

Final Project

...

Kelompok D8

Kelompok D8

Ahmad Rido Fairuji (SQL, Data Visualization)

Denny Darmawan (SQL, Data Visualization)

Desnita Rizki Irmawati (Python)

Galih Zaky Tristanaya (Python)

Jovan Aristito Arga (Python)

Modesta Berliansa Termatu Arsanta (SQL, Data Visualization)

Nadia Azahro Choirunisa (SQL, Python)

Rahmana Zahara Budi Putra (Python)

Rizky Huthama Arsyad (SQL, Data Visualization)

PYTHON

DATASET

Dataset

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from pandas.tseries.offsets import BDay
```

Mengimport beberapa library Python yaitu pandas, numpy, matplotlib, seaborn, dan BDay agar mempermudah proses analisis.

```
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)
```

Mengimport dataset yang berasal dari Github, kemudian menyimpannya sebagai dataframe.

Terdapat 4 data yang digunakan yaitu order_detail (df_od), payment_detail (df_pd), customer_detail (df_cd), dan sku_detail (df_sd).

Dataset

```
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')
```

```
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)
```

Melakukan connection menggunakan library sqlite3 agar dapat menjalankan query SQL di Google Colab. Kemudian mengubah dataframe yang sebelumnya menjadi SQL database.

Menggabungkan data pada order_detail dengan kolom payment_method, sku_name, base_price, cogs, category, dan registered_date menggunakan LEFT JOIN.

Dataset

Before

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	

After

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	

Mengubah tipe data before_discount, discount_amount, after_discount, dan base_price menjadi integer.

```
df = df.astype({"before_discount": 'int',  
               "discount_amount": 'int',  
               "after_discount": 'int',  
               "base_price": 'int'})
```

Mengubah tipe data order_date dan registered_date menjadi datetime.

```
df['order_date'] =  
pd.to_datetime(df['order_date'])  
  
df['registered_date'] =  
pd.to_datetime(df['registered_date'])
```

Dataset

order_detail

1. id → angka unik dari order / id_order
2. customer_id → angka unik dari pelanggan
3. order_date → tanggal saat dilakukan transaksi
4. sku_id → angka unik dari produk (sku adalah stock keeping unit)
5. price → harga yang tertera pada tagging harga
6. qty_ordered → jumlah barang yang dibeli oleh pelanggan
7. before_discount → nilai harga total dari produk ($\text{price} * \text{qty_ordered}$)
8. discount_amount → nilai diskon product total
9. after_discount → nilai harga total produk ketika sudah dikurangi dengan diskon
10. is_gross → menunjukkan pelanggan belum membayar pesanan
11. is_valid → menunjukkan pelanggan sudah melakukan pembayaran
12. is_net → menunjukkan transaksi sudah selesai
13. payment_id → angka unik dari metode pembayaran

Dataset

sku_detail

1. id → angka unik dari produk (dapat digunakan untuk key saat join)
2. sku_name → nama dari produk
3. base_price → harga barang yang tertera pada tagging harga / price
4. cogs → cost of goods sold / total biaya untuk menjual 1 produk
5. category → kategori produk

customer_detail

1. id → angka unik dari pelanggan
2. registered_date → tanggal pelanggan mulai mendaftarkan diri sebagai anggota

payment_detail

1. id → angka unik dari metode pembayaran
2. payment_method → metode pembayaran yang digunakan

PERTANYAAN DAN PEMBAHASAN

Pertanyaan

No. 1

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

```
top5 = df[(df['is_valid'] == 1) & \
          (df['category'] == 'Mobiles & Tablets') & \
          (df['order_date'].dt.year == 2022)]\
          .groupby('sku_name')\
          ['qty_ordered'].sum()\
          .sort_values(ascending=False)\
          .reset_index(name='Qty 2022')

top5.head(5)
```

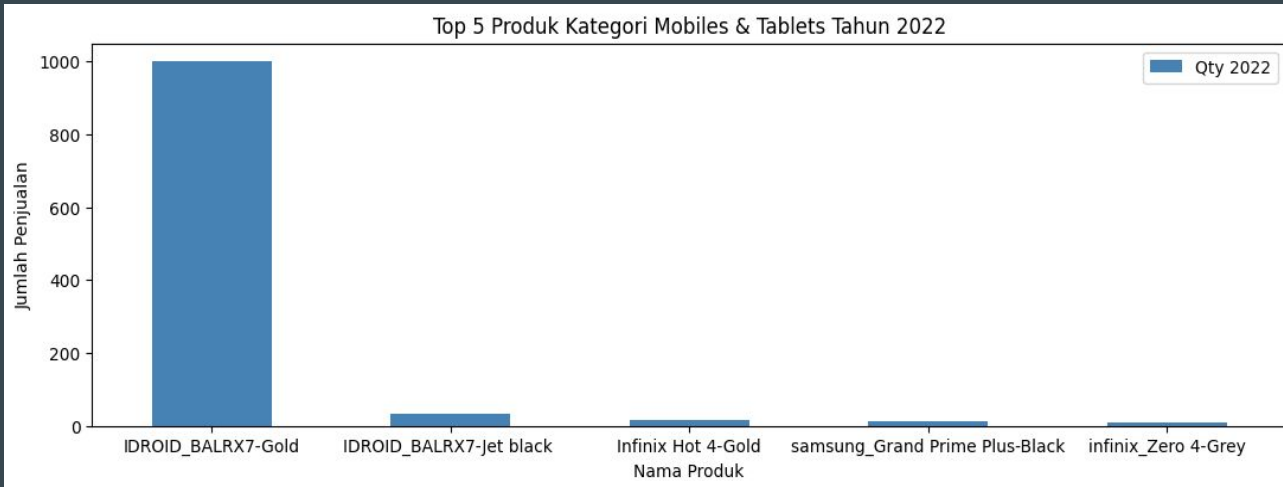
	sku_name	Qty 2022
0	IDROID_BALRX7-Gold	1000
1	IDROID_BALRX7-Jet black	31
2	Infinix Hot 4-Gold	15
3	samsung_Grand Prime Plus-Black	11
4	infinix_Zero 4-Grey	10

Dengan melakukan filtering berdasarkan **is_valid = 1**, **category = 'Mobiles & Tablets'**, dan **order_date = 2022**, kemudian melakukan grouping berdasarkan **sku_name**, dan menjumlahkan **qty_ordered**, maka didapatkan top 5 produk dengan penjualan tertinggi pada kategori tersebut selama tahun 2022.

Jawaban

```
top5.plot(x = 'sku_name',  
          y = 'Qty 2022',  
          kind = 'bar',  
          figsize = (13,4),  
          title = 'Top 5 Produk Kategori Mobiles & Tablets Tahun 2022',  
          xlabel = 'Nama Produk',  
          ylabel = 'Jumlah Penjualan',  
          rot = 0,  
          color = 'steelblue')
```

Produk-produk kategori Mobiles & Tablets yang memiliki penjualan tertinggi pada tahun 2022 adalah IDROID_BALRX7-Gold, IDROID_BALRX7-Jet black, Infinix Hot 4-Gold, samsung_Grand Prime Plus-Black, dan infinix_Zero 4-Grey.



Pertanyaan

No. 2

Menindaklanjuti meeting gabungan Tim Warehouse 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 Warehouse

Jawaban No. 1

Melakukan filtering untuk mendapatkan data penjualan setiap kategori pada tahun 2021 dan 2022. Kemudian menghitung perbandingan nilai penjualannya dan menyimpannya di kolom growth.

```
no2_21 = df[(df['is_valid'] == 1) & \
            (df['order_date'].dt.year == 2021)]\
            .groupby('category')['qty_ordered'].sum()\
            .reset_index(name='Qty 2021')\
            .sort_values(by='Qty 2021', ascending=False)

no2_22 = df[(df['is_valid'] == 1) & \
            (df['order_date'].dt.year == 2022)]\
            .groupby('category')['qty_ordered'].sum()\
            .reset_index(name='Qty 2022')\
            .sort_values(by='Qty 2022', ascending=False)

data2122 = pd.merge(no2_21, no2_22, on='category', how='left')
```

```
data2122['growth'] = data2122['Qty 2022'] - data2122['Qty 2021']
data2122.sort_values(by='growth', ascending=False)
```

	category	Qty 2021	Qty 2022	growth
13	Mobiles & Tablets	107	1154	1047
10	Women Fashion	140	489	349
2	Superstore	327	536	209
14	Entertainment	77	150	73
4	Home & Living	193	250	57
8	Kids & Baby	170	227	57
5	School & Education	184	237	53
12	Computing	109	153	44
6	Health & Sports	173	200	27
7	Books	171	195	24
11	Appliances	124	148	24
9	Beauty & Grooming	168	153	-15
3	Men Fashion	237	175	-62
0	Soghaat	759	612	-147
1	Others	426	263	-163

Jawaban No. 1

```
viz.plot(x = 'Tahun',  
        y = 'Kuantitas',  
        kind = 'bar',  
        figsize = (8,4),  
        color = ['steelblue', 'indianred'],  
        xlabel = 'Tahun',  
        ylabel = 'Kuantitas',  
        title = 'Perbandingan Kuantitas  
Penjualan Produk Kategori Others Tahun 2021 &  
2022',  
        rot = 0,  
        legend = False)
```



Dapat dilihat bahwa alasan ketersediaan stock produk kategori Others pada tahun 2022 masih banyak karena memang terdapat penurunan kuantitas penjualan pada tahun 2022.

Jawaban No. 2

Melakukan filtering untuk mendapatkan data penjualan produk kategori **Others** pada tahun 2021 dan 2022. Kemudian menghitung penurunan nilai penjualannya dan menyimpannya di kolom growth.

```
others_21 = df[(df['is_valid'] == 1) & \
               (df['category'] == 'Others') & \
               (df['order_date'].dt.year == 2021)]\
               .groupby('sku_name')\
               ['qty_ordered'].sum().reset_index(name='others_21')

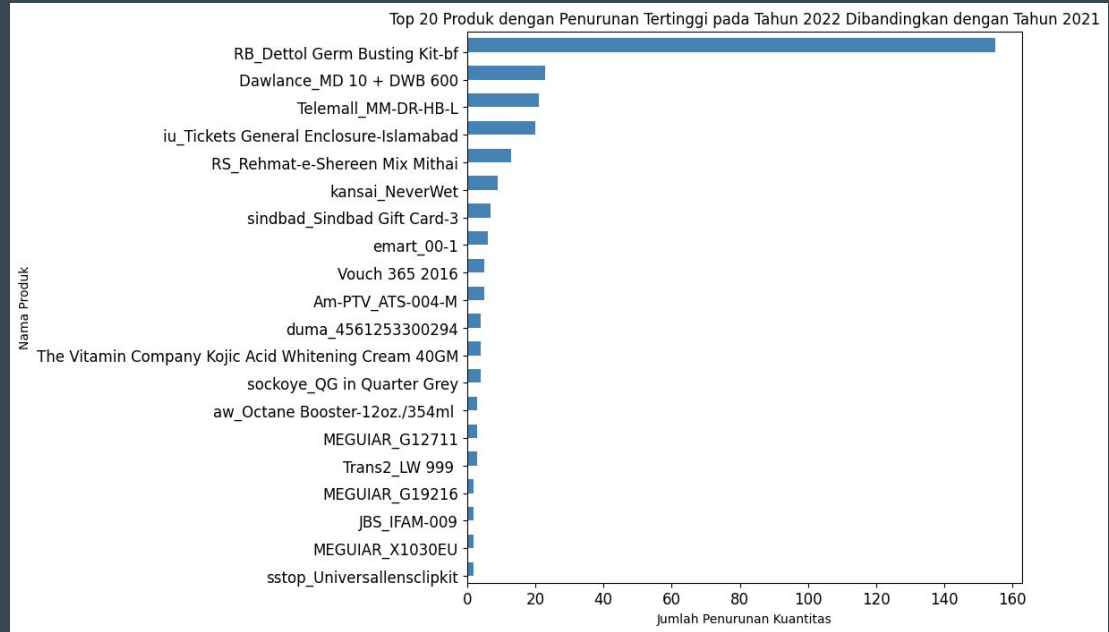
others_22 = df[(df['is_valid'] == 1) & \
               (df['category'] == 'Others') & \
               (df['order_date'].dt.year == 2022)]\
               .groupby('sku_name')\
               ['qty_ordered'].sum().reset_index(name='others_22')
```

```
others_2122 = pd.merge(others_21, others_22, on='sku_name', how='outer')
others_2122.fillna(0, inplace=True)
others_2122['growth'] = others_2122['others_22'] - others_2122['others_21']
others_2122 = others_2122.sort_values(by='growth', ascending=True).head(20)
others_2122
```

	sku_name	others_21	others_22	growth
	RB_Dettol Germ Busting Kit-bf	200.0	45.0	-155.0
	Dawlance_MD 10 + DWB 600	23.0	0.0	-23.0
	Telemall_MM-DR-HB-L	23.0	2.0	-21.0
	Iu_Tickets General Enclosure-Islamabad	20.0	0.0	-20.0
	RS_Rehmat-e-Shereen Mix Mithai	13.0	0.0	-13.0
	kansai_NeverWet	10.0	1.0	-9.0
	sindbad_Sindbad Gift Card-3	7.0	0.0	-7.0
	emart_00-1	7.0	1.0	-6.0
	Vouch 365 2016	5.0	0.0	-5.0
	Am-PTV_ATS-004-M	5.0	0.0	-5.0
	duma_4561253300294	4.0	0.0	-4.0
	sockoye_QG in Quarter Grey	4.0	0.0	-4.0
	The Vitamin Company Kojic Acid Whitening Cream...	4.0	0.0	-4.0
	aw_Octane Booster-12oz./354ml	3.0	0.0	-3.0
	MEGUIAR_G12711	4.0	1.0	-3.0
	Trans2_LW 999	3.0	0.0	-3.0
	MEGUIAR_G19216	2.0	0.0	-2.0
	JBS_IFAM-009	2.0	0.0	-2.0
	MEGUIAR_X1030EU	2.0	0.0	-2.0
	sstop_Universallensclipkit	2.0	0.0	-2.0

Jawaban No. 2

```
others_plt = others_2122([['sku_name',  
    'growth']]).sort_values(by='growth', ascending=False)  
others_plt['growth'] = others_plt['growth'].abs()  
  
others_plt.plot(x = 'sku_name',  
    y = 'growth',  
    kind = 'barh',  
    color = 'steelblue',  
    xlabel = 'Jumlah Penurunan Kuantitas',  
    ylabel = 'Nama Produk',  
    title = 'Top 20 Produk dengan Penurunan  
Tertinggi pada Tahun 2022 Dibandingkan dengan Tahun  
2021',  
    figsize = (8,8),  
    position = 0,  
    legend = False)
```



Produk dalam kategori Others yang memiliki jumlah penurunan tertinggi adalah RB_Dettol Germ Busting Kit-bf dengan jumlah penurunan sebanyak 155.

Pertanyaan

No. 3

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

```
no_3 = df[(df['is_gross'] == 1) & \
          (df['is_valid'] == 0) & \
          (df['is_net'] == 0) & \
          (df['order_date'].dt.year == 2022)]

no_3 = no_3[['customer_id', 'registered_date']] \
        .sort_values(by='registered_date',
                     ascending=False)

no_3.drop_duplicates(subset='customer_id', inplace=True)
```

Terdapat **820 customer** yang sudah melakukan check out namun belum melakukan pembayaran selama tahun 2022.

Tanggal registrasi dari customer-customer tersebut bervariasi dari tahun 2021 ke tahun 2022.

	customer_id	registered_date
315	C925840L	2022-08-28
443	C466867L	2022-08-28
157	C788348L	2022-08-25
3512	C741725L	2022-08-25
2335	C877563L	2022-08-24
...
1304	C378684L	2021-07-03
1331	C893754L	2021-07-03
606	C831683L	2021-07-03
4168	C349980L	2021-07-02
3021	C849967L	2021-07-02
820 rows × 2 columns		

Pertanyaan

No. 4

Pada bulan Oktober 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. 1

```
no4_41 = df['day'] = df['order_date'].dt.day_name()
df['month'] = df['order_date'].dt.month_name()
df['month_number'] = df['order_date'].dt.month
data4 = df[(df['is_valid']==1)&(df['month_number']\
            .isin([10,11,12]))&(df['order_date'].dt.year==2022)]
```

```
weekend = data4[(data4['day']\
                  .isin(['Saturday', 'Sunday']))]\
            .groupby(['day', 'month', 'month_number'])\
            ['before_discount'].sum()\
            .reset_index(name='weekend')

weekend
```

```
avg_weekend = weekend.groupby(['month', 'month_number'])['weekend'].mean()\
                .reset_index(name='avg_weekend')

avg_weekend
```

Berikut adalah rata-rata penjualan setiap weekend pada bulan Oktober hingga Desember tahun 2022.

	day	month	month_number	weekend
0	Saturday	December	12	2121234
1	Saturday	November	11	6545822
2	Saturday	October	10	14891674
3	Sunday	December	12	6090754
4	Sunday	November	11	5002268
5	Sunday	October	10	2233348

	month	month_number	avg_weekends
0	December	12	4105994.0
1	November	11	5774045.0
2	October	10	8562511.0

Jawaban No. 1

```
weekday = data4[(data4['day']\
                  .isin(['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday']))]\
            .groupby(['day', 'month', 'month_number'])['before_discount']\
            .sum().reset_index(name='weekday')
weekday
```

```
avg_weekday = weekday.groupby(['month', 'month_number'])['weekday'].mean()\
               .reset_index(name='avg_weekday')
avg_weekday
```

Berikut adalah rata-rata penjualan setiap weekday pada bulan Oktober hingga Desember tahun 2022.

	month	month_number	avg_weekdays
0	December	12	10251036.0
1	November	11	7445599.2
2	October	10	7872212.4

	day	month	month_number	weekday
0	Friday	December	12	2678962
1	Friday	November	11	11829274
2	Friday	October	10	7889218
3	Monday	December	12	2699668
4	Monday	November	11	3706084
5	Monday	October	10	5111076
6	Thursday	December	12	27756190
7	Thursday	November	11	4296930
8	Thursday	October	10	4050430
9	Tuesday	December	12	9767606
10	Tuesday	November	11	16613926
11	Tuesday	October	10	12259634
12	Wednesday	December	12	8352754
13	Wednesday	November	11	781782
14	Wednesday	October	10	10050704

Jawaban No. 1

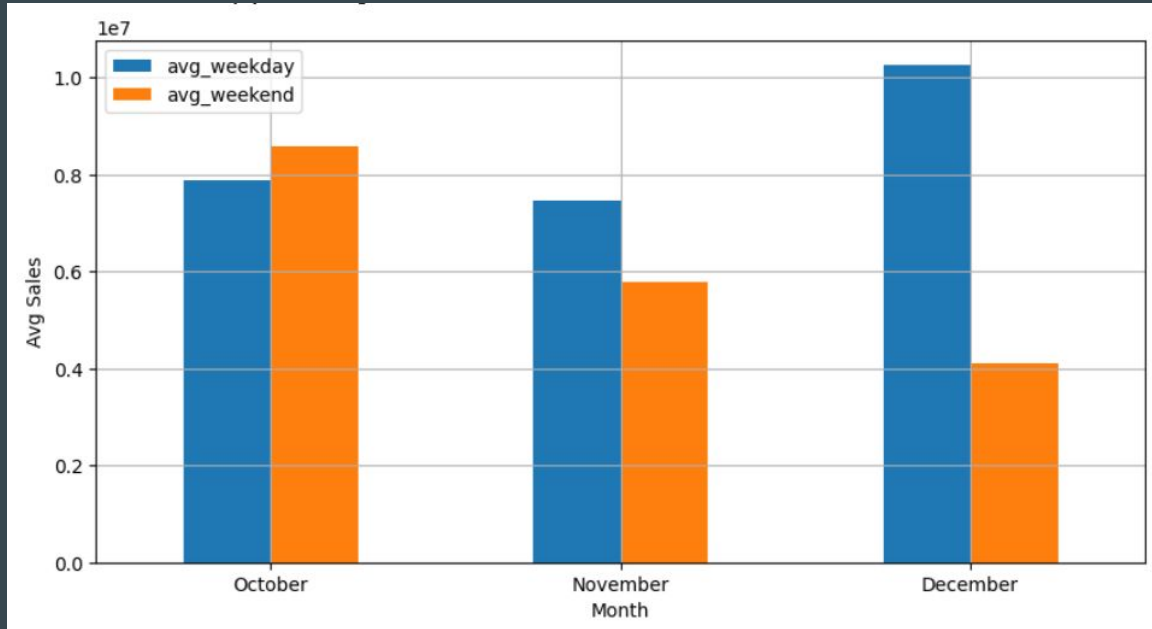
```
avg_merge = pd.merge(avg_weekends,  
                      avg_weekdays,  
                      on='month',  
                      how='left')\  
                      .sort_values(by='month_number_x',  
                                   ascending=True)  
avg_merge = avg_merge[['month', 'avg_weekdays', 'avg_weekends']]  
avg_merge['growth'] =  
avg_merge['avg_weekends'] - avg_merge['avg_weekdays']  
avg_merge
```

	month	avg_weekday	avg_weekend	growth
2	October	7872212.4	8562511.0	690298.6
1	November	7445599.2	5774045.0	-1671554.2
0	December	10251036.0	4105994.0	-6145042.0

Kolom growth menunjukkan jumlah kenaikan atau penurunan rata-rata penjualan antara weekday dan weekend pada bulan Oktober hingga Desember tahun 2022.

Jawaban No. 1

```
avg_merge.plot(x='month', y=['avg_weekday', 'avg_weekend'], kind='bar', grid=True, xlabel='Month', ylabel='Avg Sales', rot=0,
figsize=(10,5))
```



Pada bulan Oktober, rata-rata penjualan mengalami kenaikan antara weekday dan weekend.

Sedangkan pada bulan November dan Desember, rata-rata penjualan dari weekday ke weekend mengalami penurunan.

Pada bulan Desember, penurunan yang terjadi sangat banyak sehingga perlu dilakukan analisis lebih lanjut untuk mengetahui alasan penurunan tersebut.

Jawaban No. 2

```
no4_42 = data42 = {'periode' : 'Total 3 Months',  
                  'avg_sales_weekend' : weekend['weekend'].mean().round(2),  
                  'avg_sales_weekday' : weekday['weekday'].mean().round(2),  
                  'growth' : (weekend['weekend'].mean() - weekday['weekday'].mean()).round(2),  
                  'percentage' : ((weekend['weekend'].mean() -  
weekday['weekday'].mean())/weekend['weekend'].mean()*100).round(2)  
                  }  
data42 = pd.DataFrame(data42, index=[0])  
data42
```

	periode	avg_sales_weekend	avg_sales_weekday	growth	percentage
0	Total 3 Months	6147516.67	8522949.2	-2375432.53	-38.64

Jika dilihat dari keseluruhan selama 3 bulan, rata-rata penjualan antara weekday dan weekend mengalami penurunan sebanyak **38.64%**.

Thank you