

ANALISIS PENJUALAN



Amin Rais

Pendahuluan

Mengeksplorasi data penjualan perusahaan untuk mengungkap wawasan penting tentang tren pasar dan kinerja produk. Analisis ini penting untuk pengambilan keputusan strategis dan mengidentifikasi peluang untuk pertumbuhan dan peningkatan penjualan

Dataset

df.head()

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERDATE	STATUS	PRODUCTLINE	PRODUCTCODE	CUSTOMERNAME	CITY	DEALSIZE
0	10100	30	100.00	2003-01-06	Shipped	Vintage Cars	S18_1749	Online Diecast Creations Co.	Nashua	Medium
1	10100	50	67.80	2003-01-06	Shipped	Vintage Cars	S18_2248	Online Diecast Creations Co.	Nashua	Medium
2	10100	22	86.51	2003-01-06	Shipped	Vintage Cars	S18_4409	Online Diecast Creations Co.	Nashua	Small
3	10100	49	34.47	2003-01-06	Shipped	Vintage Cars	S24_3969	Online Diecast Creations Co.	Nashua	Small
4	10101	25	100.00	2003-01-09	Shipped	Vintage Cars	S18_2325	Blauer See Auto, Co.	Frankfurt	Medium

Link dataset:

https://docs.google.com/spreadsheets/d/109O7gQVDGILaFe_6jZ8ttumYDzovRyPwTtDK7bGvOcM/edit?usp=sharing

Question?

- Which product lines have the highest and the lowest sales?
- Show sales performance overtime, is there any pattern?
- How does deal size correlate with total sales? what is the percentage of contribution for each type of deal?

Exploratory Data Analysis (EDA)

Content Data

ORDERNUMBER:
Pengidentifikasi unik untuk setiap pesanan penjualan.

QUANTITYORDERED:
Jumlah unit yang dipesan untuk produk tertentu.

PRICEEACH: Harga satu unit produk.

ORDERDATE: Tanggal saat pesanan diorder.

STATUS: Status pesanan terkini.

PRODUCTLINE: Kategori atau lini tempat produk berada.

PRODUCTCODE: Kode unik yang mengidentifikasi produk.

CUSTOMERNAME: Nama pelanggan yang memesan.

CITY: Kota tempat pelanggan berada.

DEALSIZE: Kategori ukuran transaksi.

Exploratory Data Analysis (EDA)

Preprocessing Data

Data Type :

Data type merupakan klasifikasi value suatu variabel yang membantu operasi komputer untuk mengenali data. Data type berfungsi agar bahasa pemrograman dapat mengetahui instruksi dalam pengolahan data.

ORDERDATE adalah satu-satunya data yang tipe nya tidak sesuai.

```
[ ] df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2824 entries, 0 to 2823
Data columns (total 10 columns):
#   Column             Non-Null Count  Dtype
---  -
0   ORDERNUMBER        2824 non-null  int64
1   QUANTITYORDERED    2824 non-null  int64
2   PRICEEACH          2824 non-null  float64
3   ORDERDATE          2824 non-null  object
4   STATUS             2824 non-null  object
5   PRODUCTLINE        2824 non-null  object
6   PRODUCTCODE        2824 non-null  object
7   CUSTOMERNAME       2824 non-null  object
8   CITY               2824 non-null  object
9   DEALSIZE           2824 non-null  object
dtypes: float64(1), int64(2), object(7)
memory usage: 220.8+ KB

df['ORDERDATE'] = pd.to_datetime(df['ORDERDATE'])
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2824 entries, 0 to 2823
Data columns (total 10 columns):
#   Column             Non-Null Count  Dtype
---  -
0   ORDERNUMBER        2824 non-null  int64
1   QUANTITYORDERED    2824 non-null  int64
2   PRICEEACH          2824 non-null  float64
3   ORDERDATE          2824 non-null  datetime64[ns]
4   STATUS             2824 non-null  object
5   PRODUCTLINE        2824 non-null  object
6   PRODUCTCODE        2824 non-null  object
7   CUSTOMERNAME       2824 non-null  object
8   CITY               2824 non-null  object
9   DEALSIZE           2824 non-null  object
dtypes: datetime64[ns](1), float64(1), int64(2), object(6)
memory usage: 220.8+ KB
```

Exploratory Data Analysis (EDA)

Preprocessing Data

Missing Value :

- Missing value adalah data yang hilang atau tidak tercatat dalam sebuah dataset.
- Missing value dapat menyebabkan bias atau ketidakakuratan.
- Data set ini tidak terdapat missing value.

```
[ ] df.isnull().sum()
```

	0
ORDERNUMBER	0
QUANTITYORDERED	0
PRICEEACH	0
ORDERDATE	0
STATUS	0
PRODUCTLINE	0
PRODUCTCODE	0
CUSTOMERNAME	0
CITY	0
DEALSIZE	0

dtype: int64

Exploratory Data Analysis (EDA)

Preprocessing Data

Duplicate Value :

- Duplicate value merupakan duplikat data yang terdapat dalam suatu dataset.
- Untuk mengatasi duplicate value adalah dengan menghapus data tersebut.
- Dalam dataset ini terdapat 1 duplicate value

▼ Duplicate Value

```
print (f' Nilai duplikat :', {df.duplicated().sum()})
```

```
➡ Nilai duplikat : {1}
```

```
[ ] df = df.drop_duplicates()  
print (f' Nilai duplikat :', {df.duplicated().sum()})
```

```
➡ Nilai duplikat : {0}
```


Exploratory Data Analysis (EDA)

Preprocessing Data

Status Column :

Kolom status berisi tentang informasi pengiriman atau pendistribusian orderan ke pelanggan. kolom status difilter dan diambil data yang memiliki status Shipped dan Resolved

```
jumlah_status = df['STATUS'].value_counts()
print(jumlah_status)
```

```
STATUS
Shipped    2617
Cancelled    60
Resolved    47
On Hold     44
In Process  41
Disputed    14
Name: count, dtype: int64
```

```
[ ] df_proses = df.copy()
# Filter status yang tidak diinginkan
status_tidak_diinginkan = ['Cancelled', 'On Hold', 'In Process', 'Disputed']
# Buat mask untuk memfilter status
mask_status = ~df_proses['STATUS'].isin(status_tidak_diinginkan)
# Terapkan mask ke dataframe
df_proses = df_proses[mask_status]
# Cetak jumlah baris sebelum dan sesudah filtering
print("Jumlah baris sebelum filtering:", len(df))
print("Jumlah baris sesudah filtering:", len(df_proses))
```

```
Jumlah baris sebelum filtering: 2823
Jumlah baris sesudah filtering: 2664
```

```
[ ] status = df_proses['STATUS'].value_counts()
print(status)
```

```
STATUS
Shipped    2617
Resolved    47
Name: count, dtype: int64
```

Exploratory Data Analysis (EDA)

Preprocessing Data

Revenue Column :

- Menambah kolom Revenue untuk mengidentifikasi total penjualan.
- $\text{Revenue} = \text{Priceeach} \times \text{Quantityordered}$

```
# Membuat kolom baru bernama REVENUE
df_proses['REVENUE'] = df_proses['QUANTITYORDERED'] * df_proses['PRICEEACH']
# Validasi data untuk memastikan kolom REVENUE memiliki nilai yang valid
print("Validasi data untuk memastikan kolom REVENUE memiliki nilai yang valid:")
print(df_proses['REVENUE'].describe())
df_proses.head()
```

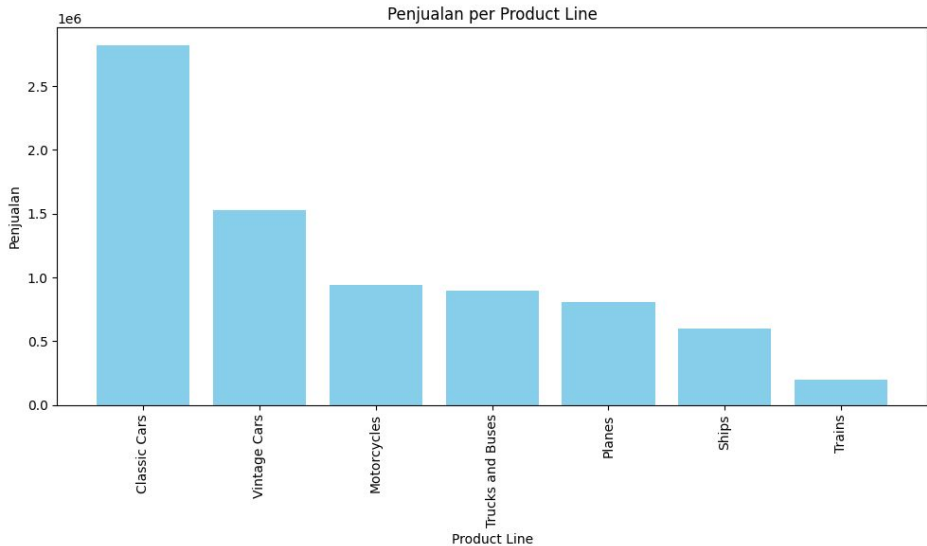
Validasi data untuk memastikan kolom REVENUE memiliki nilai yang valid:

```
count    2664.000000
mean     2922.300100
std      1877.855291
min       553.950000
25%      2100.000000
50%      2800.000000
75%      3700.000000
max      9048.160000
Name: REVENUE, dtype: float64
```

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERDATE	STATUS	PRODUCTLINE	PRODUCTCODE	CUSTOMERNAME	CITY	DEALSIZE	REVENUE
0	10100	30	100.00	2003-01-06	Shipped	Vintage Cars	S18_1749	Online Diecast Creations Co.	Nashua	Medium	3000.00
1	10100	50	67.80	2003-01-06	Shipped	Vintage Cars	S18_2248	Online Diecast Creations Co.	Nashua	Medium	3390.00
2	10100	22	86.51	2003-01-06	Shipped	Vintage Cars	S18_4409	Online Diecast Creations Co.	Nashua	Small	1903.22
3	10100	49	34.47	2003-01-06	Shipped	Vintage Cars	S24_3969	Online Diecast Creations Co.	Nashua	Small	1689.03
4	10101	25	100.00	2003-01-09	Shipped	Vintage Cars	S18_2325	Blauer See Auto, Co.	Frankfurt	Medium	2500.00

Productline With the Highest and Lowest Sales

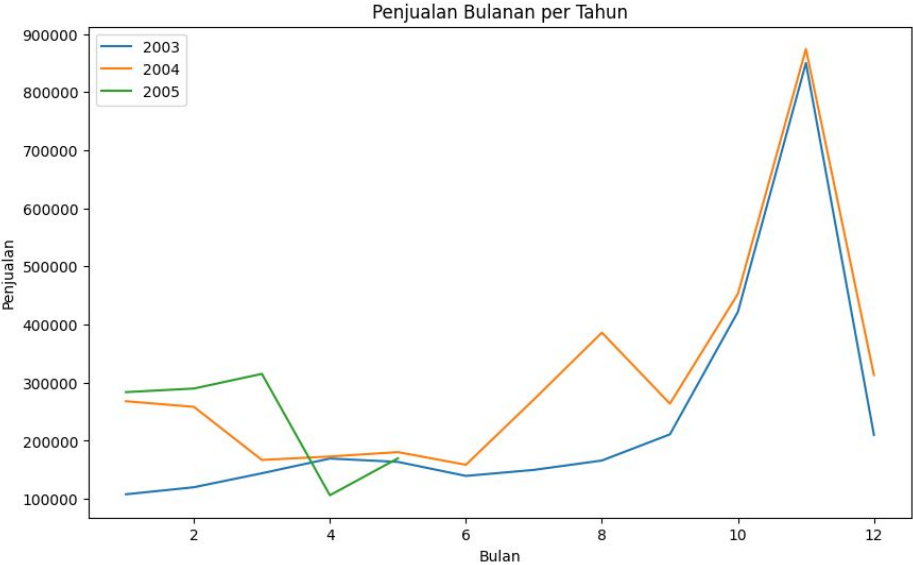
PRODUCTLINE	REVENUE
Classic Cars	2.817.515,14
Vintage Cars	1.528.213,53
Motorcycles	941.683,81
Trucks and Buses	896.017,45
Planes	807.105,12
Ships	599.668,69
Trains	194.804,26



Sales Performance Overtime

2003		2004	
BULAN	REVENUE	BULAN	REVENUE
1	107.885,96	1	268.015,87
2	120.036,80	2	258.389,05
3	144.096,23	3	166.958,56
4	169.421,03	4	172.935,80
5	163.654,12	5	180.469,64
6	139.552,84	6	158.696,24
7	149.869,73	7	271.103,61
8	166.026,32	8	386.106,59
9	211.045,86	9	263.854,26
10	421.885,09	10	452.796,13
11	850.203,27	11	874.092,26
12	210.117,21	12	313.055,93

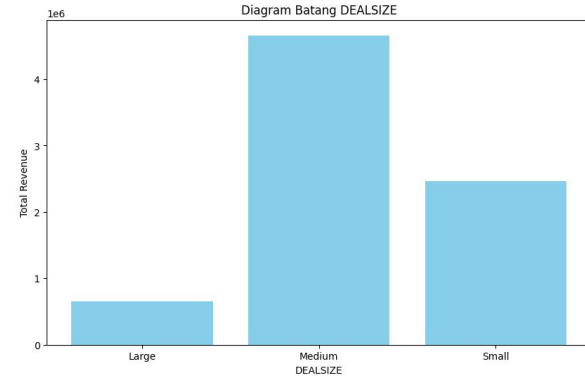
2005	
BULAN	REVENUE
1	283.680,46
2	289.902,30
3	315.131,94
4	106.220,56
5	169.804,34



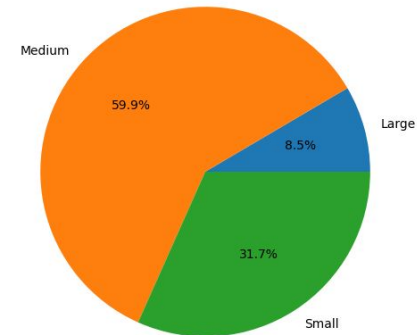
Correlation DEALSIZE with Total Sales

DEALSIZE	REVENUE
Large	658.548,16
Medium	4.660.310,00
Small	2.466.149,84

- Tidak ada korelasi yang menunjukkan semakin besar Dealsize semakin besar penjualannya maupun sebaliknya
- Medium merupakan Dealsize dengan penjualan tertinggi diikuti dengan Small dan Large



Total Penjualan Berdasarkan DEALSIZE



Conclusion

- Productline Classic Cars memiliki penjualan yang paling tinggi dan berbeda jauh dari Productline lainnya, ditakutkan ketika penjualan Classic Cars menurun maka total penjualan perusahaan akan mengalami penurunan yang signifikan, sehingga perusahaan perlu meningkatkan penjualan pada Productline lainnya.
- Penjualan terendah terjadi antara April dan Mei, perusahaan membutuhkan strategi penjualan demi meningkatkan penjualan pada bulan-bulan tersebut, dan pada November terjadi kenaikan penjualan yang sangat signifikan sehingga perlu diidentifikasi penyebab kenaikan tersebut sehingga perusahaan dapat menjaga bahkan meningkatkan penjualan pada bulan tersebut.