MySkill
Python Showcase

By : Immanuel Mayerd

01

Tentang Dataset

Dataset

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 product 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

02

Memproses Data

Memproses Data

```
[ ] #Sumber data yang digunakan
    path_od = "https://raw.githubusercontent.com/dataskillsboost/FinalProjectDA11/main/order_detail.csv"
    path_pd = "https://raw.githubusercontent.com/dataskillsboost/FinalProjectDA11/main/payment_detail.csv"
    path_cd = "https://raw.githubusercontent.com/dataskillsboost/FinalProjectDA11/main/customer_detail.csv"
    path_sd = "https://raw.githubusercontent.com/dataskillsboost/FinalProjectDA11/main/sku_detail.csv"
    df_od = pd.read_csv(path_od)
    df_pd = pd.read_csv(path_pd)
    df_cd = pd.read_csv(path_cd)
    df_sd = pd.read_csv(path_sd)
```

```
[ ] #Mengampilkan 5 baris pertama
    df_od.head()
```

	id	customer_id	order_date	sku_id	price	qty_ordered	before_discount	discount_amount	after_discount	is_gross	is_valid	is_net	payment_id
0	ODR9939707760w	C713589L	2021-11-19	P858068	26100	200	5220000.0	2610000.00	2610000.00	1	1	0	5
1	ODR7448356649d	C551551L	2021-11-19	P886455	1971942	5	9859710.0	2464927.50	7394782.50	1	0	0	5
2	ODR4011281866z	C685596L	2021-11-25	P678648	7482000	1	7482000.0	2065344.62	5416655.38	1	0	0	4
3	ODR3378927994s	C830683L	2021-11-22	P540013	3593680	1	3593680.0	1455440.40	2138239.60	1	1	1	5
4	ODR4904430099k	C191766L	2021-11-21	P491032	4413220	1	4413220.0	1059172.80	3354047.20	1	1	1	4

```
[ ] #Mengampilkan 5 baris pertama
    df_pd.head()
```

	id	payment_method
0	1	cod
1	2	jazzvoucher
2	3	customercredit
3	4	Payaxis
4	5	jazzwallet

Memproses Data

```
[ ] #Mengampilkan 5 baris pertama
    df_cd.head()
```

	id	registered_date
0	C996508L	2021-07-10
1	C180415L	2021-07-18
2	C535451L	2021-07-23
3	C177843L	2021-07-12
4	C951682L	2021-07-27

```
    #Mengampilkan 5 baris pertama
    df_sd.head()
```

	id	sku_name	base_price	cogs	category
0	P798444	AT-FSM-35	57631.70	46052	Kids & Baby
1	P938347	AYS_Haier-18HNF	3931789.26	3499256	Appliances
2	P826364	Atalian _DV206A-Brown-41	324597.00	243426	Men Fashion
3	P467533	Darul_Sakoon_Food_Bundle	2870.42	2378	Superstore
4	P229955	HP_15AY-15-Ay072NIA-ci3	2265625.00	1631250	Computing

```
[ ] #Menjalankan SQL di Colab
    from sqlite3 import connect
    conn = connect(':memory:')
    df_od.to_sql('order_detail',conn, index=False, if_exists='replace')
    df_pd.to_sql('payment_detail', conn, index=False, if_exists='replace')
    df_sd.to_sql('sku_detail', conn, index=False, if_exists='replace')
    df_cd.to_sql('customer_detail', conn, index=False, if_exists='replace')

    3998
```

```
[ ] #Query SQL untuk menggabungkan data
    df = pd.read_sql("""
    SELECT
        order_detail.*,
        payment_detail.payment_method,
        sku_detail.sku_name,
        sku_detail.base_price,
        sku_detail.cogs,
        sku_detail.category,
        customer_detail.registered_date
    FROM order_detail
    LEFT JOIN payment_detail
        on payment_detail.id = order_detail.payment_id
    LEFT JOIN sku_detail
        on sku_detail.id = order_detail.sku_id
    LEFT JOIN customer_detail
        on customer_detail.id = order_detail.customer_id
    """, conn)
```

```
[ ] #Mengampilkan 5 baris pertama
    df.head()
```

	id	customer_id	order_date	sku_id	price	qty_ordered	before_discount	discount_amount	after_discount	is_gross	is_valid	is_net	payment_id	payment_method	sku_name	base_price	cogs	category	registered_date
0	ODR9939707760w	C713589L	2021-11-19	P858068	26100	200	5220000.0	2610000.00	2610000.00	1	1	0	5	jazzwallet	RB_Dettol Germ Busting Kit-bf	26100.0	18270	Others	2021-07-07
1	ODR7448356649d	C551551L	2021-11-19	P886455	1971942	5	9859710.0	2464927.50	7394782.50	1	0	0	5	jazzwallet	PS4_Slim-500GB	1971942.0	1321182	Entertainment	2021-11-20
2	ODR4011281866z	C685596L	2021-11-25	P678648	7482000	1	7482000.0	2065344.62	5416655.38	1	0	0	4	Payaxis	Changhong Ruba 55 Inches UD55D6000i Ultra HD T...	7482000.0	5162580	Entertainment	2021-11-19
3	ODR3378927994s	C830683L	2021-11-22	P540013	3593680	1	3593680.0	1455440.40	2138239.60	1	1	1	5	jazzwallet	dawlance_Inverter 30	3593680.0	3054628	Appliances	2021-11-03
4	ODR4904430099k	C191766L	2021-11-21	P491032	4413220	1	4413220.0	1059172.80	3354047.20	1	1	1	4	Payaxis	Dawlance_Inverter-45 2.0 ton	4413220.0	3177472	Appliances	2021-07-05

Memproses Data

```
[ ] #Menampikan tipe data tiap kolom
    df.dtypes
```

```
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
```

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

```
id                 object
customer_id        object
order_date         object
sku_id             object
price               int64
qty_ordered         int64
before_discount     int64
discount_amount     int64
after_discount      int64
is_gross            int64
is_valid            int64
is_net              int64
payment_id          int64
payment_method     object
sku_name           object
base_price          int64
cogs                int64
category           object
registered_date    object
dtype: object
```

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

```
id                         object
customer_id                object
order_date         datetime64[ns]
sku_id                     object
price                       int64
qty_ordered                 int64
before_discount             int64
discount_amount             int64
after_discount              int64
is_gross                    int64
is_valid                    int64
is_net                      int64
payment_id                  int64
payment_method             object
sku_name                   object
base_price                  int64
cogs                        int64
category                   object
registered_date    datetime64[ns]
dtype: object
```

03

Pertanyaan

Pertanyaan 1

No 1

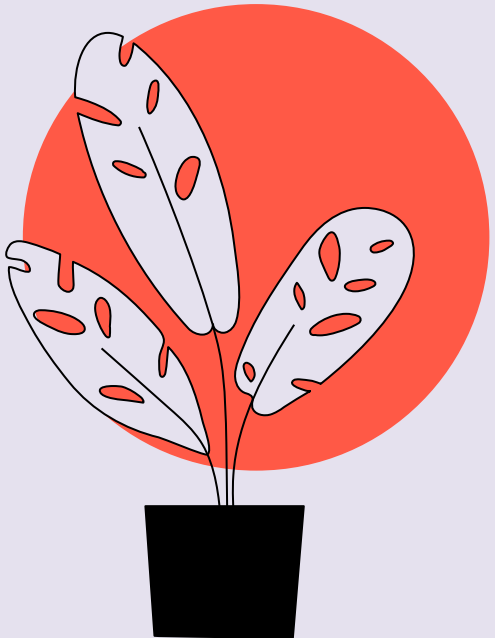
Dear Data Analyst,

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 1

```
[ ]  # Tulis kode Anda di bawah ini. Dapat menggunakan lebih dari 1 blok kode
     filtered_data = df[(df['is_valid'] == 1) & (df['category'] == 'Mobiles & Tablets') & (df['order_date'].dt.year == 2022)]
     grouped_data = filtered_data.groupby('sku_name')['qty_ordered'].sum().reset_index()
     sorted_data = grouped_data.sort_values(by='qty_ordered', ascending=False)
     top_5_products = sorted_data.head(5)

     # Menampilkan hasil
     print(top_5_products)
```

```
                        sku_name  qty_ordered
1              IDROID_BALRX7-Gold         1000
2         IDROID_BALRX7-Jet black           31
3              Infinix Hot 4-Gold           15
43  samsung_Grand Prime Plus-Black           11
34             infinix_Zero 4-Grey           10
```

Kode di atas mengambil DataFrame (df) untuk mendapatkan lima produk teratas dari kategori 'Mobiles & Tablets' dengan jumlah pesanan tertinggi pada tahun 2022. Langkah-langkahnya melibatkan filtering data, pengelompokan berdasarkan nama produk, pengurutan berdasarkan jumlah pesanan, dan akhirnya menampilkan hasilnya.

Pertanyaan 2

No 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.

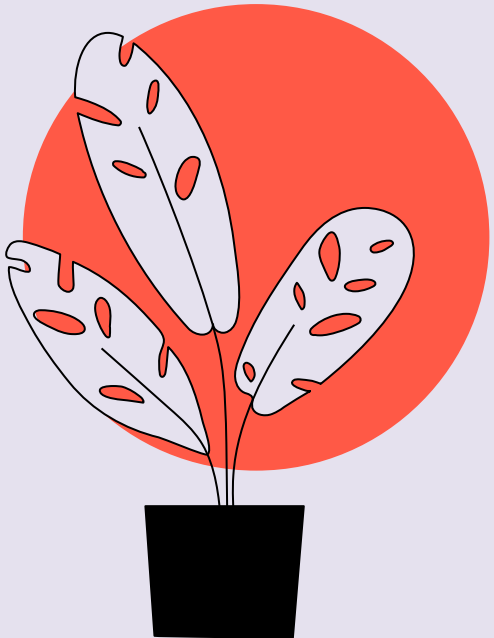
1. Kami mohon bantuan untuk melakukan pengecekan data penjualan kategori tersebut dengan tahun 2021 secara kuantitas penjualan. Dugaan sementara kami, telah terjadi penurunan kuantitas penjualan pada 2022 dibandingkan 2021. (Mohon juga menampilkan data ke-15 kategori)

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 2.1

```python
# Proses merge data
filtered_data_2021 = df[(df['is_valid'] == 1) & (df['order_date'].dt.year == 2021)]
filtered_data_2022 = df[(df['is_valid'] == 1) & (df['order_date'].dt.year == 2022)]

grouped_data_2021 = filtered_data_2021.groupby('category')['qty_ordered'].sum().reset_index()
grouped_data_2022 = filtered_data_2022.groupby('category')['qty_ordered'].sum().reset_index()

merged_data = pd.merge(grouped_data_2022, grouped_data_2021, on='category', how='left', suffixes=('_2022', '_2021'))
merged_data['qty_decrease'] = merged_data['qty_ordered_2022'] - merged_data['qty_ordered_2021']

# Menampilkan data ke-15 kategori
print('Tabel Perubahan Kuantitas kategori\n')
print(merged_data)
print('\n')
```

```
Tabel Perubahan Kuantitas kategori

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

Kode di atas mengambil DataFrame (df) untuk Mengecek apakah ada penurunan penjualan kategori ‘Others’ untuk tahun 2022 disbanding dengan 2021. Langkah-langkahnya melibatkan filtering data, pengelompokan berdasarkan kategori, kemudian menampilkan selisih order antara tahun 2022 dan 2021 kemudian didapatkan bahwa memang ada penurunan quantity ditandai dengan selisih kategori order 2022 dan 2021 sebesar -163.

Jawaban 2.2

```
[16] # Tulis kode Anda di bawah ini. Dapat menggunakan lebih dari 1 blok kode

     # Menampilkan TOP 20 nama produk yang mengalami penurunan paling tinggi
     others_2021 = df[(df['is_valid'] == 1) & (df['order_date'].dt.year == 2021) & (df['category'] == 'Others')]
     others_2022 = df[(df['is_valid'] == 1) & (df['order_date'].dt.year == 2022) & (df['category'] == 'Others')]

     sales_others_2021 = others_2021.groupby('sku_name')['qty_ordered'].sum().reset_index()
     sales_others_2022 = others_2022.groupby('sku_name')['qty_ordered'].sum().reset_index()

     merged_others = pd.merge(sales_others_2021, sales_others_2022, on='sku_name', how='left', suffixes=('_2021', '_2022'))
     merged_others['qty_decreased'] = merged_others['qty_ordered_2022'] - merged_others['qty_ordered_2021']

     top_20 = merged_others.sort_values(by='qty_decreased', ascending = True).head(20)
     print(top_20[['sku_name', 'qty_decreased']])
```

	sku_name	qty_decreased
39	RB_Dettol Germ Busting Kit-bf	-155.0
43	Telemall_MM-DR-HB-L	-21.0
73	kansai_NeverWet	-9.0
66	emart_00-1	-6.0
26	MEGUIAR_G12711	-3.0
1	Aladdin_bike_cover	-1.0
59	aw_Ultra Shine Wash & Wax-64oz./1893ml	-1.0
45	Tmall_MM-DR-PAD	-1.0
14	Entertainer Asia_Vouch 365-2017 Mobile App Lahore	-1.0
23	MEGUIAR_A1214	0.0
21	Lacie_9000146	0.0
63	electro_Humidifier	0.0
53	aw_CONSTRUCTION FOAM-700ml	1.0
13	Entertainer Asia_Vouch 365 - 2017 Book Karachi	1.0
30	MEGUIAR_G18211	1.0
86	vitamin_265	1.0
74	kansai_Undercoating Aerosol	3.0
65	emart_0-37	6.0
67	emart_Tyre Shape Air Compressor	29.0
0	Aladdin_Wrench_Snap N Grip_01	NaN

Kode di atas mengambil DataFrame (df) untuk Mengecek apakah ada penurunan penjualan sku_name pada kategori ‘Others’ untuk tahun 2022 disbanding dengan 2021. Langkah-langkahnya melibatkan filtering data, pengelompokan berdasarkan sku_name, kemudian menampilkan selisih order antara tahun 2022 dan 2021 dengan menampilkan top 20 produk dengan urutan menaik.

Pertanyaan 3

No 3

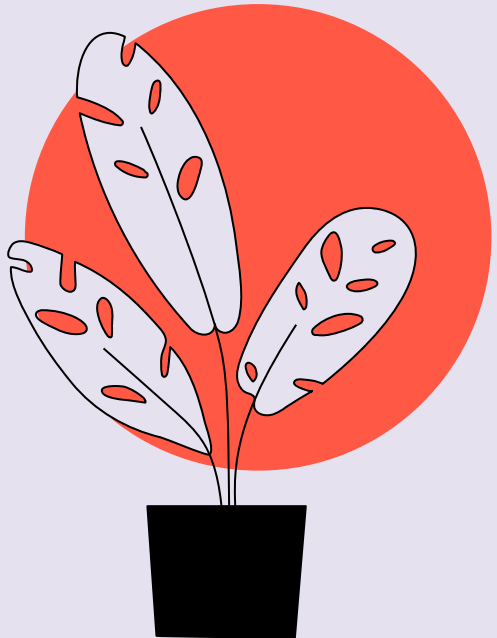
Dear Data Analyst,

Terkait ulang tahun perusahaan pada 2 bulan mendatang, Tim Digital Marketing akan memberikan informasi promo bagi pelanggan 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 3

```
[17] # Tulis kode Anda di bawah ini. Dapat menggunakan lebih dari 1 blok kode
     filtered_data = df[(df['is_gross'] == 1) & (df['is_valid'] == 0) & (df['is_net'] == 0) & (df['order_date'].dt.year == 2022)]

     # Memilih kolom yang dibutuhkan
     result_data = filtered_data[['customer_id', 'registered_date']]

     # Menampilkan hasil
     print(result_data)
```

```
     customer_id registered_date
9        C246762L      2022-05-08
18       C848774L      2021-11-07
19       C693415L      2022-04-12
21       C180595L      2022-04-22
22       C587425L      2022-03-22
...           ...             ...
5856     C394076L      2021-10-12
5859     C248585L      2022-07-10
5865     C471304L      2022-05-13
5881     C265450L      2022-02-17
5883     C676393L      2021-07-27

[1052 rows x 2 columns]
```

```
[18] #Jalankan kode ini untuk mendownload file
     from google.colab import files
     result_data.to_csv('audience_list.csv', encoding = 'utf-8-sig',index=False)
     files.download('audience_list.csv')
```

Kode di atas mengambil DataFrame (df) untuk memfilter data berdasarkan kondisi yang dibutuhkan tim digital marketing. Kemudian data tersebut diconvert ke dalam bentuk csv untuk dapat dikirim ke tim Digital Marketing dalam bentuk csv.

Pertanyaan 4

No 4

Dear Data Analyst,

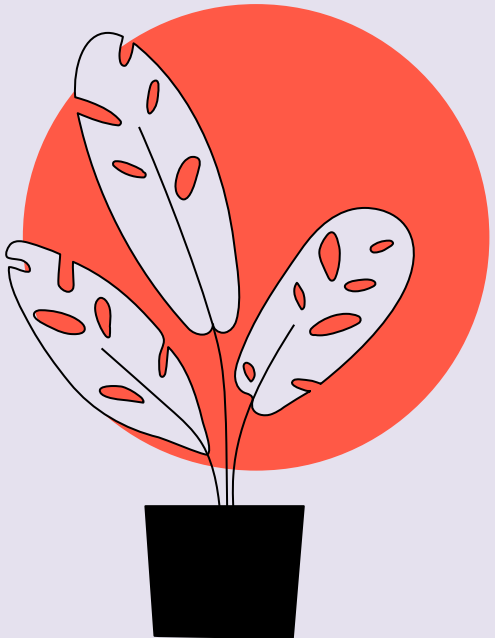
Pada bulan October hingga Desember 2022, kami melakukan campaign setiap hari Sabtu dan Minggu. Kami hendak menilai, apakah campaign tersebut cukup berdampak pada kenaikan penjualan (before_discount). Mohon bantuan untuk menampilkan data:

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 4.1

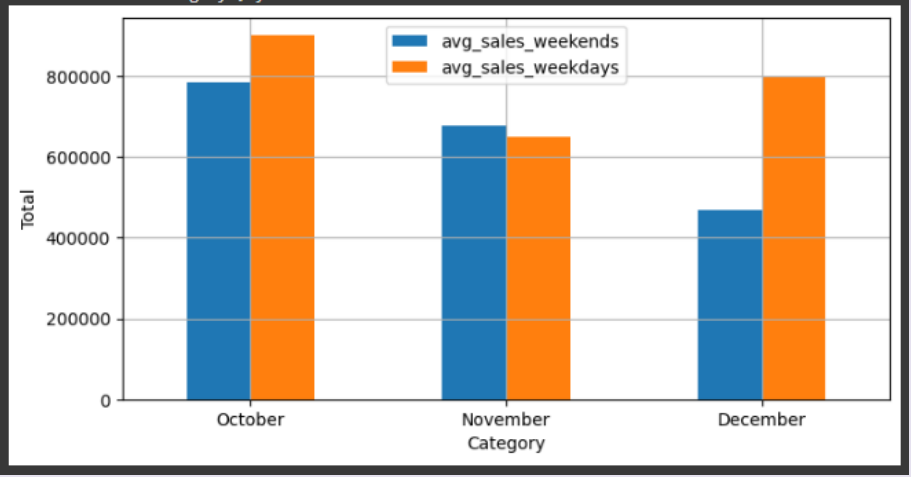
```
# Tulis kode Anda di bawah ini. Dapat menggunakan lebih dari 1 blok kode

#Mwmbuat kolom baru
df['day']=df['order_date'].dt.day_name()
df['month']=df['order_date'].dt.month_name()
df['month_number']=df['order_date'].dt.month

#Filter data
daily_weekends = (df[(df['is_valid'] == 1) & (df['day'].isin(['Saturday', 'Sunday'])) & (df['order_date'] >= '2022-10-01') & (df['order_date'] <= '2022-12-31')]
                .groupby(by=["order_date", "day", "month_number", "month"])["before_discount"]
                .mean()
                .round()
                .sort_values(ascending=False)
                .reset_index(name='average_sales_weekend')
)

daily_weekdays = (df[(df['is_valid'] == 1) & (df['day'].isin(['Monday','Tuesday','Wednesday','Thursday','Friday'])) & (df['order_date'] >= '2022-10-01') & (df['order_date'] <= '2022-12-31')]
                .groupby(by=["order_date", "day", "month_number", "month"])["before_discount"]
                .mean()
                .round()
                .sort_values(ascending=False)
                .reset_index(name='average_sales_weekday')
)

monthly_weekends = (daily_weekends.groupby(by=["month_number","month"])["average_sales_weekend"].mean()
                  .round()
                  .reset_index(name='avg_sales_weekends')
)

monthly_weekdays = (daily_weekdays.groupby(by=["month_number","month"])["average_sales_weekday"].mean()
                  .round()
                  .reset_index(name='avg_sales_weekdays')
)

# Merge data yang telah difilter
monthly = pd.merge(monthly_weekends, monthly_weekdays, on=["month_number", "month"], how="left")
monthly = monthly.sort_values(by='month_number', ascending=True)
monthly = monthly[["month", "avg_sales_weekends", "avg_sales_weekdays"]]
monthly
```

Jawaban 4.1

	month	avg_sales_weekends	avg_sales_weekdays
0	October	783975.0	899903.0
1	November	675986.0	650359.0
2	December	469057.0	795606.0

```
[20] monthly.plot(x='month',
                  y=['avg_sales_weekends','avg_sales_weekdays'],
                  kind='bar',
                  grid=True,
                  xlabel= 'Category',
                  ylabel= 'Total',
                  figsize= (8,4),
                  rot=0,
                  table= False,
                  secondary_y=False)
```

```
<Axes: xlabel='Category', ylabel='Total'>
```



Kode sebelumnya melakukan analisis penjualan produk berdasarkan hari dalam seminggu dan bulan. Setelah menambahkan kolom baru yang mencakup informasi tentang hari dan bulan dari tanggal pesanan, data difilter untuk menghitung rata-rata penjualan pada hari akhir pekan (Sabtu dan Minggu) serta hari kerja (Senin hingga Jumat) dalam rentang waktu antara Oktober hingga Desember 2022. Selanjutnya, hasilnya diagregasi untuk mendapatkan rata-rata penjualan bulanan pada hari akhir pekan dan hari kerja. Data akhirnya digabungkan dan disusun kemudian diplot dalam bentuk bar plot untuk mempermudah dalam melihat perbedaannya.

Jawaban 4.2

```
[21] # Tulis kode Anda di bawah ini. Dapat menggunakan lebih dari 1 blok kode

     #Filter data
     daily_weekends = df[(df['is_valid']==1) \
               & (df['day'].isin(['Saturday','Sunday'])) \
               & (df['order_date'] >= '2022-10-01') & (df['order_date'] <= '2022-12-31')]

     daily_weekdays = df[(df['is_valid']==1) \
               & (df['day'].isin(['Monday','Tuesday','Wednesday','Thursday','Friday'])) \
               & (df['order_date'] >= '2022-10-01') & (df['order_date'] <= '2022-12-31')]

     overal_month = {\
               'Periode' : 'Total 3 Bulan',\
               'AVG Sales Weekend' : round(daily_weekends['before_discount'].mean(),2),\
               'AVG Sales Weekday' : round(daily_weekdays['before_discount'].mean(),2),\
               }

     overal_month

     {'Periode': 'Total 3 Bulan',
      'AVG Sales Weekend': 558865.15,
      'AVG Sales Weekday': 770146.01}
```

Kode ini melanjutkan analisis penjualan pada hari akhir pekan (Sabtu dan Minggu) dan hari kerja (Senin hingga Jumat) dalam rentang Oktober hingga Desember 2022. Data difilter untuk kedua kategori tersebut, dan kemudian dihitung rata-rata penjualan (before_discount). Hasilnya disajikan dalam bentuk dictionary (overal_month) yang mencakup informasi tentang periode total tiga bulan, rata-rata penjualan pada hari akhir pekan, dan rata-rata penjualan pada hari kerja.

Thanks