SELAMAT DATANG DI TUGAS LEONARDO !!

Saya menggunakan Python karna pernah bersih-bersih string bareng temen di Python ehe.

Packages

Karna Packages begitu berharga

[ ] ↳ *1 cell hidden*

Memasukkan Data

```
# Memasukkan Data
data = pd.read_csv('/content/raw data.csv')
# Melihat sekilas Data
data.head(5)
```

	Unnamed: 0	nama	harga	seller	terjual	tanggal_ambil_data
0	1	Bertolli Extra Virgin Olive Oil / Minyak Zaitu...	Rp85.000	Food Republic	Terjual 1.779 Produk	2020-09-04
1	2	Original Extra Virgin Olive Oil (EVOO) Casa Di...	Rp60.000	Yubyre	Terjual 1.624 Produk	2020-09-04
2	3	EVOO Baby Olive Oil Casa Di Oliva Olivia For K...	Rp58.000	Papamama Babyshop	Terjual 2.161 Produk	2020-09-04
3	4	Casa di Oliva - Extra Virgin Olive Oil for Kid...	Rp59.000	Chubby Baby Shop	Terjual 1.696 Produk	2020-09-04

Hemm... Kalau diliat-liat datanya kotor juga ya. Tapi sebelum dikomentari lebih jauh akan dilihat terlebih dahulu ukuran datanya

```
# Melihat Ukuran Data
print('Banyak Baris =', data.shape[0], 'Banyak Kolom =', data.shape[1])
```

```
Banyak Baris = 3259 Banyak Kolom = 6
```

```
# Melihat Struktur Data
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3259 entries, 0 to 3258
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Unnamed: 0          3259 non-null   int64 
 1   nama                3259 non-null   object
 2   harga               3259 non-null   object
 3   seller              3259 non-null   object
 4   terjual             3259 non-null   object
 5   tanggal_ambil_data  3259 non-null   object
dtypes: int64(1), object(5)
memory usage: 152.9+ KB
```

Setelah dilihat-lihat ternyata datanya lengkap (tidak ada yang kosong) dan ukuran datanya lumayan besar. Berikutnya data akan dimanipulasi supaya bisa diambil sari/insight dari data tersebut.

Analisis Awal
Sekilas data tersebut memiliki kolom yang tidak perlu sehingga dapat dibuang.
Perhatikan juga bahwa kolom harga dan terjual memiliki tipe data object. Ini merupakan suatu masalah karna tidak bisa diolah.

Jika dilihat sekilas pada kolom nama dan seller, nama-nama terdiri dari string yang bervariasi (ada yang kapital ada yang tidak). Perhatikan bahwa pada python atau bahasa apapun huruf kapital dan tidak adalah 2 variabel yang berbeda. Sehingga perlu diubah menjadi kapital semua atau huruf kecil semua.

Kembali lihat kolom nama, terdapat 2 buah hal menarik yang dapat kita ambil, yakni merk dan volume dari jenis minyak.

Dapat juga dibuat suatu kolom baru yaitu Revenue yang diperoleh dari perkalian antara kolom terjual dengan kolom harga.

Bersih - Bersih Biar Bisa Syukuran

Langkah penanganan Pertama akan dibuang kolom yang tidak perlu, menyamakan string nama dan seller (membuat semua string kapital atau tidak) dan string yang tidak membantu seperti Rp, terjual dan produk. Dan memastikan tanggal ambil

```
1 # Membuang Kolom Unnamed
2 data = data.drop('Unnamed: 0',axis=1)
3 data.head()
```

	nama	harga	seller	terjual	tanggal_ambil_data
0	Bertolli Extra Virgin Olive Oil / Minyak Zaitu...	Rp85.000	Food Republic	Terjual 1.779	2020-09-04

```
# Mengubah semua nama string menjadi huruf kecil
data['nama'] = data['nama'].str.lower()
data['seller'] = data['seller'].str.lower()
```

```
# Membuat data Harga Menjadi Integer
harga = data['harga'].str.split('Rp')
harga = harga.str[1]
harga = harga.str.replace('.','')
harga = harga.astype(int)
data['harga'] = harga
data
```

	nama	harga	seller	terjual	tanggal_ambil_data
0	bertolli extra virgin olive oil / minyak zaitu...	85000	food republic	Terjual 1.779 Produk	2020-09-04
1	original extra virgin olive oil (evoo) casa di...	60000	yubyre	Terjual 1.624 Produk	2020-09-04
2	evoo baby olive oil casa di oliva olivia for k...	58000	papamama babyshop	Terjual 2.161 Produk	2020-09-04
3	casa di oliva - extra virgin olive oil for kid...	59000	chubby baby shop	Terjual 1.696 Produk	2020-09-04
4	best price casa di oliva extra virgin olive oi...	59900	whiz world	Terjual 3.115 Produk	2020-09-04
...	...	...	...	...	...
3254	minyak zaitun olive oil pak	70000	tokobundatitin	Terjual 34	2020-09-14

```
# Membuat data terjual Menjadi Integer
terjual =data['terjual'].str.split()
terjual = terjual.str[1]
terjual = terjual.str.replace('.','')
terjual = terjual.astype(int)
data['terjual'] = terjual
data
```

	nama	harga	seller	terjual	tanggal_ambil_data
0	bertolli extra virgin olive oil / minyak zaitu...	85000	food republic	1779	2020-09-04
1	original extra virgin olive oil (evoo) casa di...	60000	yubyre	1624	2020-09-04
2	evoo baby olive oil casa di oliva olivia for k...	58000	papamama babyshop	2161	2020-09-04
3	casa di oliva - extra virgin olive oil for kid...	59000	chubby baby shop	1696	2020-09-04
4	best price casa di oliva extra virgin olive oi...	59900	whiz world	3115	2020-09-04
...	...	...	...	...	...

```
# Utak Atik Tanggal Biar Seru Ehe
tanggal = data['tanggal_ambil_data'].str.split('-')
tahun = tanggal.str[0]
bulan = tanggal.str[1]
hari = tanggal.str[2]
data['tahun'] = tahun.astype(str)
data['bulan'] = bulan.astype(str)
data['hari'] = hari.astype(str)
```

Merk Semua Olive Oil

```
# Karna di matematika kalau ga nguli ga asyik ye ga
utik = data['nama']
b = [0]*len(data)
merk = ['fillipo','filippo ','bertol','casa','borges','yummy bites', 'pak haji',
        'mueloliva','zaituna','pomace','olio luglio','oilum','ayudya','olitalia',
        'luglio', 'cobram', 'sasso','rafael salgado','riche', 'orkide',
        'arbequina','nigella', 'mubarok','rs','syekh ali jaber','rumman',
        'forever arctic','costa','selva','acropolis','coosur','bragg','al arobi',
        'al afiat','ammara','coreysa','minyak kelapa voc','medina','co.e',
        'la rambla','dougo ','ghuroba','deep olive oil','mumtaz','balsari',
        'ecozest','amir','geofoods tartufo','orillia','naroop','afra','sesa',
        'orilia canola','az zaitun','desert miracle','sufi','jadied','jadeed',
        'espanola','altivolia','primarasa']
for i in range (len(merk)):
  a = utik.str.find(merk[i])
  for j in range(len(a)):
    if a[j] != -1 :
      b[j] = merk[i]
  for k in range(len(a)):
    if b[k] == 0:
      b[k] = 'lainnya'
# Karena ada beberapa nama produk yang 'spesial'
benar = ['bertolli','costa d oro','filippo','al ghuroba','le riche','al amir','jade
for l in range (len(b)):
  if b[l] == 'bertol':
    b[l]= benar[0]
```

```
  elif b[l] == 'costa':
    b[l]=benar[1]
  elif b[l] == 'fillipo':
    b[l] = benar[2]
  elif b[l] == 'ghuroba':
    b[l] = benar[3]
  elif b[l] == 'riche':
    b[l] =benar[4]
  elif b[l] == 'amir':
    b[l] =benar[5]
  elif b[l] == 'jadied':
    b[l] =benar[6]
data['merk'] = b
```

```
# Menemukan volum dari nama
c = ['tidak tersedia']*len(data)
ukuran = ['250', '500', '1','325']

for i in range (len(ukuran)):
  a = utik.str.find(ukuran[i])
  for j in range(len(a)):
    if a[j] != -1 :
      c[j] = ukuran[i]
      if c[j] == '1' :
        c[j] = '1000'
data['ukuran'] = c
```

```
# Membuat kolom revenue
data['revenue'] = data['terjual']*data['harga']
```

Setelah bermalam-malam dan bergelas-gelas kopi yang telah diminum akhirnya datanya bersih juga. Perhatikan output dari cell dibawah ini

```
# Rekap
print('Banyaknya Seller ',len(data['seller'].unique()))
print('Banyaknya Barang ',len(data['nama'].unique()))
print('Banyaknya Merk ',len(data['merk'].unique()))
print('Range pengumpulan Data', data['tanggal_ambil_data'].unique())
```

```
Banyaknya Seller  211
Banyaknya Barang  416
Banyaknya Merk  57
Range pengumpulan Data ['2020-09-04' '2020-09-05' '2020-09-07' '2020-09-09' '2020
 '2020-09-11' '2020-09-14']
```

Yang menarik dari data adalah, terdapat suatu produk yang bukan olive oil atau sejenisnya. Yaitu prima rasa. Produk tersebut adalah sambel untuk makan bukan untuk memasak. Jadi... tidak begitu signifikan jika diubang karna setelah diperiksa produk prima rasa ada 14 dari 3259 data. Jadi dibiarkan saja

Pertanyaan Pertama

Pada rentang waktu tersebut, siapa yang menjadi market leader?

Yang menjadi Market Leader adalah Seller yang paling banyak melakukan penjualan. Sehingga langkah pertama adalah menghitung jumlah banyak terjual dari seller

```
market_leader = data.terjual.groupby(data['seller'],axis=0).sum().sort_values(ascen
market_leader.head(1)
```

```
seller
food republic    40954
Name: terjual, dtype: int64
```

Pertanyaan Kedua

Brand minyak mana yang memiliki sales value terbesar saat big sale 9.9?

Pertama akan dikelompokkan pada waktu yang ditentukan dan akan dikelompokkan berdasarkan merk dan dijumlahkan jumlah terjualnya.

```
big_sale = data[data['tanggal_ambil_data']=='2020-09-09']
big_sale['revenue'] = big_sale['harga']*big_sale['terjual']
minyak_paling_laris = big_sale.groupby('merk',axis=0)
minyak_paling_laris['revenue'].max().sort_values(ascending=False).head(1)
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:2: SettingWithCopyWa
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl

merk
bertolli    209455000
Name: revenue, dtype: int64
```

Pertanyaan Ketiga

Bayangkan kalian bekerja di salah satu brand minyak tersebut, dari data ini hal apa yang bisa Anda sarankan ke tim sales / marketing?

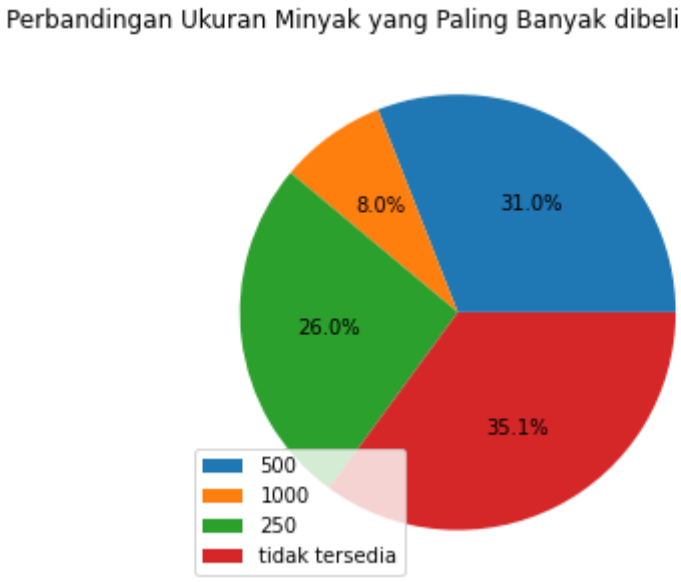
Misalkan merk yang akan ditinjau adalah Bertolli.
Pertama akan dilihat terlebih dulu karakteristik dari data.

```
bertolli = data[data.merk=='bertolli']
print('Banyak Seller dari Merk Bertolli',len(bertolli.seller.unique()))
bertolli.seller.unique()
```

```
Banyak Seller dari Merk Bertolli 35
array(['food republic', 'ceva', 'ss suppliers f&b jakarta', 'bayininja',
       'bestcmart', 'momogo id', 'w-jaya store', 'warung mas budi',
       'theonlinegrocer', 'foodsupply.co', 'w2fithealthy', 'herbs & co',
       'househerbal', 'genki plant', 'house of organix', 'niconicoshop',
       'kurakushop', 'serbasherbi', 'pd laris', 'the spices house',
       'secondition', 'jfm26', 'ampunmurahnya', 'greenara.id',
       'philocoffee', '888 seasoning', 'grandstand', 'provision master',
       'khas jaya nusantara', 'toko 3 rasa', "danieltyo's shop",
       "chef'schoice", 'choconola', 'disgro', 'dapoer canoli'],
      dtype=object)
```

Berikutnya akan dianalisis ukuran dari merk bertolli yang dijual.

```
label_analisis = bertolli.ukuran.unique()
analisis = bertolli.groupby('ukuran')
plt.figure(figsize=(5,5))
plt.pie(analisis.revenue.sum(), autopct='%1.1f%%')
plt.legend(label_analisis,loc='lower left')
plt.title('Perbandingan Ukuran Minyak yang Paling Banyak dibeli pada Merk Bertolli'
plt.show()
```

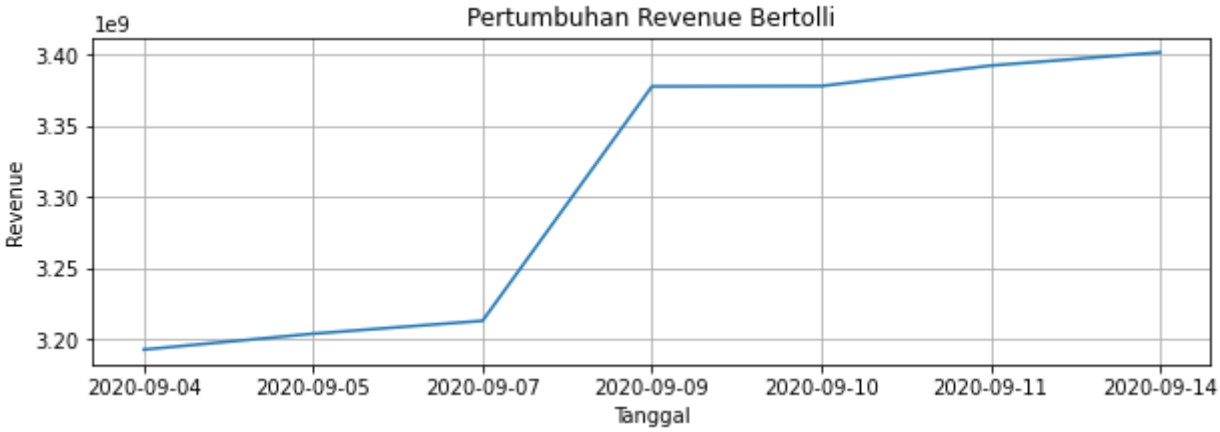


Perhatikan bahwa banyak dari produk Bertolli yang terjual masih kurang informasi mengenai ukuran dari produk yang dibeli.

Berikutnya akan dilihat pertumbuhan dari revenue Bertolli

```
bertolli = bertolli.groupby('tanggal_ambil_data')
plt.figure(figsize=(10, 3))
plt.plot(bertolli['revenue'].sum())
```

```
plt.axis()
plt.title('Pertumbuhan Revenue Bertolli')
plt.xlabel('Tanggal')
plt.ylabel('Revenue')
plt.grid(True)
plt.show()
```



Setelah melihat data dari merk Bertolli akan dilihat data secara keseluruhan. Kita akan tinjau dari segi revenuenya.

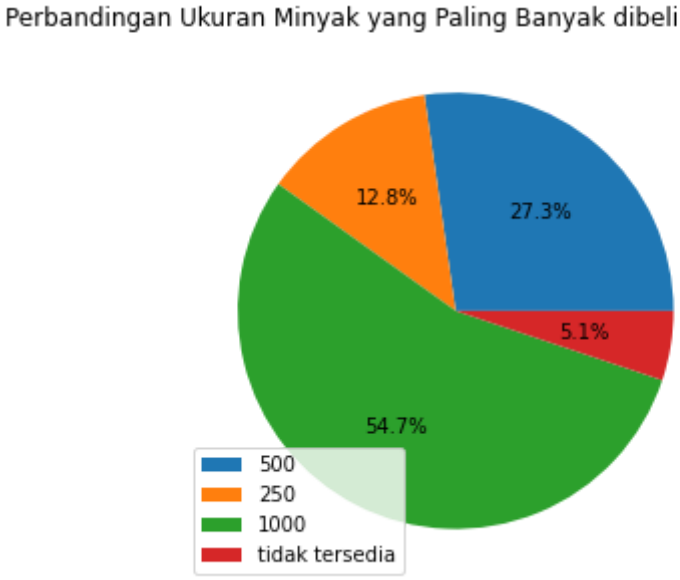
```
merk = ['bertolli','casa','borges','lainnya']
for i in range(len(b)):
  if b[i] != merk[0]:
    if b[i] != merk[1]:
      if b[i] !=merk[2]:
        b[i]=merk[3]
data13 =data
data13['merk'] = b
```

```
data_acuan = data13.groupby('merk')
labels = data_acuan['merk'].unique()

plt.figure(figsize=(5,5))
plt.pie(data_acuan['revenue'].sum().sort_values(ascending=False), autopct='%1.1f%%'
plt.legend(labels,loc='lower left')
plt.title('Perbandingan Reveneu antara 4 merk teratas')
plt.show()
```

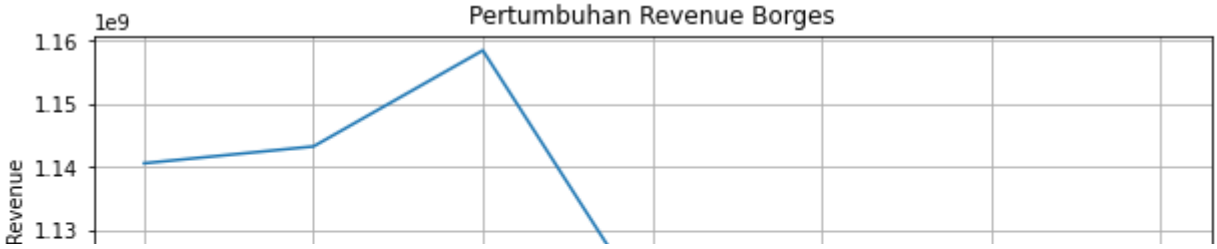
Ternyata merk Bertolli berbeda tipis dengan Borges. Akan dilihat data borges

```
1 borges = data[data.merk=='borges']
2 label_borges = borges.ukuran.unique()
3 analisis2 = borges.groupby('ukuran')
4 plt.figure(figsize=(5,5))
5 plt.pie(analisis2.revenue.sum(), autopct='%1.1f%%')
6 plt.legend(label_borges,loc='lower left')
7 plt.title('Perbandingan Ukuran Minyak yang Paling Banyak dibeli pada Merk Borges')
8 plt.show()
```



Perhatikan bahwa banyak dari produk Borges yang terjual paling banyak adalah 1000ml dan perhatikan bahwa merk yang tidak memiliki ukuran hanya 5.1%.

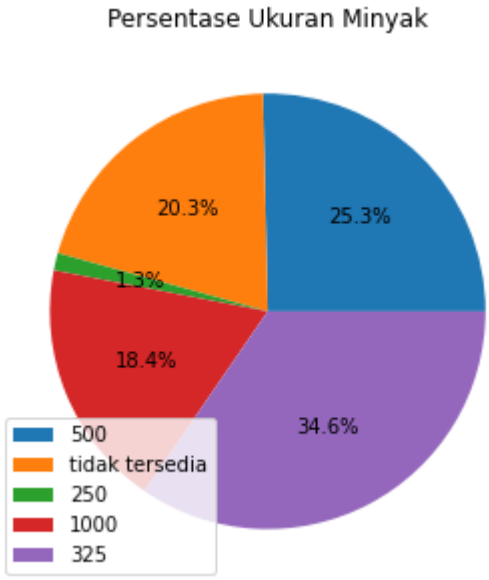
```
1 borges = borges.groupby('tanggal_ambil_data')
2 plt.figure(figsize=(10, 3))
3 plt.plot(borges['revenue'].sum())
4 plt.axis()
5 plt.title('Pertumbuhan Revenue Borges')
6 plt.xlabel('Tanggal')
7 plt.ylabel('Revenue')
8 plt.grid(True)
9 plt.show()
```



Perhatikan kembali bahwa merk Borges mengalami penurunan yang signifikan pada tanggal 9 September 2020. Hal ini berkebalikkan pada merk borges, karena secara umum revenue borges terus naik. Kemungkinan besar, merk borges mengalami penurunan karena merk borges adalah barang inferior (Karena kemungkinan ada kecenderungan bahwa merk borges tidak menarik untuk dibeli pada event tertentu)

Berikutnya akan dicoba untuk dicocokkan antara analisis sementara dengan data yang lebih holistik

```
1 data_ukuran = pd.get_dummies(data['ukuran'])
2 labels1 = data['ukuran'].unique()
3 plt.figure(figsize=(5,5))
4 plt.pie(data_ukuran.sum(),autopct='%1.1f%%')
5 plt.legend(labels1,loc='lower left')
6 plt.title('Persentase Ukuran Minyak')
7 plt.show()
```




```
1 q = data[data['ukuran']=='325']
2 print('Rataan harga dari minyak ukuran 325ml', q.harga.mean())
3 q.harga.unique()
```

```
Rataan harga dari minyak ukuran 325ml 47083.333333333336
array([43500, 42000, 48000, 58000, 48500, 42500])
```

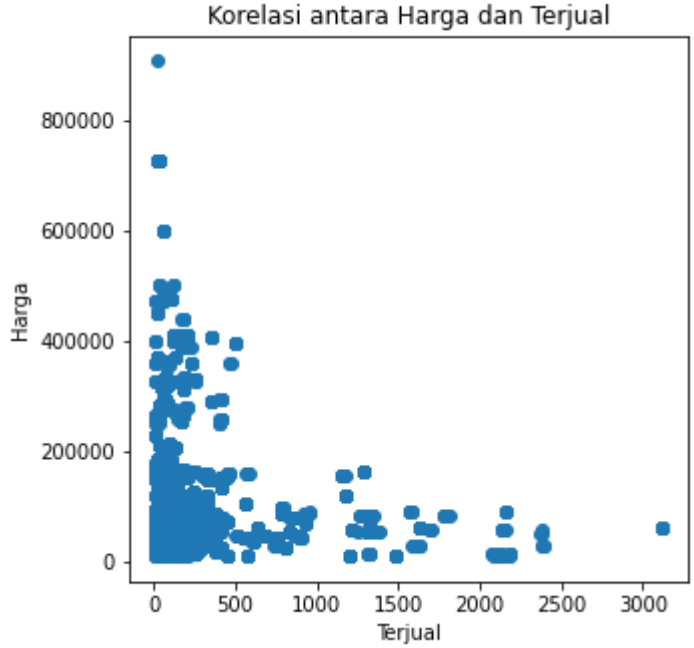
```
1 plt.figure(figsize=(10, 3))
2 q = q.groupby('tanggal_ambil_data')
```

```
plt.plot(q['revenue'].sum())
plt.axis()
plt.title('Pertumbuhan Revenue Untuk Minyak yang Berukuran 325ml')
plt.xlabel('Tanggal')
plt.ylabel('Revenue')
plt.grid(True)
plt.show()
```

Perhatikan bahwa ukuran yang paling banyak laku adalah ukuran 325 ml dengan harga di kisaran 47000 rupiah hal ini berbeda dengan merk bertolli dan merk borges yang penjualan terbanyaknya bukan pada kedua merk tersebut.

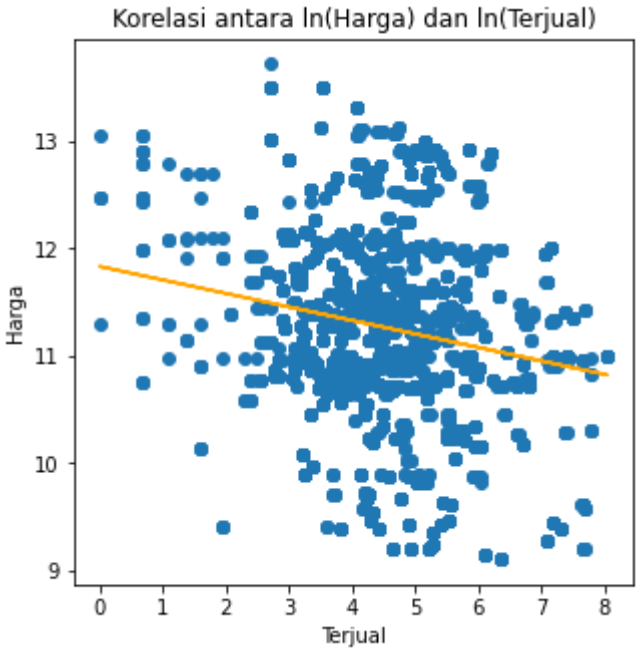
Berikutnya akan dilihat apakah terdapat korelasi antara harga dan jumlah barang yang dijual

```
plt.figure(figsize=(5,5))
plt.scatter(x=data['terjual'],y=data['harga'])
plt.title('Korelasi antara Harga dan Terjual')
plt.ylabel('Harga')
plt.xlabel('Terjual')
plt.show()
```



Ternyata tidak ada korelasi antara harga dan produk yang terjual. Tapi jika data kita transformasikan dengan menggunakan log natural diperoleh.

```
m, b = np.polyfit(np.log(data['terjual']), np.log(data['harga']), 1)
plt.figure(figsize=(5,5))
plt.scatter(x=np.log(data['terjual']),y=np.log(data['harga']))
plt.plot(np.log(data['terjual']),m*np.log(data['terjual'])+b, color='orange')
plt.title('Korelasi antara ln(Harga) dan ln(Terjual)')
plt.ylabel('Harga')
plt.xlabel('Terjual')
plt.show()
```



Perhatikan bahwa sedikit terlihat ada korelasi negatif pada ln(harga) terhadap ln(terjual)

Rekomendasi

Lebih gencar dengan penjualan dengan ukuran 325 ml.

LEBIH LANJUT

Ke depannya ada beberapa hal yang bisa dilakukan, salah satunya adalah implementasi Machine Learning untuk menentukan merk dari barang-barang yang tidak memiliki merk maupun ukuran. Idenya adalah menggunakan kemiripan nama ataupun menggunakan nama seller maupun harga yang ditetapkan. Jika memenuhi parameter yang telah disampaikan maka dapat membuat data menjadi lebih baik untuk diolah.

Demikian tugas ini saya buat, saya merasa saya masih kurang mengerti dan kurang paham

```
#f = data1.nama[data1.merk=='lainnya']
#f.iloc[125]
#merk = data['nama'].str.replace('oil','')
#merk = merk.str.replace('virgin','')
#merk = merk.str.replace('minyak','')
#merk = merk.str.replace('olive','')
#
#merk = data['nama'].str.replace('best','')
#merk = merk.str.replace('price','')
#merk = merk.str.split(n=0)
#merk = merk.str[0]
#merk
#data['merk'] = merk
#data['merk'].unique()
#f = f.str.replace('oil','')
#f = f.str.replace('virgin','')
#f = f.str.replace('minyak','')
#f = f.str.replace('olive','')
#f = f.str.replace('zaitun','')
#f.unique()
#len(f)
#q = data['nama'].str.find('ayudya')
#for i in range (len(q)):
#  if q[i] != -1:
#    q[i] == 1
#for j in range(len(q)):
#  if q[j] == -1:
#    q.pop(j)
#q
#data['tanggal_ambil_data'] = pd.to_datetime(data['tanggal_ambil_data'],format='%Y-
#data['nama'].iloc[450]
```