MySkill

#RintisKarirImpian

Python – Ecommerce Analysis

Final Project Myskill | Luthfi Mahfuzh

python™ + colab

Table Of Contents

MySkill

Overview

This Python Final Project is one of practice to answer any business question including import library, creating logic syntax and basic with “Real data” in Industry. With using python we can analyze any data to be insight to facing any question or business task with fast and efficient

In this case, we already provided sales dataset from MySkill team, The dataset will be analyzed to be answered business question using SQL for handling any complex query and big database

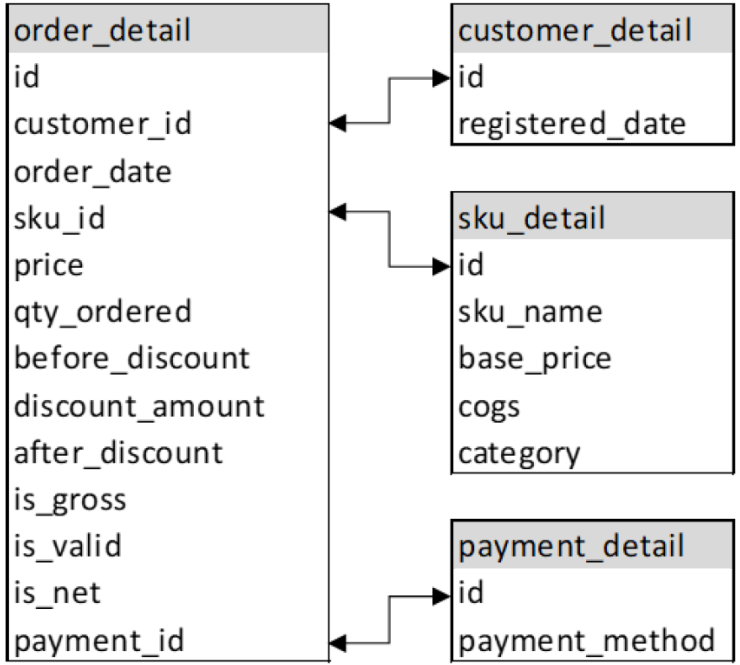
Dataset

Data source that we used is sales data from one of e-commerce platform which already modified for exercise using phyton

The Dataset is consist of 4 datas :

1. Order Detail
2. Customer Detail
3. SKU Detail
4. Payment Detail

Table sceme



Tools & Preparation

What is Python ?

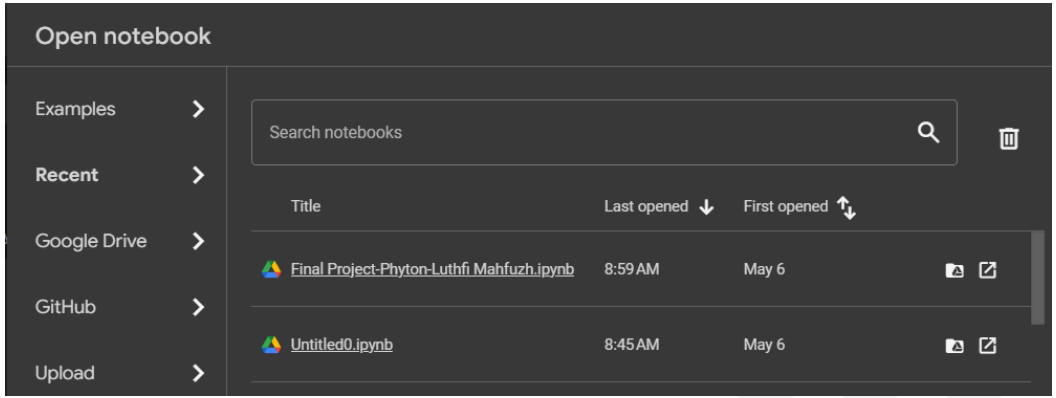
- Python is a high-level programming language known for its simplicity and readability. It was created by Guido van Rossum and first released in 1991
- Python emphasizes code readability and a clean syntax, making it an ideal language for beginners and experienced programmers alike
- Python's simplicity, readability, versatility, and strong community support have contributed to its widespread popularity and adoption across various industries and domains.

"Python has a comprehensive toolbox to assist us in statistical calculations, mathematics, data analysis, and machine learning with the support of a wide community-based ecosystem."

Tools & Preparation

In this case we run python with google colab
https://colab.research.google.com

and the open notebook that provided from MySkill Team. Click File -> Open Notebook -> Select Notebook



Pre-Processing

Import libraries :

- Pandas
- Numpy
- Matplotlib
- Seaborn
- Pandas.Tseries

Final Project-Phyton-Luthfi Mahfuzh.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

```
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,
In this case we use data (.csv) that stored in github :

1. order_detail
2. paymentdetail
3. customer_detail
4. sku_detail

Final Project-Phyton-Luthfi Mahfuzh.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] #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

Displaying Top 5 Rows,
Using function : **head()**

Final Project-Phyton-Luthfi Mahfuzh.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment

+ Code + Text

```
#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

Displaying Bottom 5 Rows,
Using function : **tail()**

```
#Menampilkan 5 baris terakhir
df_sd.tail(5)
```

	id	sku_name	base_price	cogs	category
3201	P606727	PNG_Pampers_10003447-Lahore	58.0	0	Superstore
3202	P894743	emarthazir_BBQ Pro Kit-Lahore	58.0	0	Home & Living
3203	P194155	Bahr-e-Shifa	0.0	0	Books
3204	P446333	stinnos_1500	0.0	0	Kids & Baby
3205	P181645	JBS_IFAM-032	0.0	0	Others

Pre-Processing

Run SQL in Google Collab

Final Project-Phyton-Luthfi Mahfuzh.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] #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')
```

JOIN 4 table using SQL in Google Collab

```
#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

Displaying data type

```python
#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
```

Change data type from float to integer

```python
#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
```

Change data type from date to datetime

```python
#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
```

Question 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

Answer - Question 1

Final Project-Phyton-Luthfi Mahfuzh.ipynb

File Edit View Insert Runtime Tools Help Saving failed since 9:59 AM

+ Code + Text

```python
# Save Data with pandas dataframe form
data_question1 = pd.DataFrame (\
# Filter data with is_valid = 1
df[(df['is_valid'] == 1) &\
# Filter data with Category = Mobiles & Tablets
(df['category'] == "Mobiles & Tablets") &\
# Filter order date only in year 2022
(df['order_date'].dt.year==2022)]
# Grouping data by SKU Name
.groupby(by=['sku_name'])\
# Aggregating value qty ordered
.agg({'qty_ordered':'sum'})\
# Reseting index
.reset_index()\
# Sorting qty ordered value with descending order
.sort_values('qty_ordered', ascending=False))
data_question1.head()
```

	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

Result

Question 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

Answer - Question 2.1

python™

Jawaban No 2.1

```python
# Save Data with pandas dataframe form
data_question2_part1 = pd.DataFrame(\
# Filter with is_valid = 1
df[(df['is_valid']==1) &\
# Filter order date only in year 2021
(df['order_date'].dt.year==2021)]
# Grouping by category
.groupby(by=["category"])["qty_ordered"]\
# aggregat by summarize of qty ordered
.sum()\
# Sorting Value by Descending order
.sort_values(ascending=False)\
# Reset header name with : qty_2021
.reset_index(name='qty_2021'))
data_question2_part1
```

Result

	category	qty_2021
0	Soghaat	759
1	Others	426
2	Superstore	327
3	Men Fashion	237
4	Home & Living	193
5	School & Education	184
6	Health & Sports	173
7	Books	171
8	Kids & Baby	170
9	Beauty & Grooming	168
10	Women Fashion	140
11	Appliances	124
12	Computing	109
13	Mobiles & Tablets	107
14	Entertainment	77

Answer - Question 2.1 (cont')

```python
# Save Data with pandas dataframe form
data_question2_part2 = pd.DataFrame(\
# Filter with is_valid = 1
df[(df['is_valid']==1) &\
# Filter order date only in year 2022
(df['order_date'].dt.year==2022)]
# Grouping by category
.groupby(by=["category"])["qty_ordered"]\
# aggregate by summarize of qty ordered
.sum()\
# Sorting Value by Descending order
.sort_values(ascending=False)\
# Reset header name with : qty_2022
.reset_index(name='qty_2022'))
data_question2_part2
```

Result

	category	qty_2022
0	Mobiles & Tablets	1154
1	Soghaat	612
2	Superstore	536
3	Women Fashion	489
4	Others	263
5	Home & Living	250
6	School & Education	237
7	Kids & Baby	227
8	Health & Sports	200
9	Books	195
10	Men Fashion	175
11	Beauty & Grooming	153
12	Computing	153
13	Entertainment	150
14	Appliances	148

Answer - Question 2.1 (cont')

```python
# Combine data 2021 & 2022
data_question2_merge = data_question2_part1.merge (data_question2_part2, left_on ='category', right_on ='category')
data_question2_merge
```

	category	qty_2021	qty_2022
0	Soghaat	759	612
1	Others	426	263
2	Superstore	327	536
3	Men Fashion	237	175
4	Home & Living	193	250
5	School & Education	184	237
6	Health & Sports	173	200
7	Books	171	195
8	Kids & Baby	170	227
9	Beauty & Grooming	168	153
10	Women Fashion	140	489
11	Appliances	124	148
12	Computing	109	153
13	Mobiles & Tablets	107	1154
14	Entertainment	77	150

Question 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

Answer - Question 3

Jawaban No 3

```python
# assign variable with dataframe form
data_customer = df[\
# filter data by is_gross = 1
(df['is_gross']==1) &\
# filter data by is_valid = 0
(df['is_valid']==0) &\
# filter data by is_net = 0
(df['is_net']==0) &\
# filter data by order_date in 2022
(df['order_date'].dt.year==2022)]
# add registered_date column
data_question3 = data_customer[['customer_id','registered_date']]
data_question3
```

Result

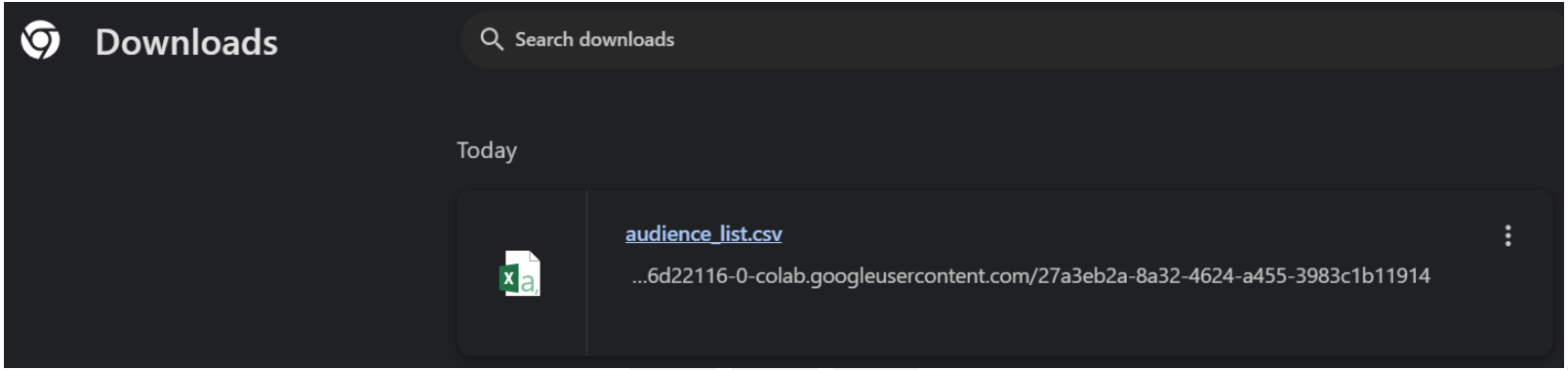
	customer_id	registered_date
9	C246762L	2022-05-08
18	C848774L	2021-11-07
19	C693415L	2022-04-12
21	C180595L	2022-04-22
22	C587425L	2022-03-22
...	...	...
5856	C394076L	2021-10-12
5859	C248585L	2022-07-10
5865	C471304L	2022-05-13
5881	C265450L	2022-02-17
5883	C676393L	2021-07-27

Answer - Question 3 (cont')

Syntax for download as .csv file

```
# code for download file
from google.colab import files
data_question3.to_csv('audience_list.csv', encoding = 'utf-8-sig',index=False) #ganti [nama variabel file] dengan nama variabel yang digunakan
files.download('audience_list.csv')
```

Result



Question 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

Answer - Question 4.1

Syntax for add columns in data frame

Jawaban No 4.1

[28]
```
#Create additional column day, month, month_num
df['day']=df['order_date'].dt.day_name()
df['month']=df['order_date'].dt.month_name()
df['month_num']=df['order_date'].dt.month
df.head()
```

	id	customer_id	order_date	sku_id	price	qty_ordered	before_discount	discount_amount	after_discount	is_gross	...	payment_id	payment_method	sku_name	base_price	cogs
0	ODR9939707760w	C713589L	2021-11-19	P858068	26100	200	5220000	2610000	2610000	1	...	5	jazzwallet	RB_Dettol Germ Busting Kit-bf	26100	18270
1	ODR7448356649d	C551551L	2021-11-19	P886455	1971942	5	9859710	2464927	7394782	1	...	5	jazzwallet	PS4_Slim-500GB	1971942	1321182
2	ODR4011281866z	C685596L	2021-11-25	P678648	7482000	1	7482000	2065344	5416655	1	...	4	Payaxis	Changhong Ruba 55 Inches UD55D6000i Ultra HD T...	7482000	5162580
3	ODR3378927994s	C830683L	2021-11-22	P540013	3593680	1	3593680	1455440	2138239	1	...	5	jazzwallet	dawlance_Inverter 30	3593680	3054628
4	ODR4904430099k	C191766L	2021-11-21	P491032	4413220	1	4413220	1059172	3354047	1	...	4	Payaxis	Dawlance_Inverter-45 2.0 ton	4413220	3177472

5 rows × 22 columns

Answer - Question 4.1 (cont')

Syntax for create data average sales in weekend from oct to dec 2022

```
[31] #assign for data average sales in weekend
     data_avg_weekend = pd.DataFrame (\
     #Filter data by is valid = 1
         df[(df['is_valid']==1) &\
     #Filter data only for saturday & sunday
            (df['day'].isin(['Saturday','Sunday'])) &\
     #filter data only for transaction in Oct until Dec 2022
            (df['order_date'] >='2022-10-01') & (df['order_date'] <='2022-12-31')]
     #grouping data by month number, month name
     .groupby(by=["month_num","month"])["before_discount"]
     #aggregating data before discount with mean
     .mean()\
     # rounding result
     .round()\
     # sorting average sales by descending order
     .sort_values(ascending=False)\
     # reset header name to avg_sales_weekend
     .reset_index(name='avg_sales_weekend'))
     data_avg_weekend
```

	month_num	month	avg_sales_weekend
0	10	October	634260.0
1	11	November	607794.0
2	12	December	410599.0

Answer - Question 4.1 (cont')

Syntax for create data average sales in weekdays from oct to dec 2022

```
[31] #assign for data average sales in weekend
     data_avg_weekend = pd.DataFrame (\
     #Filter data by is valid = 1
         df[(df['is_valid']==1) &\
     #Filter data only for saturday & sunday
            (df['day'].isin(['Saturday','Sunday'])) &\
     #filter data only for transaction in Oct until Dec 2022
            (df['order_date'] >='2022-10-01') & (df['order_date'] <='2022-12-31')]
     #grouping data by month number, month name
     .groupby(by=["month_num","month"])["before_discount"]
     #aggregating data before discount with mean
     .mean()\
     # rounding result
     .round()\
     # sorting average sales by descending order
     .sort_values(ascending=False)\
     # reset header name to avg_sales_weekend
     .reset_index(name='avg_sales_weekend'))
     data_avg_weekend
```

	month_num	month	avg_sales_weekend
0	10	October	634260.0
1	11	November	607794.0
2	12	December	410599.0

Answer - Question 4.1 (cont')

Syntax for combine data average sales in weekdays and weekend from oct to dec 2022

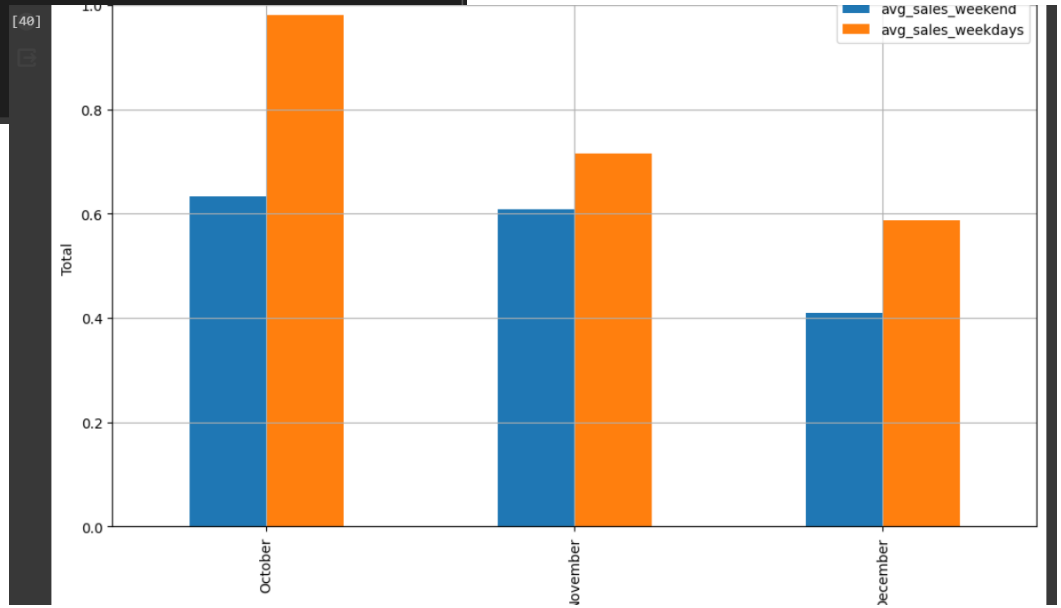
```
data_avg_combine = data_avg_weekend.merge(data_avg_weekdays, left_on ='month', right_on ='month')
data_avg_combine.sort_values(by='month_num_x',ascending=True, inplace=True)
data_avg_combine = data_avg_combine[["month","avg_sales_weekend","avg_sales_weekdays"]]
data_avg_combine
```

	month	avg_sales_weekend	avg_sales_weekdays
0	October	634260.0	980851.0
1	November	607794.0	715893.0
2	December	410599.0	587475.0

Answer - Question 4.1 (cont')

Syntax for create chart or data visualization with **".plot"** Function

```
[40] data_avg_combine.plot(x='month',y=['avg_sales_weekend','avg_sales_weekdays'],
     kind='bar',grid =True,
     xlabel ='Category',
     ylabel ='Total',
     figsize=(12,7),
     rot =90,
     table =False,
     secondary_y =False)
```



Answer - Question 4.2

Syntax for Assign data average sales in weekdays and weekend from oct to dec 2022 to new variable

Jawaban No 4.2

```
[42] #assign for data average sales in weekend
     data_avg_weekend_part2 = pd.DataFrame (\
     #Filter data by is valid = 1
         df[(df['is_valid']==1) &\
     #Filter data only for saturday & sunday
            (df['day'].isin(['Saturday','Sunday'])) &\
     #filter data only for transaction in Oct until Dec 2022
            (df['order_date'] >='2022-10-01') & (df['order_date'] <='2022-12-31')])
```

```
[43] #assign for data average sales in weekdays
     data_avg_weekdays_part2 = pd.DataFrame (\
     #Filter data by is valid = 1
         df[(df['is_valid']==1) &\
     #Filter data only for monday until friday
            (df['day'].isin(['Monday','Tuesday','Wednesday','Thusday','Friday'])) &\
     #filter data only for transaction in Oct until Dec 2022
            (df['order_date'] >='2022-10-01') & (df['order_date'] <='2022-12-31')])
```

Answer - Question 4.2 (cont')

Syntax for combine data average sales in weekend & weekdays and then display the gap in value and percentage

```
#assign for data average sales combine weekend & weekdays
data_average_combine2 = {'Periode':'Total 3 months',\
          'Avg Weekend Sales':round(data_avg_weekend_part2['before_discount'].mean(),2), \
          'Avg Weekdays Sales':round(data_avg_weekdays_part2['before_discount'].mean(),2),\
          #assign for different value between average sales weekend & weekdays
          'Diff (Value)': round(data_avg_weekend_part2['before_discount'].mean() -
                                data_avg_weekdays_part2['before_discount'].mean()),\
          #assign for different value in percentage
          'Diff (%)': pd.Series(round(((data_avg_weekend_part2['before_discount'].mean() - data_avg_weekdays_part2['before_discount'].mean()))))/
                                data_avg_weekend_part2['before_discount'].mean())}
pd.DataFrame(data=data_average_combine2, index=[0])
```

	Periode	Avg Weekend Sales	Avg Weekdays Sales	Diff (Value)	Diff (%)
0	Total 3 months	558865.15	751972.85	-193108	-0.345536

Please give me feedback to :
Email : luthfimahfuzh03@gmail.com
Telegram : @luthfimahfuzh
Linkedin : https://www.linkedin.com/in/luthfim/

Link notebook – google colab:
https://colab.research.google.com/drive/1aStWFggEX2lE6iTqUW39lvjhxpcTMQsd?usp=sharing

Thank You!

#RintisKarirImpian

MySkill