set_option('display.max_columns', None)          #memberitahu panda untuk menampilkan semua kolom dataframe
68 pd.set_option('display.width', 1000)                #syarat menampilkan karakter pada row 1000 karakter, supaya tidak tertutup
69 pd.set_option('display.max_colwidth', None)          #mengatur lebar kolom sesuai isi karakter
70 pd.set_option('display.expand_frame_repr', False)   #mencegah data yang tampil melintang melewati beberapa baris
71
72 column_to_move = 'Others_product_item'
```

Jawaban No 2.2

Query + Penjelasannya :

```
74 # Buat daftar kolom baru: kolom yang dipindahkan di posisi pertama, diikuti semua kolom lainnya kecuali yang dipindahkan
75 new_column_order = [column_to_move] + [col for col in result_comparison.columns if col != column_to_move]
76
77 # Terapkan urutan kolom baru ke DataFrame
78 result_comparison = result_comparison[new_column_order]
79
80 # Menghapus kolom 'category_x' dan 'category_y' secara permanen
81 #result_comparison = result_comparison.drop(columns=['category_x', 'category_y'])
82
83 # Merapihkan tampilan Other_product_item rata kiri
84 styled_df = result_comparison.style.set_properties(
85     subset=['Others_product_item'],
86     **{'text-align': 'left'}
87 )
88
89 #Langkah 12
90 #Menampilkan hasil
91 print("Top_20_Product_decreases_others_category_sales_quantity")
92 styled_df
```



Jawaban No 2.2

Result:

Top_20_Product_decreases_others_category_sales_quantity						
	Others_product_item	category_x	qty_ordered_2021	category_y	qty_ordered_2022	Selisih_penjualan_others
0	RB_Dettol Germ Busting Kit-bf	Others	200.000000	Others	45.000000	-155.000000
1	Dawlance_MD 10 + DWB 600	Others	23.000000	0	0.000000	-23.000000
2	Telemall_MM-DR-HB-L	Others	23.000000	Others	2.000000	-21.000000
3	iu_Tickets General Enclosure-Islamabad	Others	20.000000	0	0.000000	-20.000000
4	RS_Rehmat-e-Shereen Mix Mithai	Others	13.000000	0	0.000000	-13.000000
5	kansai_NeverWet	Others	10.000000	Others	1.000000	-9.000000
6	sindbad_Sindbad Gift Card-3	Others	7.000000	0	0.000000	-7.000000
7	emart_00-1	Others	7.000000	Others	1.000000	-6.000000
8	Vouch 365 2016	Others	5.000000	0	0.000000	-5.000000
9	Am-PTV_ATS-004-M	Others	5.000000	0	0.000000	-5.000000
10	duma_4561253300294	Others	4.000000	0	0.000000	-4.000000
11	sockeye_QG in Quarter Grey	Others	4.000000	0	0.000000	-4.000000
12	The Vitamin Company Kojic Acid Whitening Cream 40GM	Others	4.000000	0	0.000000	-4.000000
13	MEGUIAR_G12711	Others	4.000000	Others	1.000000	-3.000000
14	Trans2_LW 999	Others	3.000000	0	0.000000	-3.000000
15	aw_Octane Booster-12oz./354ml	Others	3.000000	0	0.000000	-3.000000
16	aw_CRC DUST FREE -400ml	Others	2.000000	0	0.000000	-2.000000
17	MEGUIAR_X1030EU	Others	2.000000	0	0.000000	-2.000000
18	MEGUIAR_G19216	Others	2.000000	0	0.000000	-2.000000
19	MEGUIAR_G13616	Others	2.000000	0	0.000000	-2.000000

Insight:

- Terdapat 76 Product item dengan category others yang mengalami penurunan penjualan yang sangat signifikan
- RB Dettol Germ Busting Kit-bf menjadi item penyumbang terbanyak sebanyak -155 pcs, hal ini bisa disebabkan product paket kebersihan dari dettol Reckitt Benckiser (Hand sanitizer, sabun, handwash) tidak dibeli lagi oleh konsumen pada 2022 karena Pandemi sudah mulai berangsor berakhir, sehingga product-product ini wajar mengalami penurunan
- Dari Top 20 Decreases item product category others 75% product pada tahun 2022 sama sekali tidak terjadi pembelian, dimana sebelumnya pada 2021 terdapat pembelian 2-23 pcs/item

Rekomendasi :

- Melakukan startegi pemberian diskon kepada para konsumen, hal ini dimaksudkan untuk meningkatkan penjualan di season-season selanjutnya
- mengiklankan produk-produk tersebut secara masif baik media cetak, sosmed, dll. bertujuan untuk memperkenalkan atau meng influence masyarakat luas akan produk yang ditawarkan
- mengandeng influencer dalam pengiklanan guna mengajakan menggunakan produk-produk yang akan dijual

Client Request 3

Dear Data Analyst,

Terkait ulang tahun perusahaan pada 2 bulan mendatang, Tim Digital Marketing akan memberikan informasi_promo_bagipelanggan pada akhir bulan ini. Kriteria pelanggan yang akan kami butuhkan adalah mereka yang sudah melakukan check-out namun belum melakukan pembayaran (is_gross = 1) selama tahun 2022. Data yang kami butuhkan adalah ID Customer dan Registered Date.

Mohon bantuan, untuk mengirimkan data tersebut sebelum akhir bulan ini ke Tim Digital Marketing. Atas bantuan yang diberikan, kami mengucapkan terima kasih.

Regards
Tim Digital Marketing

Jawaban No 3

Query + Penjelasannya :

```
1 #Filter data sesuai kriteria:  
2 customers_for_promo = df[  
3     (df['is_gross'] == 1) &      # Filter based on is_gross = 1 (belum membayar)  
4     (df['is_valid'] == 0) &      # Filter based on is_valid = 0 (belum melakukan pembayaran)  
5     (df['is_net'] == 0) &      # Filter based on is_net = 0 (transaksi belum selesai)  
6     (df['order_date'].dt.year == 2022) # Filter based on registered_date = 2022 saja  
7 ].copy()  
8  
9 # Select unique customer_id and registered_date & delete duplicate  
10 customers_for_promo = customers_for_promo[['customer_id', 'registered_date']].drop_duplicates()  
11  
12 # Rename columns for clarity  
13 customers_for_promo.columns = ['ID Customer', 'Registered Date']  
14  
15 # Urutkan sesuai ID Customer  
16 customers_for_promo = customers_for_promo.sort_values(by='ID Customer') #Urutkan data sesuai ID Customer  
17 #memastikan indeks kolom lama tidak ditambahkan sebagai kolom baru di dataframe  
18 customers_for_promo = customers_for_promo.reset_index(drop=True)  
19 #insert kolom baru untuk indexing nomor secara urut dari 1 - ....  
20 customers_for_promo.insert(0, 'No', range(1, len(customers_for_promo) + 1))  
22 # Menampilkan hasil  
23 print("Daftar Pelanggan yang sudah melakukan check-out namun belum membayar di tahun 2022:")  
24 print(customers_for_promo.to_string(index=False))
```

Jawaban No 3

Result:

Daftar Pelanggan yang sudah melakukan check-out namun belum membayar di tahun 2022:

No	ID	Customer Registered Date
1	C107850L	2022-08-03
2	C110122L	2022-08-14
3	C114766L	2022-01-28
4	C115342L	2022-06-17
5	C115470L	2022-06-11
6	C116160L	2022-05-05
7	C117258L	2022-08-01
8	C117628L	2022-05-16
9	C118563L	2022-06-07
10	C120098L	2022-05-21
11	C125637L	2022-04-05
12	C127081L	2022-04-22
807	C988571L	2022-06-17
808	C989951L	2021-11-25
809	C990726L	2022-03-16
810	C992621L	2021-11-17
811	C993531L	2021-11-08
812	C993996L	2022-03-22
813	C994155L	2022-01-27
814	C994467L	2022-03-15
815	C995583L	2022-03-02
816	C995774L	2022-02-18
817	C995819L	2022-07-21
818	C998017L	2021-11-14
819	C998847L	2022-07-03
820	C999472L	2021-07-20

Insight:

- Sebanyak 820 pelanggan yang sudah melakukan check out tapi belum membayar di tahun 2022.
- Pelanggan ini sudah menunjukkan minat beli tetapi ada hambatan/kendala sehingga tidak menyelesaikan transaksi
- segmentasi pelanggan berdasarkan pendaftaran pelanggan baru dan pelanggan lama beberapa pelanggan mungkin baru saja pertama kali check out

Rekomendasi:

- pelanggan baru bisa di beri diskon first purchase (pembelian pertama)
- pelanggan lama bisa diberi reward berupa diskon ongkir atau voucher promo pembelian untuk menarik minat beli
- analisa hambatan pembayaran dari metode pembayarannya, kesalahan sistem check out atau karena promo tidak menarik

Client Request 4

Dear Data Analyst,

Pada bulan October hingga Desember 2022, kami melakukan campaign setiap hari Sabtu dan Minggu. Kami hendak menilai, apakah campaign tersebut cukup berdampak pada kenaikan penjualan (before discount). Mohon bantuan untuk menampilkan data:

1. Rata-rata harian penjualan weekends (Sabtu dan Minggu) vs rata-rata harian penjualan weekdays (Senin-Jumat) per bulan tersebut. Apakah ada peningkatan penjualan pada masing-masing bulan tersebut.
2. Rata-rata harian penjualan weekends (Sabtu dan Minggu) vs rata-rata harian penjualan weekdays (Senin-Jumat) keseluruhan 3 bulan tersebut.

Mohon bantuan untuk mengirimkan data tersebut paling lambat minggu depan. Atas bantuan yang diberikan, kami mengucapkan terima kasih.

Regards
Tim Campaign

Jawaban No 4.1

Query + Penjelasannya :

```
1 #1 Membuat kolom baru untuk day, month, dan month number
2 df['day'] = df['order_date'].dt.day_name()      #Mengekstrak nama hari pada data order_date dan membuat kolom baru dengan nama day
3 df['month'] = df['order_date'].dt.month_name()    #Mengekstrak nama bulan pada data order_date dan membuat kolom baru dengan nama month
4 df['month_number'] = df['order_date'].dt.month   #Mengekstrak bulan angka pada data order_date dan membuat kolom baru dengan nama month_
5
6 #2 Filter data dengan kondisi
7 df_valid = df[df['is_valid'] == 1]  #transaksi yang sudah pembayaran is_valid = 1
8
9 weekends = df_valid[
10    df_valid['day'].isin(['Saturday', 'Sunday'])] #weekends
11
12 weekdays = df_valid[
13    df_valid['day'].isin([
14        'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'])] #weekdays
15
16 #3 Filter data transaksi dari Oktober - Desember
17 weekends_q4 = weekends[
18    (weekends['order_date'].dt.year == 2022) &      #filter data tahun 2022 saja pada data weekend order_date
19    (weekends['month_number'].isin([10, 11, 12]))] #filter data bulan 10,11,12 saja pada data weekend month_number
20
21 weekdays_q4 = weekdays[
22    (weekdays['order_date'].dt.year == 2022) &      #filter data tahun 2022 saja pada data weekday order_date
23    (weekdays['month_number'].isin([10, 11, 12]))] #filter data bulan 10,11,12 saja pada data weekday month_number
24
25 # Hitung rata-rata harian penjualan weekends (Sabtu dan Minggu) per bulan
26 weekend_monthly = (
27    weekends_q4
28    .groupby('month')['before_discount']
29    .mean()
30    .reset_index(name='avg_weekend_sales'))
31
32 # Hitung rata-rata harian penjualan weekdays (Senin-Jumat) per bulan
33 weekday_monthly = (
34    weekdays_q4
35    .groupby('month')['before_discount']
36    .mean()
37    .reset_index(name='avg_weekday_sales'))
```

Jawaban No 4.1

Query + Penjelasannya :

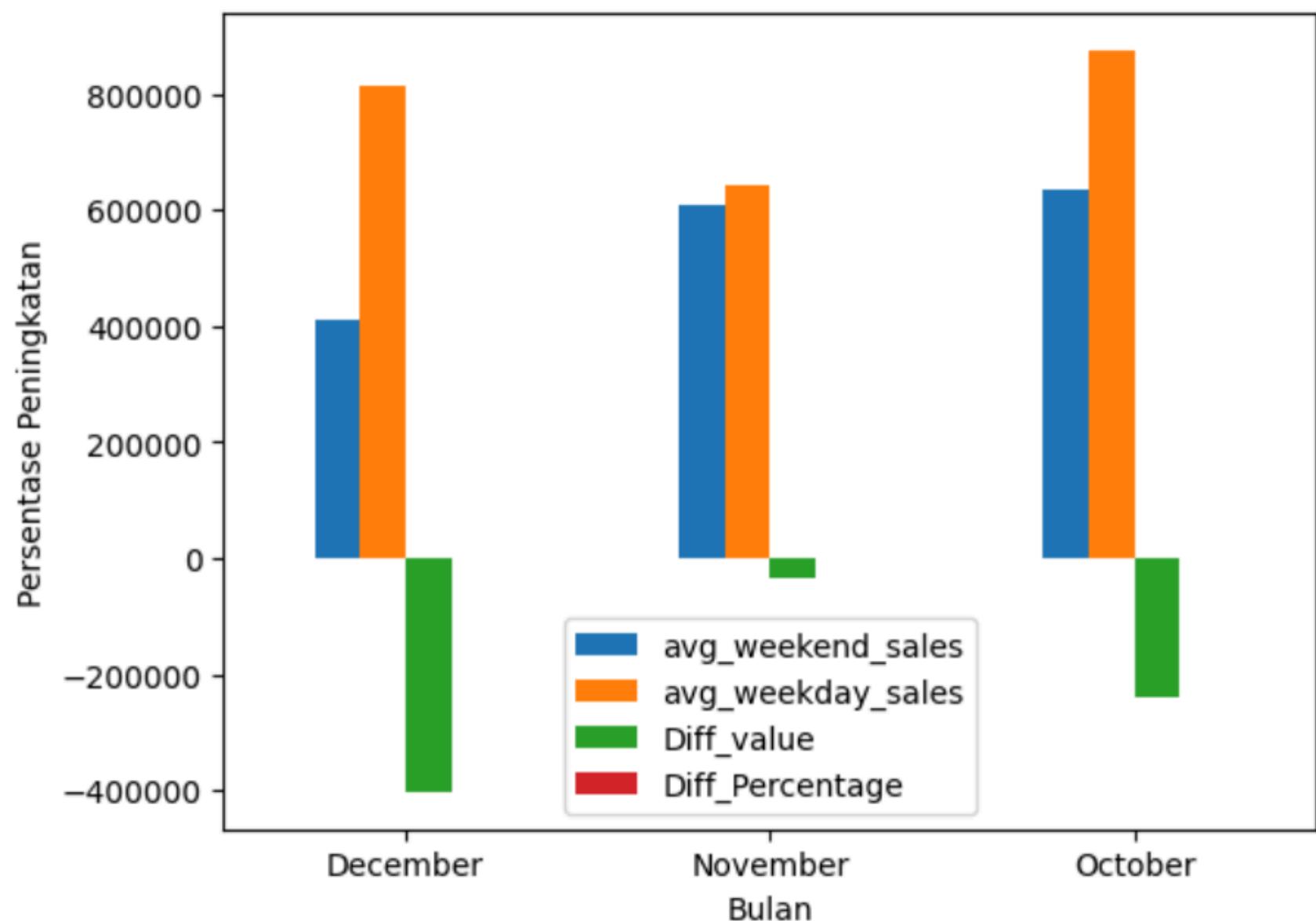
```
42 # Menghitung Selisih Peningkatan/Penurunan
43 monthly_comparison['Diff_value'] = (
44     (monthly_comparison['avg_weekend_sales'] - monthly_comparison['avg_weekday_sales']))
45
46 # Menghitung Persentase Peningkatan/Penurunan
47 monthly_comparison['Diff_Percentage'] = (
48     (monthly_comparison['avg_weekend_sales'] - monthly_comparison['avg_weekday_sales']) / monthly_comparison['avg_weekday_sales'] * 100)
49
50 # Menampilkan hasil
51 monthly_comparison.round(2)
52
53 #Grafik per bulan
54 monthly_comparison.plot(kind='bar', x='month')
55
56 plt.title('Perbandingan rata-rata harian penjualan\n'
57 'weekends vs weekdays (Oktober - Desember 2022)\n')
58 plt.xlabel('Bulan')
59 plt.ylabel('Persentase Peningkatan')
60 plt.xticks(rotation=0)
61 plt.show()
```

Jawaban No 4.1

Result:

	month	avg_weekend_sales	avg_weekday_sales	Diff_value	Diff_Percentage
0	December	410599.40	813574.29	-402974.89	-49.53
1	November	607794.21	641862.00	-34067.79	-5.31
2	October	634260.07	874690.27	-240430.19	-27.49

Perbandingan rata-rata harian penjualan weekends vs weekdays (Oktober - Desember 2022)



Insight:

- Bulan Oktober: Penurunan sebesar -27.49%**
 - Rata-rata penjualan harian saat weekend masih tertinggal cukup jauh dibanding weekday. Hal ini menunjukkan bahwa campaign awal belum cukup kuat menarik minat belanja pelanggan di akhir pekan.
- Bulan November: Penurunan sebesar -5.31% (paling kecil)**
 - Gap antara weekend dan weekday menjadi sangat kecil. Ini merupakan sinyal positif, karena menunjukkan adanya perbaikan performa campaign dibanding bulan sebelumnya.
- Bulan Desember: Penurunan sebesar -49.53% (paling besar)**
 - Campaign weekend tidak menjadi pilihan utama pelanggan di momen belanja besar.

Rekomendasi:

Berdasarkan data, penjualan pada hari kerja masih menjadi kontributor terbesar dibandingkan akhir pekan. Karena itu, fokus campaign sebaiknya tidak hanya terbatas pada weekend saja. Perusahaan bisa mulai mencoba strategi campaign hybrid dengan mengombinasikan aktivitas promosi di weekday dan weekend agar potensi penjualan bisa lebih optimal. Selain itu, campaign di akhir pekan juga perlu memiliki nilai tambah yang lebih kuat. Promo weekend sebaiknya dibuat lebih menarik, misalnya dengan diskon yang lebih besar, penawaran yang lebih eksklusif, serta batas waktu yang jelas agar mendorong rasa urgensi dan meningkatkan minat beli konsumen.

Jawaban No 4.2

Query + Penjelasannya :

▶ # Membuat tabel ringkasan sederhana agar mudah dibaca
overall_comparison = pd.DataFrame({
 'type': ['Weekdays', 'Weekends'],
 'avg_daily_sales': [
 weekdays_q4['before_discount'].mean(), # Mengambil kolom before_discount dari data weekdays & menghitung rata-rata harian selama 3 bulan
 weekends_q4['before_discount'].mean()]) # Mengambil kolom before_discount dari data weekend & menghitung rata-rata harian selama 3 bulan

Menampilkan hasil
overall_comparison

▶ # Grafik keseluruhan 3 Bulan

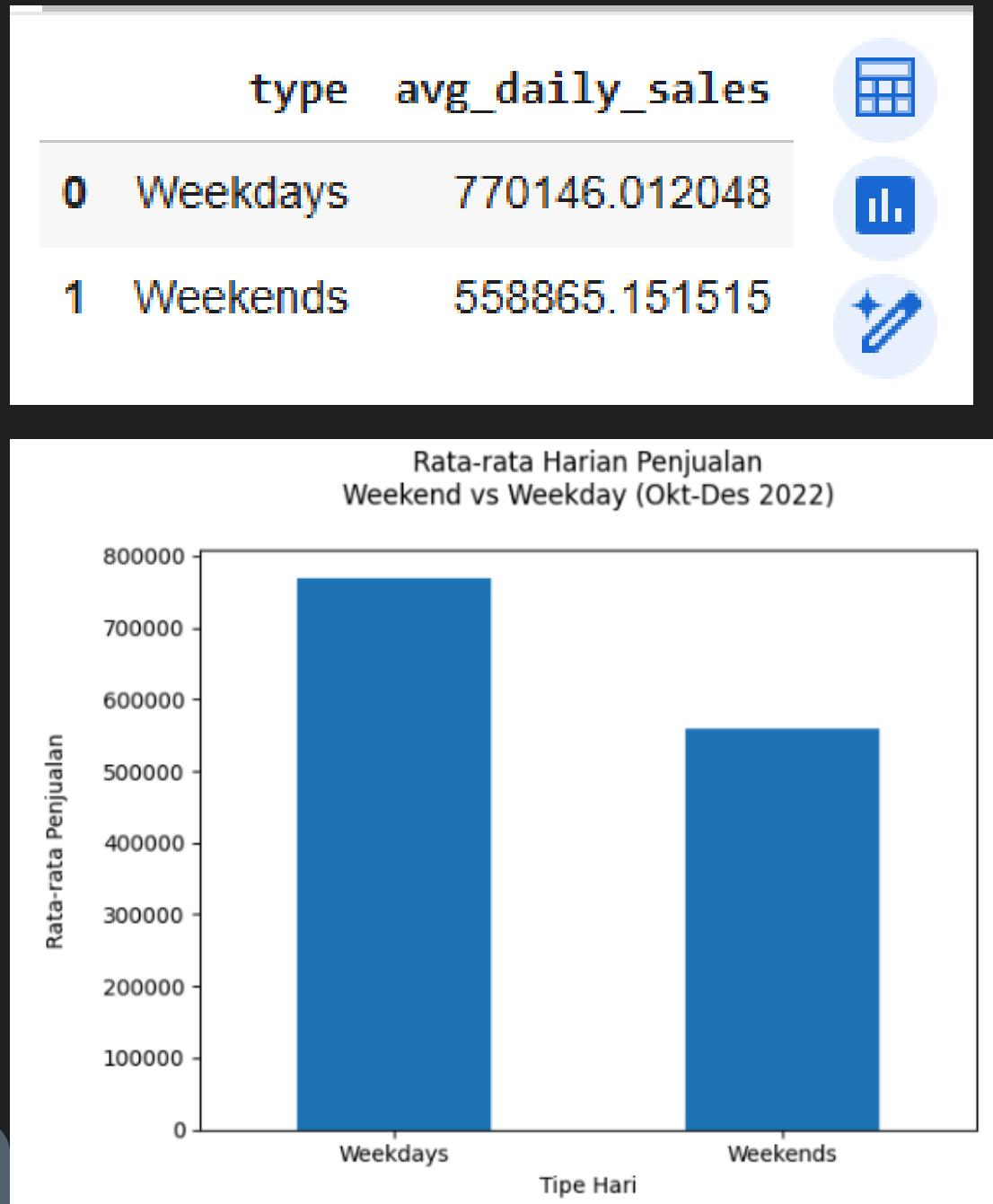
Perintahkan pandas untuk membuat grafik dari df overall_comparison
overall_comparison.plot(
 x='type', # Menentukan sumbu X berisi kategori weekday & weekend
 y='avg_daily_sales', # Menentukan sumbu Y berisi nilai rata-rata penjualan harian
 kind='bar', # Menentukan jenis grafiknya, yaitu bar chart
 legend=False) # Menyembunyikan legenda

plt.title(# Memberi judul grafik
 'Rata-rata Harian Penjualan\n' # Memberi label sumbu X
 'Weekend vs Weekday (Okt-Des 2022)\n') # Memberi label sumbu Y
plt.xlabel('Tipe Hari\n') # Mengatur label sumbu X
plt.ylabel('Rata-rata Penjualan\n')
plt.xticks(rotation=0)

Menampilkan grafik
plt.show()

Jawaban No 4.2

Result:



Insight:

Weekdays: rata-rata harian penjualan sebesar 770.146

- Angka ini menunjukkan bahwa hari kerja merupakan periode belanja utama bagi pelanggan. Di weekday, pelanggan cenderung lebih siap secara finansial dan lebih aktif dalam menyelesaikan transaksi. Bahkan tanpa campaign khusus sekalipun, penjualan tetap bisa berjalan tinggi.

Weekends: rata-rata harian penjualan sebesar 558.865

- Angka ini menunjukkan bahwa aktivitas belanja tetap terjadi di weekend, namun nilainya belum maksimal. Pelanggan memang masih bertransaksi, tetapi jumlah transaksinya lebih sedikit atau nilai belanjanya cenderung lebih kecil. Artinya, campaign weekend sudah berhasil menarik perhatian pelanggan, namun belum cukup kuat untuk mendorong pembelian dengan nilai yang lebih besar.

Rekomendasi:

- Weekday bisa dimanfaatkan untuk menjalankan promo dengan nilai yang lebih besar, seperti penawaran produk high value, bundling premium, atau campaign dengan volume besar. Di saat yang sama, weekday juga dapat digunakan untuk mengarahkan traffic ke weekend, misalnya dengan memberikan teaser promo di hari Jumat atau membagikan voucher yang baru aktif pada Sabtu–Minggu.
- Di weekend, tambahkan unsur urgensi seperti waktu atau kuota promo yang terbatas, dan buat penawaran yang benar-benar berbeda dari weekday agar terasa lebih spesial. Selain itu, penting juga untuk mendorong penyelesaian transaksi, misalnya dengan mengirim reminder pembayaran atau memberikan promo khusus bagi pelanggan yang menyelesaikan checkout.

THANK YOU