Nama : Sandria Amelia Putri

NPM : 2083010005

Kelas : Data Wrangling A

1. Load Pustaka/Library yang dibutuhkan.
   Library pandas berfungsi untuk memanipulasi dan menganalisis data secara efisien. Library ini menyediakan struktur data yang fleksibel dan efisien untuk mengolah data tabular (seperti tabel pada database atau spreadsheet), yaitu DataFrame dan Series. Library numpy berfungsi sebagai dasar bagi komputasi numerik di Python. Library matplotlib berfungsi untuk membuat visualisasi statis, animasi, dan interaktif dengan Python. Pyplot merupakan modul dari matplotlib yang menyediakan interface tingkat tinggi untuk membuat berbagai bagan dan plot, termasuk line plots, scatter plots, bar plots, histograms, dan lainnya.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```
005_Sandria Amelia Putri

2. Tampilkan Boston Housing dataset (.csv file).
   Gunakan fungsi pd.read_csv() untuk membaca file csv Boston Housing. Dataset ini akan tersimpan pada variabel baru yaitu boston_housing.

```
boston_housing = pd.read_csv(r"C:\Users\Sandria\Python\Semester 4\Data Wrangling\Boston_housing.csv")
boston_housing
```
005_Sandria Amelia Putri

   Berikut merupakan outputnya.

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	PRICE
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21.0	391.99	9.67	22.4
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21.0	396.90	9.08	20.6
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21.0	396.90	5.64	23.9
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21.0	393.45	6.48	22.0
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21.0	396.90	7.88	11.9

005_Sandria Amelia Putri

506 rows × 14 columns

3. Periksa 10 baris pertama. Temukan total baris dari dataset tersebut.
   Gunakan fungsi head(10) untuk memeriksa 10 baris pertama dataset.

```
boston_housing.head(10)
```
005_Sandria Amelia Putri

   Berikut merupakan outputnya.

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	PRICE
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
5	0.02985	0.0	2.18	0	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	28.7
6	0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.60	12.43	22.9
7	0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.90	19.15	27.1
8	0.21124	12.5	7.87	0	0.524	5.631	100.0	6.0821	5	311	15.2	386.63	29.93	16.5
9	0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.10	18.9

005_Sandria Amelia Putri

   Gunakan fungsi len() untuk memeriksa jumlah total suatu data.

```
print("Total baris: ", len(boston_housing.head(10)))
```
005_Sandria Amelia Putri

Berikut merupakan outputnya, didapatkan total baris dari boston_housing.head(10) adalah 10 baris.

005_Sandria Amelia Putri

```
Total baris:  10
```

Gunakan fungsi len() untuk memeriksa jumlah total suatu data.

```
print("Total baris: ", len(boston_housing))
```

005_Sandria Amelia Putri

Berikut merupakan outputnya, didapatkan total baris dari keseluruhan Boston Housing adalah 506 baris.

005_Sandria Amelia Putri

```
Total baris:  506
```

4. Buat DataFrame yang lebih kecil dengan kolom yang tidak terdapat CHAS, NOX, B, dan LSTAT.

   Variabel boston_housing2 disiapkan untuk menyimpan dataframe terbaru setelah membuang beberapa kolom yang diminta. Gunakan fungsi drop() untuk membuang beberapa kolom yaitu kolom CHAS, NOX, B, dan LSTAT. axis=1 berarti kolom akan dibuang sepanjang axis horizontal.

```
boston_housing2 = boston_housing.drop(['CHAS', 'NOX', 'B', 'LSTAT'], axis=1)
boston_housing2
```

005_Sandria Amelia Putri

Berikut merupakan outputnya, terlihat bahwa 4 kolom yang diminta sudah terhapus sehingga tersisa 10 kolom.

	CRIM	ZN	INDUS	RM	AGE	DIS	RAD	TAX	PTRATIO	PRICE
0	0.00632	18.0	2.31	6.575	65.2	4.0900	1	296	15.3	24.0
1	0.02731	0.0	7.07	6.421	78.9	4.9671	2	242	17.8	21.6
2	0.02729	0.0	7.07	7.185	61.1	4.9671	2	242	17.8	34.7
3	0.03237	0.0	2.18	6.998	45.8	6.0622	3	222	18.7	33.4
4	0.06905	0.0	2.18	7.147	54.2	6.0622	3	222	18.7	36.2
...	...	...	...	...	...	...	...	005_Sandria Amelia Putri		
501	0.06263	0.0	11.93	6.593	69.1	2.4786	1	273	21.0	22.4
502	0.04527	0.0	11.93	6.120	76.7	2.2875	1	273	21.0	20.6
503	0.06076	0.0	11.93	6.976	91.0	2.1675	1	273	21.0	23.9
504	0.10959	0.0	11.93	6.794	89.3	2.3889	1	273	21.0	22.0
505	0.04741	0.0	11.93	6.030	80.8	2.5050	1	273	21.0	11.9

506 rows × 10 columns

5. Periksa 7 baris terakhir dari DataFrame yang baru.

   Gunakan fungsi tail(7) untuk memeriksa 7 baris terakhir dataset.

005_Sandria Amelia Putri

```
boston_housing2.tail(7)
```

Berikut merupakan outputnya.

	CRIM	ZN	INDUS	RM	AGE	DIS	RAD	TAX	PTRATIO	PRICE
499	0.17783	0.0	9.69	5.569	73.5	2.3999	6	391	19.2	17.5
500	0.22438	0.0	9.69	6.027	79.7	2.4982	6	391	19.2	16.8
501	0.06263	0.0	11.93	6.593	69.1	2.4786	1	273	21.0	22.4
502	0.04527	0.0	11.93	6.120	76.7	2.2875	1	273	21.0	20.6
503	0.06076	0.0	11.93	6.976	91.0	2.1675	1	273	21.0	23.9
504	0.10959	0.0	11.93	6.794	89.3	2.3889	1	273	21.0	22.0
505	0.04741	0.0	11.93	6.030	80.8	2.5050	1	273	21.0	11.9

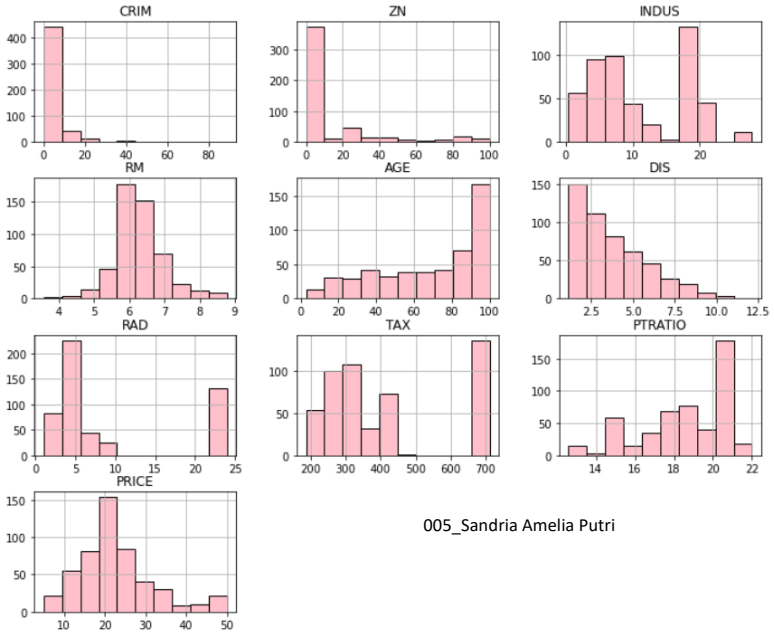
005_Sandria Amelia Putri

6. Gambarkan visualisasi histogram dari semua variable/kolom pada DataFrame yang baru.

   Untuk membuat histogram dari setiap kolom dataframe boston_housing2 gunakan fungsi hist() dari pandas. Figsize menyatakan ukuran figure, color dan edgecolor menyatakan warna bar histogram. Selanjutnya plt.show() digunakan untuk menampilkan histogram pada output.

```
boston_housing2.hist(figsize=(12,10), color='pink', edgecolor='k')
plt.show()
```
005_Sandria Amelia Putri

Berikut merupakan outputnya. Karena tidak menggunakan looping for maka output histogram setiap kolom akan muncul sekaligus.



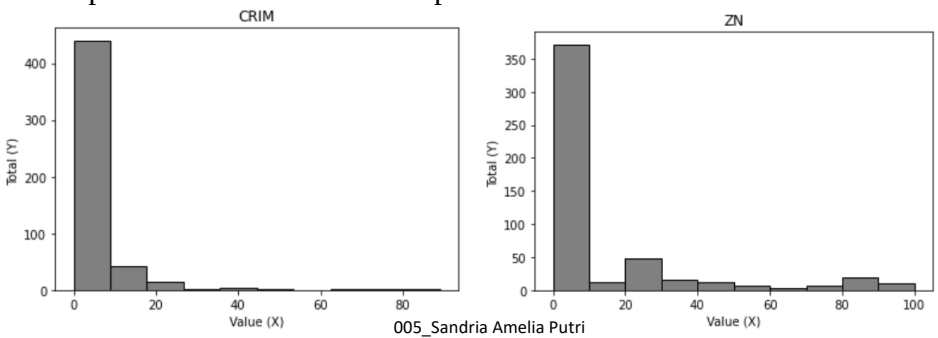
005_Sandria Amelia Putri

7. Gambarkan semua histogram menggunakan for loop. Tambahkan nama yang unik pada plot.

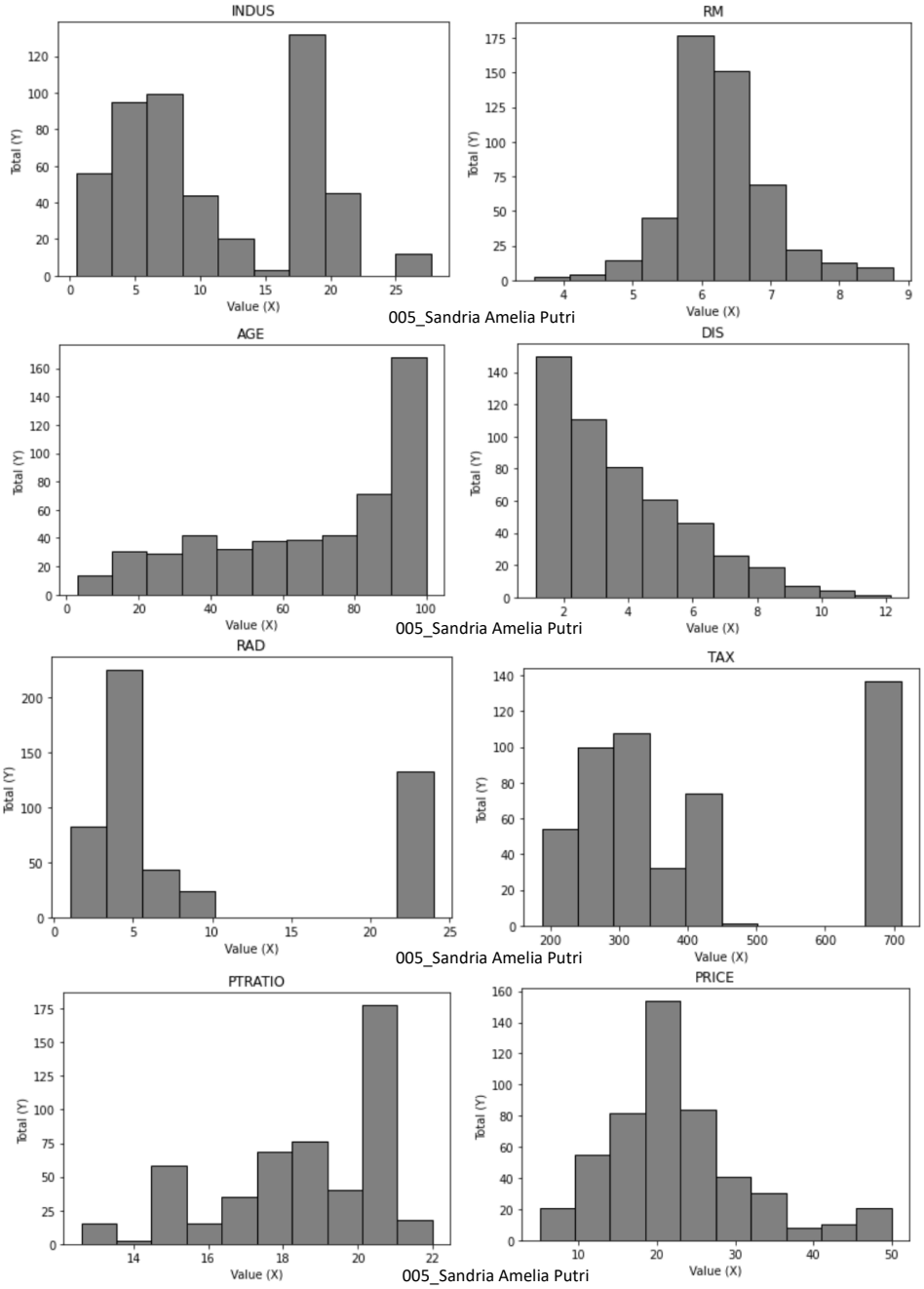
   Gunakan looping for plot in kolom boston_housing2. Kemudian gunakan fungsi plt.hist() untuk boston_housing2[plot] dengan color grey dan edgecolor k. Fungsi plt.title() digunakan untuk memberikan judul pada histogram. Fungsi plt.xlabel() digunakan untuk memberikan label pada nilai x dan fungsi plt.ylabel() digunakan untuk memberikan label pada nilai y. Selanjutnya plt.show() digunakan untuk menampilkan histogram pada output.

```
for plot in boston_housing2.columns:
    plt.hist(boston_housing2[plot], color='grey', edgecolor='k')
    plt.title(plot)
    plt.xlabel("Value (X)")
    plt.ylabel("Total (Y) ")
    plt.show()
```
005_Sandria Amelia Putri

Berikut merupakan outputnya. Karena dilakukan dengan looping for maka output histogram setiap kolom akan muncul satu per satu.



005_Sandria Amelia Putri

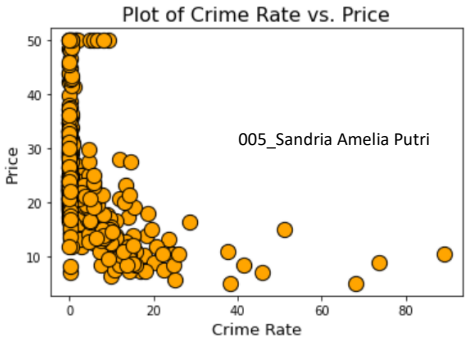


8. Buat scatter plot dari crime rate versus price.

   Gunakan fungsi plot.scatter() untuk membuat scatter plot hubungan antara crime rate (CRIM) dan price of housing (PROCING) di Boston. S menyatakan size=150, c menyatakan color, dan edgecolor menggunakan black. Fungsi plt.title() digunakan untuk memberikan judul pada scatter plot dengan ketentuan fontsize=16. Fungsi plt.xlabel() digunakan untuk memberikan label pada nilai x dan fungsi plt.ylabel() digunakan untuk memberikan label pada nilai y dengan ketentuan fontsize=13. Selanjutnya plt.show() digunakan untuk menampilkan scatter plot pada output.

```
boston_housing.plot.scatter('CRIM','PRICE',s=150,\
                            c='orange',edgecolor='k')
plt.title("Plot of Crime Rate vs. Price", fontsize=16)
plt.xlabel("Crime Rate",fontsize=13)
plt.ylabel("Price",fontsize=13)  005_Sandria Amelia Putri
plt.show()
```

   Berikut merupakan outputnya.

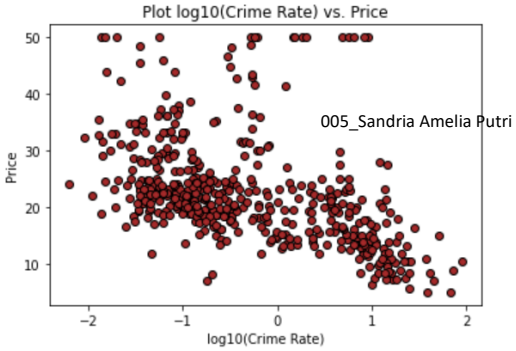


9. Buat Plot log10(crime) versus price.
   Variabel x disiapkan untuk menyimpan nilai dari log10(crime) dengan menggunakan fungsi np.log10(). Variabel y disiapkan untuk menyimpan nilai dari kolom price. Kemudian fungsi plt.scatter digunakan untuk membuat scatter plot hubungan antara log10(crime) dan price dengan color=brown dan edgecolor=black. Fungsi plt.xlabel() digunakan untuk memberikan label pada nilai x dan fungsi plt.ylabel() digunakan untuk memberikan label pada nilai y. Fungsi plt.title() digunakan untuk memberikan judul pada scatter plot. Selanjutnya plt.show() digunakan untuk menampilkan scatter plot pada output.

```
# mendefinisikan variabel x dan y
x = np.log10(boston_housing['CRIM'])
y = boston_housing['PRICE']
                                    005_Sandria Amelia Putri
# membuat plot
plt.scatter(x, y, color ='brown', edgecolor='k')

# memberi nama label x, y, dan judul
plt.xlabel('log10(Crime Rate)')
plt.ylabel('Price')
plt.title('Plot log10(Crime Rate) vs. Price')
plt.show()
```

   Berikut merupakan outputnya.



10. Kalkulasikan statistic yang berguna, seperti mean rooms per dwelling, median age, mean distances to five Boston employment centers, and the percentage of houses with a low price (< $20,000).
    Variabel mean_rooms disiapkan untuk menyimpan hasil perhitungan dari nilai mean rooms per dwelling (rata-rata jumlah kamar per hunian) [RM]. Variabel median_age disiapkan untuk menyimpan hasil perhitungan dari nilai median age (umur setiap rumah) [age]. Variabel mean_distances disiapkan untuk menyimpan hasil perhitungan nilai mean distances to five boston employment centers (jarak rata-rata ke lima pusat pekerjaan Boston) [DIS]. Variabel low_price_percentage disiapkan untuk menyimpan hasil perhitungan persentase rumah dengan harga rendah (< $20,000). Periksa terlebih dahulu rumah mana yang memiliki harga kurang dari 20 dan kemudian hitung rata-rata dikalikan

100 untuk mendapatkan nilai persentase. {:.2f}% digunakan untuk mencetak nilai persentase dengan 2 angka desimal di belakang koma.

```
# menghitung mean rooms per dwelling
mean_rooms = boston_housing['RM'].mean()
print("Mean rooms per dwelling: ", mean_rooms)

# menghitung median age
median_age = boston_housing['AGE'].median()
print("Median age: ", median_age)

# menghitung mean distances to five Boston employment centers
mean_distances = boston_housing['DIS'].mean()
print("Mean distances to five Boston employment centers: ", mean_distances)

# menghitung persentase rumah dengan harga rendah (< $20,000)
low_price_percentage = (boston_housing['PRICE'] < 20).mean() * 100
print('Percentage of houses with low price (<$20,000): {:.2f}%'.format(low_price_percentage))
```


Berikut merupakan outputnya.

```
Mean rooms per dwelling:  6.284634387351787
Median age:  77.5
Mean distances to five Boston employment centers:  3.795042687747034
Percentage of houses with low price (<$20,000): 41.50%
```