

**LAPORAN PRAKTIKUM**  
**PRAKTIK PEMROGRAMAN PYTHON**  
**Data Visualization**



**Disusun oleh :**

**Yunita Kartika Putri (V3923019)**

**Dosen : Yusuf Fadila Rachman. S.Kom., M.Kom**

**PS D-III TEKNIK INFORMATIKA**  
**SEKOLAH VOKASI**  
**UNIVERSITAS SEBELAS MARET**  
**2024**

## HASIL DAN PEMBAHASAN

### 1. Membaca File CSV

#### Sourcode

```
1 import pandas as pd
2
3 # reading the database
4 data = pd.read_csv("Data Sales3.csv", delimiter = ";")
5
6 # printing the top 10 rows
7 display(data.head(10))
8
9
```

Kode yang diberikan menggunakan pustaka `pandas` untuk membaca file CSV bernama "Data Sales3.csv" dengan menggunakan fungsi `read\_csv()`. File CSV tersebut memiliki delimiter (pemisah) yang disetel sebagai ";" menggunakan argumen `delimiter`. Setelah data dibaca, fungsi `head(10)` digunakan untuk menampilkan 10 baris pertama dari data. Fungsi `display()` digunakan untuk menampilkan output dengan format yang lebih rapi daripada hanya menggunakan `print()`. Dengan cara ini, pengguna dapat dengan mudah melihat struktur dan isi dari data dalam file CSV tersebut.

#### Output

	Order ID	Month	Year	City	Segment	Category	Sub-Category	Product Name	Sales	Quantity	Profit
0	CA-2014-156587	Maret	2014	DKI Jakarta	Consumer	Furniture	Chairs	Global Push Button Manager's Chair, Indigo	\$48,71	1	\$5,48
1	CA-2014-156587	Maret	2014	Jawa Barat	Consumer	Office Supplies	Storage	Carina 42"Hx23 3/4"W Media Storage Unit	\$242,94	3	\$4,86
2	CA-2014-152905	Februari	2014	Banten	Consumer	Office Supplies	Storage	Akro Stacking Bins	\$12,62	2	\$2,52
3	CA-2014-156587	Maret	2014	Jawa Tengah	Consumer	Office Supplies	Art	Newell 330	\$17,94	3	\$4,66
4	CA-2014-157644	Desember	2014	DI Yogyakarta	Corporate	Technology	Accessories	Verbatim 25 GB 6x Blu-ray Single Layer Recordable	\$34,77	3	\$11,47
5	CA-2014-113768	Mei	2014	Jawa Timur	Corporate	Furniture	Chairs	Iceberg Nesting Folding Chair, 19w x 6d x 43h	\$279,46	6	\$20,96
6	CA-2014-122070	April	2014	DKI Jakarta	Corporate	Office Supplies	Binders	ACCOHIDE 3-Ring Binder, Blue, 1"	\$9,91	3	\$3,35
7	US-2014-158400	Oktober	2014	Jawa Barat	Corporate	Office Supplies	Binders	GBC VeloBind Cover Sets	\$49,41	4	\$18,53
8	CA-2014-113768	Mei	2014	Banten	Corporate	Office Supplies	Paper	EcoTones Memo Sheets	\$8,00	2	\$3,84
9	CA-2014-122070	April	2014	Jawa Tengah	Corporate	Office Supplies	Envelopes	Staple envelope	\$247,84	8	\$121,44

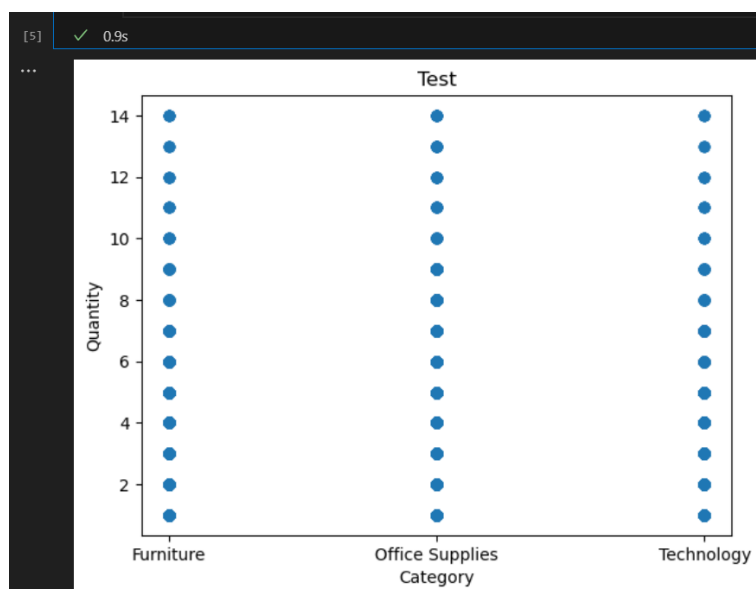
## 2. GRAFIK SCATTER PLOT

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # reading the database
5 data = pd.read_csv("Data Sales3.csv", delimiter = ";")
6
7 # Scatter plot w day against tip
8 plt.scatter(data['Category'], data['Quantity'])
9
10 # Adding Title to the plot
11 plt.title("Test")
12
13 # Setting the X and Y labels
14 plt.xlabel('Category')
15 plt.ylabel('Quantity')
16
17 # Save the plot as a PNG file
18 plt.savefig('scatter_plot.png', dpi=300, bbox_inches='tight')
19
20 plt.show()
21
```

Kode yang diberikan menggunakan pustaka `pandas` untuk membaca file CSV bernama "Data Sales3.csv" dengan menggunakan fungsi `read\_csv()`. Data tersebut kemudian digunakan untuk membuat scatter plot dengan menggunakan fungsi `scatter()` dari pustaka `matplotlib.pyplot`. Pada sumbu x (horizontal), kita menggunakan data dari kolom 'Category', sedangkan pada sumbu y (vertikal), kita menggunakan data dari kolom 'Quantity'. Setelah plot dibuat, judul ditambahkan menggunakan fungsi `title()`, dan label sumbu x dan y ditentukan menggunakan fungsi `xlabel()` dan `ylabel()`.

Plot yang telah dibuat kemudian disimpan sebagai file PNG dengan nama 'scatter\_plot.png' menggunakan fungsi `savefig()`. Argumen `dpi=300` digunakan untuk menentukan resolusi gambar (dots per inch), dan `bbox\_inches='tight'` digunakan untuk memastikan bahwa area gambar yang disimpan tidak terpotong. Terakhir, fungsi `show()` digunakan untuk menampilkan plot tersebut. Dengan cara ini, pengguna dapat dengan mudah membuat dan menyimpan scatter plot dari data dalam file CSV.

### Output



### 3. Grafik Line Plot

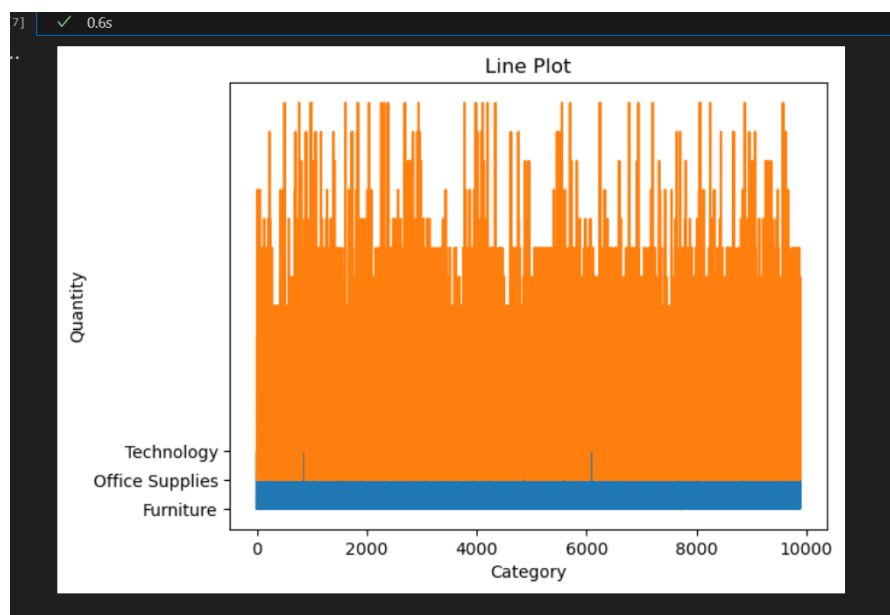
#### Sourcode

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # reading the database
5 data = pd.read_csv("Data Sales3.csv", delimiter = ";")
6
7 # Scatter plot w day against tip
8 plt.plot(data['Category'])
9 plt.plot(data['Quantity'])
10
11
12 # Adding Title to the plot
13 plt.title("Line Plot")
14
15 # Setting the X and Y labels
16 plt.xlabel('Category')
17 plt.ylabel('Quantity')
18
19 # Save the plot as a PNG file
20 plt.savefig('line.png', dpi=300, bbox_inches='tight')
21
22 plt.show()
23
```

Kode ini menggunakan pustaka 'pandas' untuk membaca file CSV "Data Sales3.csv" menggunakan fungsi 'read\_csv()'. Setelah data dibaca, plot garis dibuat menggunakan fungsi 'plot()' dari 'matplotlib.pyplot'. Pada sumbu x (horizontal), data dari kolom 'Category' digunakan, sedangkan pada sumbu y (vertikal), data dari kolom 'Quantity' digunakan.

Setelah plot dibuat, judul ditambahkan menggunakan fungsi 'title()', dan label sumbu x dan y ditetapkan menggunakan fungsi 'xlabel()' dan 'ylabel()'. Plot yang dihasilkan kemudian disimpan sebagai file PNG dengan nama 'line.png' menggunakan fungsi 'savefig()'. Argumen 'dpi=300' digunakan untuk menentukan resolusi gambar (dots per inch), dan 'bbox\_inches='tight'' digunakan untuk memastikan bahwa area gambar yang disimpan tidak terpotong. Terakhir, fungsi 'show()' digunakan untuk menampilkan plot tersebut. Dengan cara ini, pengguna dapat membuat dan menyimpan plot garis berdasarkan data dalam file CSV.

#### Ouput



## 4. GRAFIK BAR CHART

### Sourcode

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # Membaca data dari file CSV
5 data = pd.read_csv("Data Sales3.csv", delimiter=";")
6
7 # Menghitung jumlah Quantity untuk setiap Category
8 category_counts = data.groupby('Category')['Quantity'].sum()
9
10 # Menggunakan bar plot untuk memvisualisasikan jumlah Quantity tiap Category
11 plt.figure(figsize=(10, 6)) # Mengatur ukuran gambar plot
12 plt.bar(category_counts.index, category_counts.values)
13
14 # Menambahkan judul dan label sumbu
15 plt.title("Total Quantity per Category")
16 plt.xlabel('Category')
17 plt.ylabel('Total Quantity')
18
19 # Memutar label sumbu x jika diperlukan agar lebih mudah dibaca
20 plt.xticks(rotation=45, ha='right')
21
22 # Menyimpan plot sebagai file PNG dengan resolusi tinggi
23 plt.savefig('bar.png', dpi=300, bbox_inches='tight')
24
25 # Menampilkan plot
26 plt.show()
27
```

Kode ini menggunakan pustaka `pandas` untuk membaca data dari file CSV "Data Sales3.csv" dengan menggunakan fungsi `read\_csv()`, dan data tersebut disimpan dalam variabel `data`. Selanjutnya, kode menghitung jumlah kuantitas (`Quantity`) untuk setiap kategori (`Category`) dengan menggunakan fungsi `groupby()` dan `sum()` dari pustaka `pandas`.

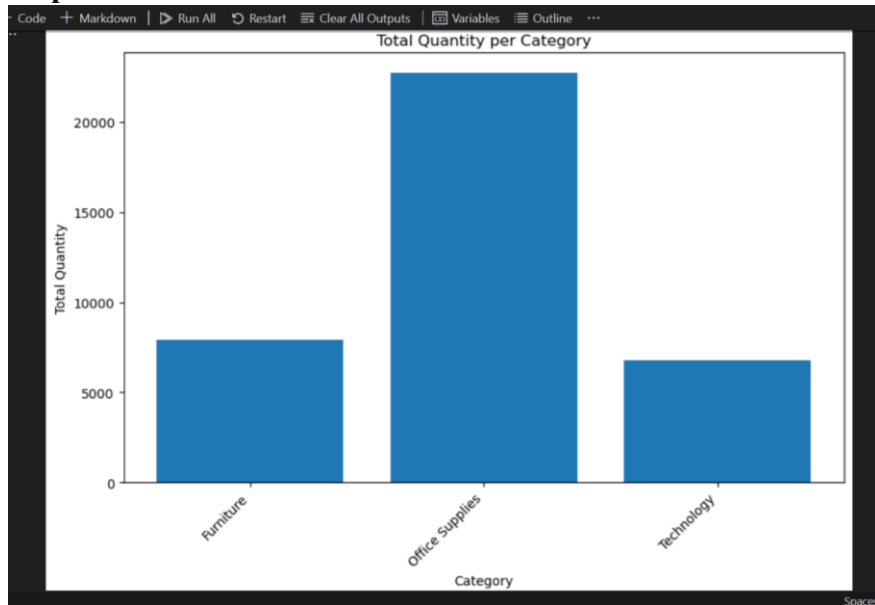
Setelah jumlah kuantitas tiap kategori dihitung, plot batang (`bar plot`) digunakan untuk memvisualisasikan jumlah kuantitas untuk setiap kategori. Ukuran gambar plot diatur dengan `plt.figure(figsize=(10, 6))`, kemudian fungsi `bar()` digunakan untuk membuat plot batang.

Judul plot dan label sumbu (`x` dan `y`) ditambahkan menggunakan fungsi `title()`, `xlabel()`, dan `ylabel()`. Fungsi `xticks()` digunakan untuk memutar label sumbu `x` jika diperlukan agar lebih mudah dibaca.

Plot yang dihasilkan kemudian disimpan sebagai file PNG dengan nama 'bar.png' menggunakan fungsi `savefig()`. Argumen `dpi=300` digunakan untuk menentukan resolusi gambar (dots per inch), dan `bbox\_inches='tight'` digunakan untuk memastikan bahwa area gambar yang disimpan tidak terpotong.

Terakhir, fungsi `show()` digunakan untuk menampilkan plot tersebut. Dengan cara ini, pengguna dapat membuat dan menyimpan plot batang yang menunjukkan total kuantitas per kategori berdasarkan data dari file CSV.

## Ouput

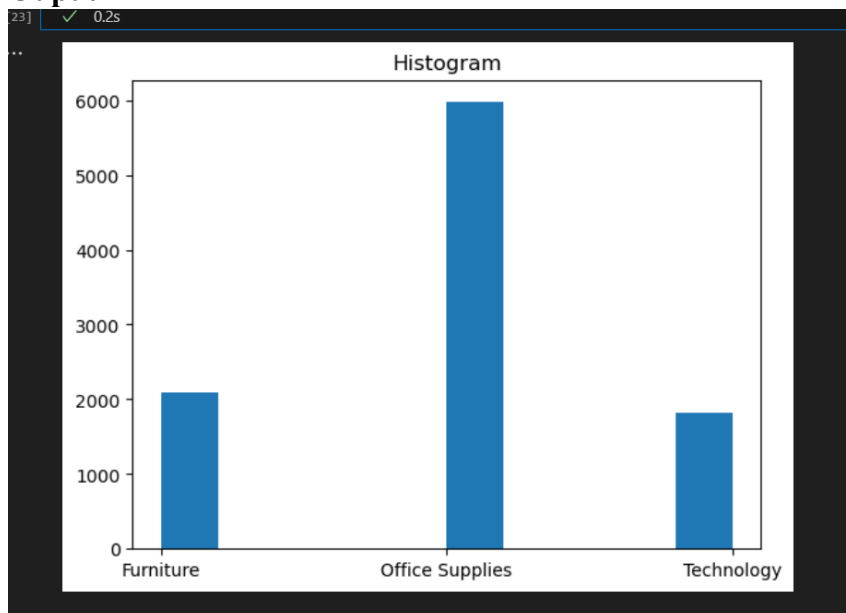


## 5. Grafik Histogram

### Sourcode

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # reading the database
5 data = pd.read_csv("Data Sales3.csv", delimiter = ";")
6
7 # Scatter plot w day against tip
8 plt.hist(data['Category'])
9
10
11 # Adding Title to the plot
12 plt.title("Histogram")
13
14 # Save the plot as a PNG file
15 plt.savefig('histogram.png', dpi=300, bbox_inches='tight')
16
17 plt.show()
18
```

## Ouput



## 6. Grafik Pie Chart

### Sourcode

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 # reading the database
5 data = pd.read_csv("Data Sales3.csv", delimiter = ";")
6
7 # Scatter plot w day against tip
8 sales = ['Category', 'Quantity']
9 datasales = [23, 10]
10
11 plt.pie(datasales, labels=sales)
12
13 plt.title("Sales Data")
14
15 # Save the plot as a PNG file
16 plt.savefig('pie.png', dpi=300, bbox_inches='tight')
17
18 plt.show()
```

### Ouput

