



PROJECT

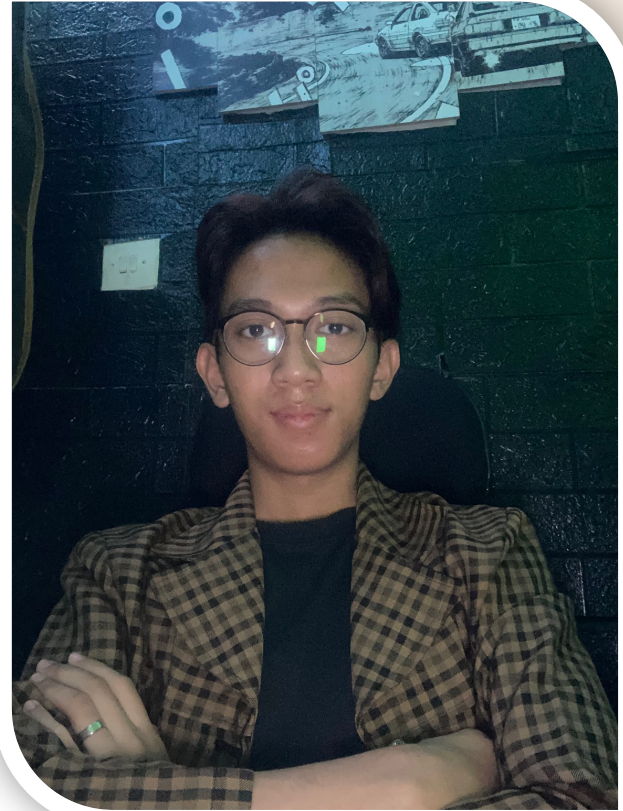
MUHAMMAD ADIKA ZAHRAN

PORTFOLIO

ABOUT ME

Halo, nama saya Adika, mahasiswa semester 4 Teknik Informatika di STMIK Jayakarta. Saya memiliki Pengalaman selama 3 bulan di PT. Intikom Berlian Mustika, di mana saya belajar tentang perangkat keras komputer dan berpartisipasi dalam project migrasi Microsoft Office On-Premise ke ExChange Online Office 365 di Bank Danamon Indonesia selama 7 Bulan.

Selain itu, saya sering mengikuti webinar dan bootcamp mengenai analisis data (SQL, Python, dan visualisasi data), Microsoft Office, serta pemrograman untuk meningkatkan wawasan, jaringan, dan keterampilan saya.



ABSTRAK

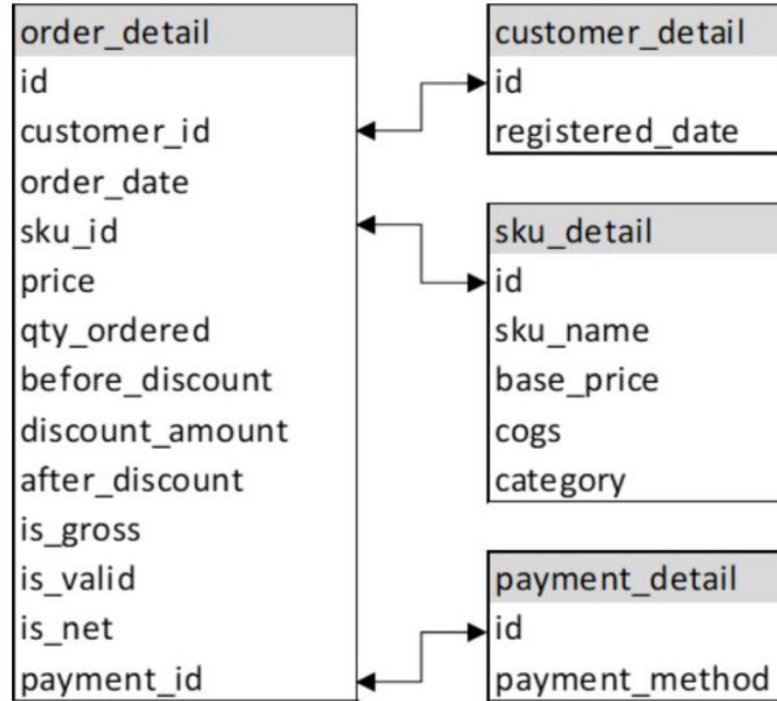
Diberikan sebuah data yang berasal dari Tokopedia (data dummy).

Terdapat empat tabel/dataset yang akan digunakan yaitu,

order_detail, sku_detail, customer_detail, dan payment_detail.

Peserta bootcamp diminta untuk menyelesaikan lima pertanyaan yang diberikan berdasarkan skema dan tabel yang sudah diberikan.

Schema Table





01

SQL

Query

The background features several abstract decorative elements: a series of concentric blue circles at the top center, a light blue wavy shape in the top right containing three small brown dots, a light brown wavy shape in the bottom right, and a light blue wavy shape in the bottom left. A thin blue line curves from the left side towards the bottom center.

CASE 1

Bandingkan nilai transaksi dari masing-masing kategori pada tahun 2021 dengan 2022. Sebutkan kategori apa saja yang mengalami peningkatan dan kategori apa yang mengalami penurunan nilai transaksi dari tahun 2021 ke 2022. Gunakan `is_valid = 1` untuk memfilter data transaksi. Source table: `order_detail`, `sku_detail`

CASE 1

- ⊖ — Bandingkan nilai transaksi dari masing-masing kategori pada tahun 2021 dengan 2022. Sebutkan kategori apa saja yang mengalami peningkatan dan
- kategori apa yang mengalami penurunan nilai transaksi dari tahun 2021 ke 2022. Gunakan is_valid = 1 untuk memfilter data transaksi.
- Source table: order_detail, sku_detail

```
SELECT category AS Category,  
       SUM(CASE WHEN strftime('%Y', od.order_date) = '2021' THEN od.after_discount ELSE 0 END) AS Transaction_2021,  
       SUM(CASE WHEN strftime('%Y', od.order_date) = '2022' THEN od.after_discount ELSE 0 END) AS Transaction_2022,  
       SUM(CASE WHEN strftime('%Y', od.order_date) = '2022' THEN od.after_discount ELSE 0 END) -  
         SUM(CASE WHEN strftime('%Y', od.order_date) = '2021' THEN od.after_discount ELSE 0 END) AS Transaction_Difference  
FROM order_detail od JOIN sku_detail sd ON od.sku_id = sd.id  
WHERE od.is_valid = 1 GROUP BY category  
HAVING Transaction_Difference <> 0;
```

CASE 1

	ABC Category ▼	123 Transaction_2021 ▼	123 Transaction_2022 ▼	123 Transaction_Difference ▼
1	Appliances	218,550,177	316,358,100	97,807,923
2	Beauty & Grooming	46,047,360	46,211,019.18	163,659.1799999997
3	Books	10,124,596	6,792,519.2	-3,332,076.8
4	Computing	172,878,860	214,028,543.4	41,149,683.400000006
5	Entertainment	162,326,357.4	365,344,148.9	203,017,791.49999997
6	Health & Sports	33,837,965.6	54,235,579.6	20,397,614
7	Home & Living	45,797,873	79,483,716.2	33,685,843.2
8	Kids & Baby	23,971,057.8	25,931,276.84	1,960,219.039999999
9	Men Fashion	58,628,198	135,588,253	76,960,055
10	Mobiles & Tablets	370,606,718	918,451,576	547,844,858
11	Others	40,468,515.74	21,744,646.02	-18,723,869.720000003
12	School & Education	11,558,982.4	17,362,465.3	5,803,482.9
13	Soghaat	15,056,202.6	17,658,332	2,602,129.4000000004
14	Superstore	28,828,088	32,643,266.52	3,815,178.5199999996
15	Women Fashion	84,045,961.4	93,014,970.62	8,969,009.219999999

CASE 2

Urutkan dari ke-5 produk ini berdasarkan nilai transaksinya.

1. Samsung
2. Apple
3. Sony
4. Huawei
5. Lenovo


Gunakan `is_valid = 1` untuk memfilter data transaksi. Source table: `order_detail`, `sku_detail`


CASE 2

```
-- Q: Urutkan dari ke-5 produk ini berdasarkan nilai transaksinya. 1. Samsung, 2. Apple, 3. Sony, 4. Huawei, 5. Lenovo. Gunakan is_valid = 1
-- untuk memfilter data transaksi. Source table: order_detail, sku_detail

SELECT
  CASE
    WHEN LOWER(sd.sku_name) LIKE '%samsung%' THEN 'Samsung'
    WHEN LOWER(sd.sku_name) LIKE '%apple%' THEN 'Apple'
    WHEN LOWER(sd.sku_name) LIKE '%sony%' THEN 'Sony'
    WHEN LOWER(sd.sku_name) LIKE '%huawei%' THEN 'Huawei'
    WHEN LOWER(sd.sku_name) LIKE '%lenovo%' THEN 'Lenovo'
  END AS product_name,
  SUM(od.after_discount * od.qty_ordered) AS total_sales
FROM
  order_detail od
JOIN
  sku_detail sd ON od.sku_id = sd.id
WHERE
  od.is_valid = 1
  AND (
    LOWER(sd.sku_name) LIKE '%samsung%'
    OR LOWER(sd.sku_name) LIKE '%apple%'
    OR LOWER(sd.sku_name) LIKE '%sony%'
    OR LOWER(sd.sku_name) LIKE '%huawei%'
    OR LOWER(sd.sku_name) LIKE '%lenovo%'
  )
GROUP BY
  product_name
ORDER BY
  total_sales DESC;
```

CASE 2

	 ABC product_name ▼	123 total_sales ▼
1	Samsung	2,081,154,492
2	Apple	213,310,080
3	Huawei	141,963,700
4	Sony	63,960,718
5	Lenovo	62,379,800.4



02

Python

Query

Pre Processing Data

Import Libraries →



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

Import Data Sources →

```
[ ] #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)
```

Pre Processing Data

Join The Table →

```
[ ] #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)
```

Pre Processing Data

Table result after join :

	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

CASE 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

Syntax 1

```
▶ df1 = pd.DataFrame(  
    df[(df.category == 'Mobiles & Tablets')  
      & (df.order_date.between('2022-01-01', '2022-12-31'))  
      & (df.is_valid == 1)]\  
    .groupby(by = 'sku_name')[['qty_ordered', 'after_discount']].sum()\  
    .sort_values(by='qty_ordered', ascending=False))  
df1.head()
```

Result 1

Berikut merupakan hasil dari top 5 produk dari kategori mobiles & tables. Dari hasil analisis menunjukkan bahwa IDROID_BALRX7-Gold paling banyak diminati selama festival akhir tahun pada tahun 2022



sku_name	qty_ordered	after_discount
Samsung-Galaxy-S8-G955-Plus-Black	5	25853210
iphone_7-128GB-wof-Matt Black	4	22828568
samsungGALAXY J510F-2016 DS LTE-Gold	4	5800000
cc_samsung_G935F-Blue	4	14924386
Samsung_S7 Edge-Gold	3	10896576



CASE 2

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

- ❖ 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

Syntax 2

```
▶ # Memfilter data dengan valid = 1 & category = Others & tahun = 2021
df_2021 = df[(df['is_valid'] == 1) & (df['category'] == 'Others') & (df['order_date'].dt.year == 2021)]

# Memfilter data dengan valid = 1 & category = Others & tahun = 2022
df_2022 = df[(df['is_valid'] == 1) & (df['category'] == 'Others') & (df['order_date'].dt.year == 2022)]

# Menghitung total penjualan per produk per tahun
df_2021_top_20 = df_2021.groupby('sku_name')[['after_discount']].agg(
    {'after_discount': 'sum'}
).sort_values(by='after_discount', ascending=False)

df_2022_top_20 = df_2022.groupby('sku_name')[['after_discount']].agg(
    {'after_discount': 'sum'}
).sort_values(by='after_discount', ascending=False)

# Menggabungkan data
df_combined_decrease = pd.merge(df_2022_top_20, df_2021_top_20, on='sku_name', how='inner', suffixes=('_2022', '_2021'))

# Menghitung penurunan after discount
df_combined_decrease['decrease'] = df_combined_decrease['after_discount_2022'] - df_combined_decrease['after_discount_2021']

# Mengurutkan berdasarkan penurunan after discount
df_combined_decrease = df_combined_decrease.sort_values(by='decrease', ascending=True)
df_combined_decrease.head(20)
```

Result 2

Berikut merupakan top 20 produk pada kategori others. Dari hasil analisis ditunjukkan bahwa produk RB_Dettol Germ Busting Kit- bf mengalami penurunan sebesar Rp. 1.418.156 pada tahun 2022 dibandingkan dengan tahun 2021.

	after_discount_2022	after_discount_2021	decrease
sku_name			
RB_Dettol Germ Busting Kit-bf	1191844	2610000	-1418156
kansai_NeverWet	76850	1203500	-1126650
Telemall_MM-DR-HB-L	52084	590266	-538182
MEGUIAR_G12711	147552	605056	-457504
emart_00-1	39904	320566	-280662
Aladdin_bike_cover	68440	250212	-181772
Entertainer Asia_Vouch 365-2017 Mobile App Lahore	87000	174000	-87000
aw_Ultra Shine Wash & Wax-64oz./1893ml	60320	120640	-60320
Tmall_MM-DR-PAD	57942	115884	-57942
Lacie_9000146	439002	463420	-24418
Entertainer Asia_Vouch 365 - 2017 Book Karachi	222314	232000	-9686
electro_Humidifier	57942	57942	0
MEGUIAR_A1214	81142	81142	0
kansai_Undercoating Aerosol	334080	317550	16530
aw_CONSTRUCTION FOAM-700ml	99760	43790	55970
vitamin_265	114840	57420	57420
MEGUIAR_G18211	368880	155440	213440
emart_0-37	489080	74240	414840
emart_Tyre Shape Air Compressor	2000362	579710	1420652

CASE 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**


Syntax 3





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

Result 3

Berikut merupakan list dari `customer_id` dan juga `registered_date` yang sudah melakukan check-out, akan tetapi belum melakukan pembayaran pada tahun 2022 yaitu sebanyak 975 orang.



	customer_id	registered_date
7	C215074L	2021-09-13
26	C869172L	2022-01-20
53	C256400L	2022-07-06
54	C544122L	2022-05-13
58	C225454L	2022-05-12
...
5871	C130292L	2022-04-28
5873	C357056L	2022-06-26
5877	C714630L	2022-06-24
5878	C300950L	2021-09-05
5879	C295679L	2022-07-23



975 rows x 2 columns

CASE 4

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:

- ❖ 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.
- ❖ 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

Syntax 4.1

```
▶ # Filter data yang valid dan pada tahun 2022
df4 = df[(df['is_valid'] == 1) & (df['order_date'].dt.year == 2022)]

# Ekstrak informasi tanggal
df4['day'] = df4['order_date'].dt.weekday
df4['month'] = df4['order_date'].dt.month

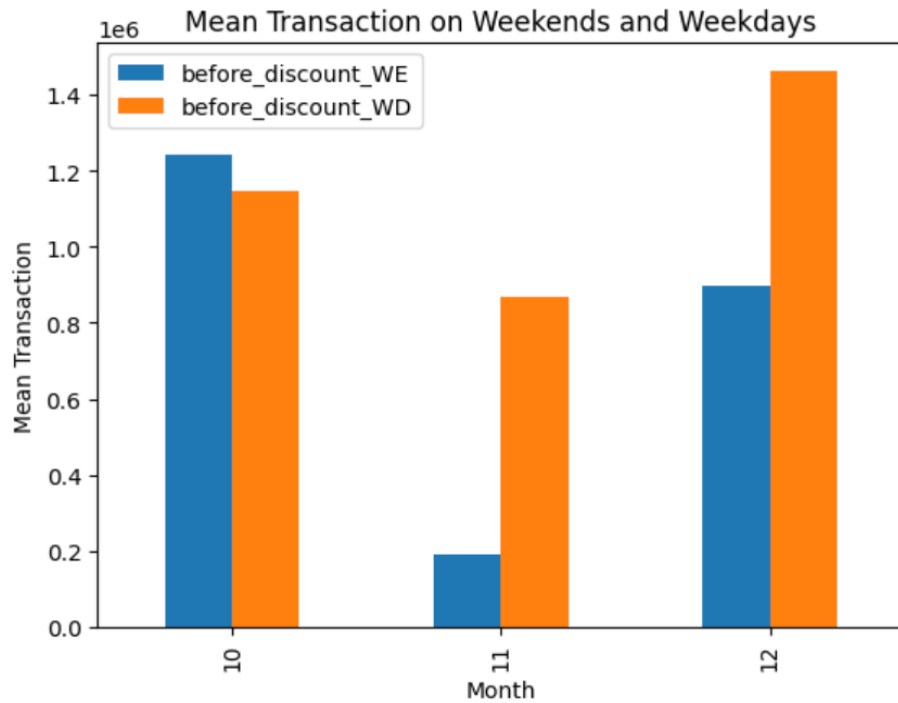
# Pisahkan data akhir pekan dan hari kerja pada bulan Oktober, November, dan Desember
dfweekends = df4[(df4['day'].isin([5, 6])) & (df4['month'].isin([10, 11, 12]))]
dfweekdays = df4[(df4['day'].isin([0, 1, 2, 3, 4])) & (df4['month'].isin([10, 11, 12]))]

# Hitung rata-rata pembelian per bulan untuk akhir pekan dan hari kerja
dfweekends_mean = dfweekends.groupby('month')['before_discount'].mean()
dfweekdays_mean = dfweekdays.groupby('month')['before_discount'].mean()

# Plot grafik
df4A = pd.concat([dfweekends_mean, dfweekdays_mean], axis=1)
df4A.columns = ['before_discount_WE', 'before_discount_WD']
df4A.plot(kind='bar')
plt.xlabel('Month')
plt.ylabel('Mean Transaction')
plt.title('Mean Transaction on Weekends and Weekdays')
plt.show()
```

Result 4.1

[↕]



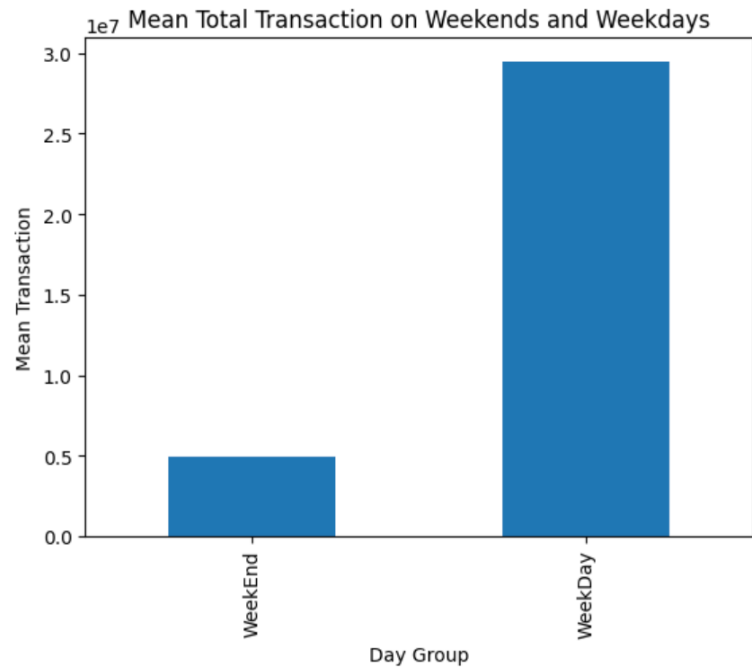
Syntax 4.2

```
# Pisahkan data akhir pekan dan hari kerja pada bulan Oktober, November, dan Desember
df4WE = df4[(df4['day'].isin([5,6])) & (df4['month'].isin([10,11,12]))]
df4WD = df4[(df4['day'].isin([0,1,2,3,4])) & (df4['month'].isin([10,11,12]))]

# Hitung total transaksi per bulan untuk akhir pekan dan hari kerja
df4B = pd.concat([df4WE.groupby('month')['before_discount'].sum(), df4WD.groupby('month')['before_discount'].sum()], axis=1)
df4B.columns = ['WeekEnd', 'WeekDay']

# Plot grafik
df4B.mean().plot(kind='bar')
plt.xlabel('Day Group')
plt.ylabel('Mean Transaction')
plt.title('Mean Total Transaction on Weekends and Weekdays')
plt.show()
```

Result 4.2





03

Data Visualization

Looker Studio

CASE 1

1. Tim marketing ingin melihat perkembangan campaign pada tahun 2022, trend nya seperti apa? Diberikan penjelasan dan bagaimana call to action-nya agar tim dapat mengambil sebuah keputusan ke depannya. Dashboard yang berupa:

A. Hubungan antara Value Sales (*before discount*), Net Profit, dan AOV (*average order value*), Note:

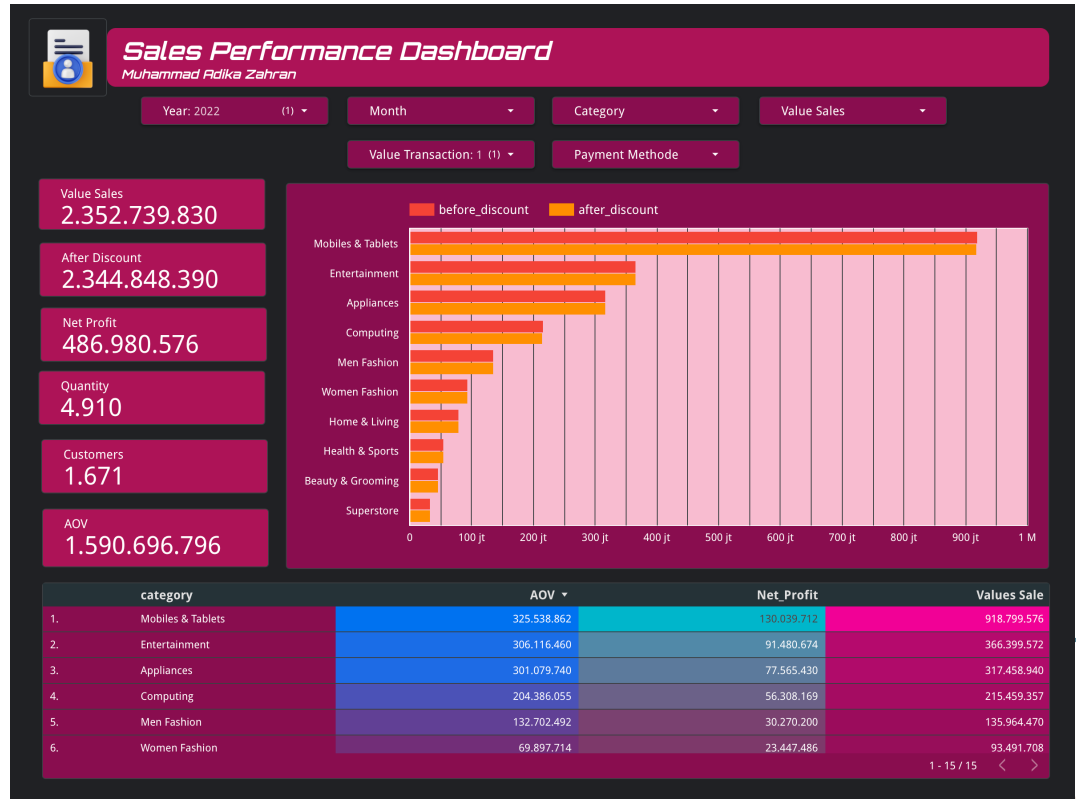
- Net profit = Value Sales (before discount) - (cogs * qty)
- AOV = Value Sales (before discount) / Total Unique Order.

B. Terdapat *slicer* Order Date, Category, Sales Value, Value Transaction, Pay ment, Note:

- Value Transaction:
 - ❖ Valid → is_valid = 1
 - ❖ Not Valid → is_valid = 0
- Payment → payment_method :

Result 1

Berikut merupakan hasil hubungan antara sales value, net profit, dan juga AOV selama tahun 2022. Berdasarkan hasil analisis dari gambar grafik disamping menunjukkan adanya penurunan penjualan dalam 3 bulan terakhir (okt, nov, des). Selain itu, terdapat lonjakan tertinggi dari bulan Juli hingga Agustus 2022.



CASE 2

Dapat memperlihatkan:

A. Tabel berisi:

- Product Name
- Category
- Before Discount
- After Discount
- Net Profit
- Qty
- Customer (*unique value*)

B. Terdapat slicer: Order Date, Category, Value Transaction, Payment

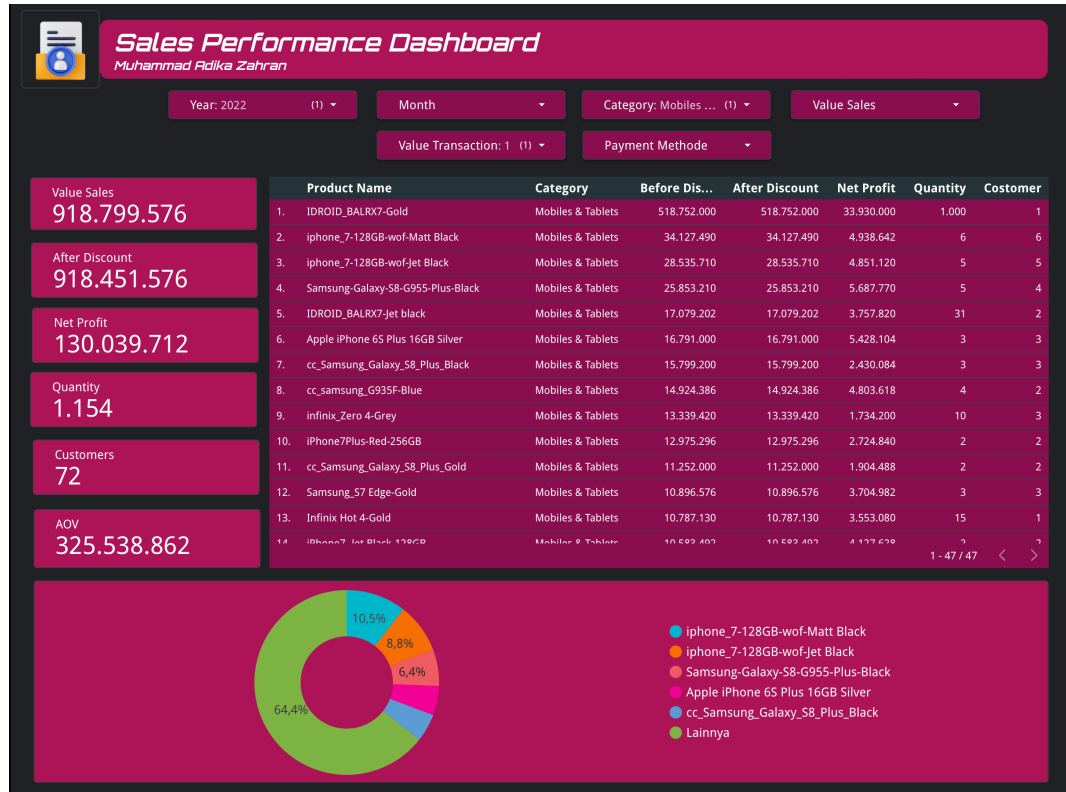
C. Scorecard :

- Before Discount
- After Discount
- Net Profit
- Quantity
- Customer (*unique value*)
- AOV

Selama 2022, tampilkan category mobile & tablet yang sudah terjadi pembayaran secara jazzwallet. Berapa quantity dan customer nya?

Result 2

Berdasarkan hasil filter, tidak terdapat customer yang melakukan pembayaran menggunakan jazzwallet pada kategori Mobiles & Tablets. Analisis mengapa tidak ada customer yang menggunakan jazzwallet pada transaksi yaitu kemungkinan terdapat perbaikan pada aplikasi tersebut. Selain itu, lakukan kolaborasi dengan pihak jazzwallet untuk meningkatkan visibilitas dan popularitasnya di platform Tokopaedi.





Thank
You