




<b>Nama:</b> April lesa Farsilis  <b>NIM:</b> 065002300005	  <b>Praktikum Statistika</b>	<b>MODUL 3</b>  <b>Nama Dosen:</b> Dedy Sugiarto
<b>Hari/Tanggal:</b> Rabu, 20 Maret 2024		<b>Nama Asisten Labratorium:</b>  1. <b>Kharisma Maulida Saara</b> 064002200024  2. <b>Tarum Widyasti</b>  Pertiwi 064002200027

## Pengelolaan Data pada Data Frame

### 1. Teori Singkat

Pengelolaan data pada DataFrame merupakan konsep penting dalam analisis data menggunakan Python, terutama dengan library seperti pandas. Berikut adalah beberapa teori terkait pengelolaan data pada DataFrame:

- **DataFrame:** DataFrame adalah struktur data dua dimensi yang digunakan untuk menyimpan dan mengelola data dalam bentuk tabel yang terorganisir. Setiap kolom dalam DataFrame mewakili sebuah variabel, sedangkan setiap baris mewakili sebuah catatan atau observasi.
- **Kolom dan Baris:** Kolom dalam DataFrame mewakili variabel, sementara baris mewakili catatan atau observasi. Kolom dan baris dapat diakses menggunakan label atau indeks numerik.
- **Pengindeksan:** Pengindeksan memungkinkan akses ke bagian tertentu dari DataFrame. Anda dapat mengakses kolom menggunakan nama kolom, sedangkan baris dapat diakses menggunakan indeks numerik atau label baris.
- **Manipulasi Data:** Anda dapat melakukan berbagai operasi manipulasi data pada DataFrame, seperti menambah atau menghapus kolom dan baris, mengubah nilai, menggabungkan DataFrame, mengurutkan data, serta melakukan operasi agregasi seperti penghitungan rata-rata atau jumlah.
- **Pembersihan Data:** Pembersihan data adalah proses untuk menghapus atau



- memperbaiki data yang tidak valid, hilang, atau tidak lengkap. Ini melibatkan deteksi dan penanganan nilai yang hilang, duplikat, atau outlier.
- **Pengindeksan dan Pemotongan:** Pengindeksan dan pemotongan memungkinkan Anda untuk memilih subset dari DataFrame berdasarkan kriteria tertentu. Anda dapat menggunakan metode seperti `.loc[]`, `.iloc[]`, dan metode pemotongan untuk melakukan ini.
  - **Pengurutan Data:** Pengurutan data memungkinkan Anda untuk mengurutkan DataFrame berdasarkan nilai pada satu atau beberapa kolom. Hal ini mempermudah analisis dan pemrosesan data yang memerlukan data yang diurutkan.
  - **Operasi Statistik:** Anda dapat melakukan berbagai operasi statistik pada DataFrame, seperti penghitungan rata-rata, median, standar deviasi, korelasi, dan lainnya. Library seperti pandas menyediakan fungsi bawaan untuk melakukan operasi ini dengan mudah.
  - **Menggabungkan DataFrame:** Anda dapat menggabungkan dua atau lebih DataFrame berdasarkan kunci tertentu menggunakan metode seperti `.merge()` atau `.concat()`. Ini berguna saat Anda perlu menggabungkan data dari berbagai sumber.
  - **Ekspor dan Impor Data:** Anda dapat mengimpor data dari berbagai sumber seperti file CSV, Excel, SQL database, atau web API ke dalam DataFrame. Anda juga dapat mengekspor DataFrame ke berbagai format file untuk berbagi atau menyimpan data.

Pengelolaan data pada DataFrame sangat penting dalam analisis data menggunakan Python, karena memungkinkan Anda untuk menyusun, membersihkan, memanipulasi, dan menganalisis data dengan efisien. Dengan menggunakan library seperti pandas, Anda dapat dengan mudah mengelola data dalam format tabular dan menjalankan berbagai analisis data dengan Python.

## 2. Alat dan Bahan

Hardware : Laptop/PC  
Software: Jupyter Notebook

## 3. Elemen Kompetensi

### a. Latihan pertama – Merge Data

1. Jalankan source code berikut. Ganti nama variable (seperti `data_nama`) menjadi variable dengan nama kalian masing - masing. Data yang digunakan adalah data `houseprice.csv`. Melakukan Read CSV dengan cara menginput data `houseprice`, sesuaikan dengan lokasi direktori dimana kalian menyimpan file csvnya.

```
import pandas as pd

data_nama = pd.read_clipboard()

# Menampilkan data
print(data_nama)
```

Output:

**#JANGANLUPAMENGGANTIOUTPUTMENJADIHASILSENDIRI**

```
import pandas as pd

data_april = pd.read_clipboard()

# Menampilkan data
print(data_april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
0	114300	1790	2	2	2	No	East
1	114200	2030	4	2	3	No	East
2	114800	1740	3	2	1	No	East
3	94700	1980	3	2	3	No	East
4	119800	2130	3	3	3	No	East
...	...	...	...	...	...	...	...
123	119700	1900	3	3	3	Yes	East
124	147900	2160	4	3	3	Yes	East
125	113500	2070	2	2	2	No	North
126	149900	2020	3	3	1	No	West
127	124600	2250	3	3	4	No	North

[128 rows x 7 columns]

2. Lakukan subset data dengan cara sub set data frame khusus yang memiliki nilai variabel Bedrooms =2

```
nama = data_nama[data_nama['Bedrooms'] == 2]

# Menampilkan nama
print(nama)
```

Output:

```
april = data_april[data_april['Bedrooms'] == 2]
```

```
# Menampilkan nama
```

```
print(april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
0	114300	1790	2	2	2	No	East
11	123000	1870	2	2	2	Yes	East
18	111400	1700	2	2	1	Yes	East
28	69100	1600	2	2	3	No	North
31	112300	1930	2	2	2	Yes	North
34	117800	2000	2	2	3	No	North
36	117500	1880	2	2	2	No	North
40	106600	1560	2	2	1	No	East
42	105600	1990	2	2	3	No	East
46	129800	1990	2	3	2	No	North
48	115900	1980	2	2	2	No	East
51	91100	1860	2	2	3	No	North
52	117400	2150	2	3	4	No	North
55	125700	1720	2	2	2	Yes	East
61	100900	1610	2	2	2	No	North
63	120500	1910	2	3	2	No	East
65	111100	1450	2	2	1	Yes	North
75	106900	1900	2	2	2	No	North
82	164800	2050	2	2	1	Yes	West
84	90500	1520	2	2	3	No	North
89	97800	2010	2	2	4	No	North
95	152500	1970	2	2	1	Yes	West
97	126800	2000	2	2	1	Yes	East
110	111600	1710	2	2	1	No	North
111	114900	1740	2	2	2	No	North
112	123600	1940	2	2	2	Yes	East
119	109700	1920	2	2	4	No	North

3. Selanjutnya ialah mengubah nama variabel. Berikut adalah tahapan untuk mengubah nilai dalam kolom Bathrooms dalam kondisi jika Bathrooms >2 , maka akan diganti dengan nilai large, jika tidak maka small.

```
nama['Bathrooms'] = pd.to_numeric(nama['Bathrooms'])

import numpy as np

nama['Bathrooms'] = nama['Bathrooms'].apply(lambda x: 'large' if x > 2 else
'small')

# Menampilkan DataFrame setelah modifikasi
print(nama)
```

Output:

```
import numpy as np

april['newvariable'] = np.where(april['Offers'] > 2, 'large', 'small')

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	newvariable
0	114300	1790	2	small	2	No	East	small
11	123000	1870	2	small	2	Yes	East	small
18	111400	1700	2	small	1	Yes	East	small
28	69100	1600	2	small	3	No	North	large
31	112300	1930	2	small	2	Yes	North	small
34	117800	2000	2	small	3	No	North	large
36	117500	1880	2	small	2	No	North	small
40	106600	1560	2	small	1	No	East	small
42	105600	1990	2	small	3	No	East	large
46	129800	1990	2	large	2	No	North	small
48	115900	1980	2	small	2	No	East	small
51	91100	1860	2	small	3	No	North	large
52	117400	2150	2	large	4	No	North	large
55	125700	1720	2	small	2	Yes	East	small
61	100900	1610	2	small	2	No	North	small
63	120500	1910	2	large	2	No	East	small
65	111100	1450	2	small	1	Yes	North	small
75	106900	1900	2	small	2	No	North	small
82	164800	2050	2	small	1	Yes	West	small
84	90500	1520	2	small	3	No	North	large
89	97800	2010	2	small	4	No	North	large
95	152500	1970	2	small	1	Yes	West	small
97	126800	2000	2	small	1	Yes	East	small
110	111600	1710	2	small	1	No	North	small
111	114900	1740	2	small	2	No	North	small
112	123600	1940	2	small	2	Yes	East	small
119	109700	1920	2	small	4	No	North	large
120	110400	1930	2	large	3	No	North	large
122	144800	2060	2	small	1	Yes	East	small
125	113500	2070	2	small	2	No	North	small





4. Lalu langkah selanjutnya adalah menambah variabel. Tahap 1 □ Membuat variable baru dari dataku1 dengan nilai sesuai kondisi ifelse yang ditentukan. Sesuaikan nama variable baru dengan nama praktikan.

```
import numpy as np

nama['newvariable'] = np.where(nama['Offers'] > 2, 'large', 'small')

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(nama)
```

Output

```
import numpy as np

april['newvariable'] = np.where(april['Offers'] > 2, 'large', 'small')

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	Unnamed: 7 \
0	114300	1790	2	small	2	No	East	NaN
11	123000	1870	2	small	2	Yes	East	NaN
18	111400	1700	2	small	1	Yes	East	NaN
28	69100	1600	2	small	3	No	North	NaN
31	112300	1930	2	small	2	Yes	North	NaN
34	117800	2000	2	small	3	No	North	NaN
36	117500	1880	2	small	2	No	North	NaN
40	106600	1560	2	small	1	No	East	NaN
42	105600	1990	2	small	3	No	East	NaN
46	129800	1990	2	large	2	No	North	NaN
48	115900	1980	2	small	2	No	East	NaN
51	91100	1860	2	small	3	No	North	NaN
52	117400	2150	2	large	4	No	North	NaN
55	125700	1720	2	small	2	Yes	East	NaN
61	100900	1610	2	small	2	No	North	NaN
63	120500	1910	2	large	2	No	East	NaN
65	111100	1450	2	small	1	Yes	North	NaN
75	106900	1900	2	small	2	No	North	NaN
82	164800	2050	2	small	1	Yes	West	NaN
84	90500	1520	2	small	3	No	North	NaN
89	97800	2010	2	small	4	No	North	NaN
95	152500	1970	2	small	1	Yes	West	NaN
97	126800	2000	2	small	1	Yes	East	NaN
110	111600	1710	2	small	1	No	North	NaN
111	114900	1740	2	small	2	No	North	NaN
112	123600	1940	2	small	2	Yes	East	NaN
119	109700	1920	2	small	4	No	North	NaN
120	110400	1930	2	large	3	No	North	NaN
122	144800	2060	2	small	1	Yes	East	NaN
125	113500	2070	2	small	2	No	North	NaN

	newvariable
0	small
11	small
18	small
28	large
31	small
34	large
36	small
40	small
42	large



```
# Menambahkan kolom baru 'newvariable'
nama['newvariable'] = nama['Price'] / nama['SqFt']

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(nama)
```

Output:

```
# Menambahkan kolom baru 'newvariable'
april['newvariable'] = april['Price'] / april['SqFt']

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	Unnamed: 7	\
0	114300	1790	2	small	2	No	East	NaN	
11	123000	1870	2	small	2	Yes	East	NaN	
18	111400	1700	2	small	1	Yes	East	NaN	
28	69100	1600	2	small	3	No	North	NaN	
31	112300	1930	2	small	2	Yes	North	NaN	
34	117800	2000	2	small	3	No	North	NaN	
36	117500	1880	2	small	2	No	North	NaN	
40	106600	1560	2	small	1	No	East	NaN	
42	105600	1990	2	small	3	No	East	NaN	
46	129800	1990	2	large	2	No	North	NaN	
48	115900	1980	2	small	2	No	East	NaN	
51	91100	1860	2	small	3	No	North	NaN	
52	117400	2150	2	large	4	No	North	NaN	
55	125700	1720	2	small	2	Yes	East	NaN	
61	100900	1610	2	small	2	No	North	NaN	
63	120500	1910	2	large	2	No	East	NaN	
65	111100	1450	2	small	1	Yes	North	NaN	
75	106900	1900	2	small	2	No	North	NaN	
82	164800	2050	2	small	1	Yes	West	NaN	
84	90500	1520	2	small	3	No	North	NaN	
89	97800	2010	2	small	4	No	North	NaN	
95	152500	1970	2	small	1	Yes	West	NaN	
97	126800	2000	2	small	1	Yes	East	NaN	
110	111600	1710	2	small	1	No	North	NaN	
111	114900	1740	2	small	2	No	North	NaN	
112	123600	1940	2	small	2	Yes	East	NaN	
119	109700	1920	2	small	4	No	North	NaN	
120	110400	1930	2	large	3	No	North	NaN	
122	144800	2060	2	small	1	Yes	East	NaN	
125	113500	2070	2	small	2	No	North	NaN	

```

newvariable
0      63.854749
11     65.775401
18     65.529412
28     43.187500
31     58.186528

```

5. Delete Variabel. Selain bisa menambah, kita juga bisa menghapus variable. Dalam percobaan ini kita akan menghapus variable yang baru saja kita buat.

```
nama = nama.drop(columns=['newvariable'])  
  
# Menampilkan DataFrame 'nama'  
print(nama)
```

Output:

```
april = april.drop(columns=['newvariable'])  
  
# Menampilkan DataFrame 'nama'  
print(april)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	Unnamed: 7
0	114300	1790	2	small	2	No	East	NaN
11	123000	1870	2	small	2	Yes	East	NaN
18	111400	1700	2	small	1	Yes	East	NaN
28	69100	1600	2	small	3	No	North	NaN
31	112300	1930	2	small	2	Yes	North	NaN
34	117800	2000	2	small	3	No	North	NaN
36	117500	1880	2	small	2	No	North	NaN
40	106600	1560	2	small	1	No	East	NaN
42	105600	1990	2	small	3	No	East	NaN
46	129800	1990	2	large	2	No	North	NaN
48	115900	1980	2	small	2	No	East	NaN
51	91100	1860	2	small	3	No	North	NaN
52	117400	2150	2	large	4	No	North	NaN
55	125700	1720	2	small	2	Yes	East	NaN
61	100900	1610	2	small	2	No	North	NaN
63	120500	1910	2	large	2	No	East	NaN
65	111100	1450	2	small	1	Yes	North	NaN
75	106900	1900	2	small	2	No	North	NaN
82	164800	2050	2	small	1	Yes	West	NaN
84	90500	1520	2	small	3	No	North	NaN
89	97800	2010	2	small	4	No	North	NaN
95	152500	1970	2	small	1	Yes	West	NaN
97	126800	2000	2	small	1	Yes	East	NaN
110	111600	1710	2	small	1	No	North	NaN
111	114900	1740	2	small	2	No	North	NaN
112	123600	1940	2	small	2	Yes	East	NaN
119	109700	1920	2	small	4	No	North	NaN
120	110400	1930	2	large	3	No	North	NaN
122	144800	2060	2	small	1	Yes	East	NaN
125	113500	2070	2	small	2	No	North	NaN

6. Merge Data Frame. Tahap 1 □ Merge kolom Artinya memisahkan data sesuai kolom yang diinginkan lalu kemudian digabungkan.

Output:

```
kolom1dan2 = data_nama.iloc[:, 0:2]
```

```
# Menampilkan DataFrame kolom1dan2  
print(kolom1dan2)
```

```
kolom1dan2 = data_april.iloc[:, 0:2]  
  
# Menampilkan DataFrame kolom1dan2  
print(kolom1dan2)
```

	Price	SqFt
0	114300	1790
1	114200	2030
2	114800	1740
3	94700	1980
4	119800	2130
..	...	...
123	119700	1900
124	147900	2160
125	113500	2070
126	149900	2020
127	124600	2250

```
[128 rows x 2 columns]
```

Tahap 2 □ Merge kolom Artinya memisahkan data sesuai kolom yang diinginkan lalu kemudian digabungkan.

```
# Memilih kolom 1 dan 2 dari DataFrame data_nama  
kolom3dan4 = data_nama.iloc[:, 2:4]  
  
# Menampilkan DataFrame kolom3dan4  
print(kolom3dan4)
```

Output:

```
# Memilih kolom 1 dan 2 dari DataFrame data_nama  
kolom3dan4 = data_april.iloc[:, 2:4]  
  
# Menampilkan DataFrame kolom3dan4  
print(kolom3dan4)
```

	Bedrooms	Bathrooms
0	2	2
1	4	2
2	3	2
3	3	2
4	3	3
..	...	...
123	3	3
124	4	3
125	2	2
126	3	3
127	3	3

[128 rows x 2 columns]

Tahap 3 □ Merge kolom Artinya memisahkan data sesuai kolom yang diinginkan lalu kemudian digabungkan.

```
# Menggabungkan dua DataFrame
kolom1sd4 = pd.concat([kolom1dan2, kolom3dan4], axis=1)

# Menampilkan DataFrame kolom1sd4
print(kolom1sd4)
```

Output:

```
#Menggabungkan dua DataFrame
kolom1sd4 = pd.concat([kolom1dan2, kolom3dan4], axis=1)

# Menampilkan DataFrame kolom1sd4
print(kolom1sd4)
```

	Price	SqFt	Bedrooms	Bathrooms
0	114300	1790	2	2
1	114200	2030	4	2
2	114800	1740	3	2
3	94700	1980	3	2
4	119800	2130	3	3
...	...	...	...	...
123	119700	1900	3	3
124	147900	2160	4	3
125	113500	2070	2	2
126	149900	2020	3	3
127	124600	2250	3	3

[128 rows x 4 columns]

7. Merge Baris artinya memisahkan data sesuai baris yang diinginkan dengan menggunakan range baris. Lalu kemudian digabungkan.

```
# Menggabungkan baris dari dua DataFrame
baris1sd3 = data_nama.iloc[0:3, :]
baris4sd6 = data_nama.iloc[3:6, :]
baris1sd6 = baris1sd3.append(baris4sd6)

# Menampilkan DataFrame baris1sd6
print(baris1sd6)
```



## Output

```
import pandas as pd

# Menggabungkan baris dari dua DataFrame
baris1sd3 = data_april.iloc[0:3, :]
baris4sd6 = data_april.iloc[3:6, :]
baris1sd6 = pd.concat([baris1sd3, baris4sd6])

# Menampilkan DataFrame baris1sd6
print(baris1sd6)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	Unnamed: 7
0	114300	1790	2	2	2	No	East	NaN
1	114200	2030	4	2	3	No	East	NaN
2	114800	1740	3	2	1	No	East	NaN
3	94700	1980	3	2	3	No	East	NaN
4	119800	2130	3	3	3	No	East	NaN
5	114600	1780	3	2	2	No	North	NaN

8. Sort data frame. Apa yang terjadi dengan data setelah di sort? Jawaban: ?

```
data_nama_sort = data_nama.sort_values(by='Price')  
  
print(data_nama_sort)
```

Output:

```
data_april_sort = data_april.sort_values(by='Price')  
print(data_april_sort)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood	Unnamed: 7
28	69100	1600	2	2	3	No	North	NaN
54	81300	1650	3	2	3	No	North	NaN
104	82300	1910	3	2	4	No	East	NaN
17	83600	1990	3	3	4	No	North	NaN
47	90300	2050	3	2	6	No	North	NaN
..	...	...	...	...	...	...	...	...
81	184300	2140	4	3	2	Yes	West	NaN
29	188000	2040	4	3	1	Yes	West	NaN
85	188300	2250	4	3	2	Yes	West	NaN
116	199500	2290	5	4	1	Yes	West	NaN
103	211200	2440	4	3	3	Yes	West	NaN

[128 rows x 8 columns]

b. Latihan Kedua – Tugas

Gunakan dataset pada tugas 1 yang telah ditambah lagi datanya dengan 10 mhs TIF/SI. Sehingga total baris data pada file tersebut berjumlah 30. Ulangi kembali perintah-perintah di atas dan sesuaikan dengan data anda. Dan lampirkan Screen Capture untuk tiap poin yang ada.

1. Read CSV

```
import pandas as pd

data_april = pd.read_clipboard()
print(data_april)
```

Output:

```
import pandas as pd
data_april = pd.read_clipboard()
print(data_april)
```

No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan	\
0	1	Achmad Muqafi	L	2023	165	
1	2	Aditya Fajrian Aryadeva	L	2023	165	
2	3	Almakius Felix Bariq Hekopung	L	2023	165	
3	4	Andrew Hendrian Nathanael	L	2023	165	
4	5	April lesa farsilis	P	2023	165	
5	6	Ariel Febrio Hadi	L	2023	160	
6	7	Arik Dias Putra	L	2023	160	
7	8	Aurel regina	P	2023	160	
8	9	BAHRUDIN IZMI AULIA	L	2023	160	
9	10	Bambang Aji Wicaksono	L	2023	160	
10	11	Bidzi Mawfa Ferdiansyah	L	2023	160	
11	12	Calista Azzahra	P	2023	160	
12	13	Dafasyah adinata	L	2023	160	
13	14	Daud Marthinus Paul Jitmau	L	2023	163	
14	15	Dave Ryano F.M	L	2023	163	
15	16	Dewanto Maulana Sukarno Putra	L	2023	163	
16	17	Dyah Kartika Putri (Putri)	P	2023	163	
17	18	Ersa Ashari	L	2023	163	
18	19	Fadhil Ibrahim Wira Dharma	L	2023	163	
19	20	Fairuz Maulidya	L	2023	163	
20	21	Naswa Aulia	P	2023	159	
21	22	Nicholas Saputra	L	2023	159	
22	23	Nicodemus Arjuna Prasetyo	L	2023	159	
23	24	Noval Susanto	L	2023	159	
24	25	Novan Andreas Imanuel pasaribu	L	2023	167	
25	26	Nur Imam	L	2023	167	
26	27	NUZURUL HAQI FADILLAH	L	2023	167	
27	28	Partahi daniel	L	2023	167	
28	29	rachel azzahra putri lukito	P	2023	170	
29	30	rafael gala herlambang	L	2023	170	

waktu	Perjalanan	wilayah	Tinggal
0	35	TANGGERANG	
1	35	TANGGERANG	
2	35	TANGGERANG	
3	35	TANGGERANG	
4	35	TANGGERANG	
5	35	TANGGERANG	



2. Subset Data □ Gunakan Kolom Tinggi Badan untuk bagian ini

```
april = data_april[data_april['Tinggi Badan'] == 160 ]  
  
# Menampilkan nama  
print(april)
```

Output:

```
april = data_april[data_april['Tinggi Badan'] == 160 ]  
  
# Menampilkan nama  
print(april)
```

	No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
5	6	Ariel Febrio Hadi	L	2023		160
6	7	Arik Dias Putra	L	2023		160
7	8	Aurel regina	P	2023		160
8	9	BAHRUDIN IZMI AULIA	L	2023		160
9	10	Bambang Aji Wicaksono	L	2023		160
10	11	Bidzi Mawfa Ferdiansyah	L	2023		160
11	12	Calista Azzahra	P	2023		160
12	13	Dafasyah adinata	L	2023		160

	Waktu Perjalanan	Wilayah	Tinggal
5	35	TANGGERANG	
6	35	BEKASI	
7	35	BEKASI	
8	35	BEKASI	
9	60	BEKASI	
10	60	JAKARTA	
11	60	JAKARTA	
12	60	JAKARTA	

3. Mengubah nilai suatu variable □ Ubah isi kolom tinggi badan untuk tinggi > 160 menjadi “Tinggi” dan jika bukan berarti “Pendek”

```
april['Tinggi Badan'] = pd.to_numeric(april['Tinggi Badan'])

import numpy as np

april['Tinggi Badan'] = april['Tinggi Badan'].apply(lambda x: 'tinggi' if x > 160 else 'pendek')

# Menampilkan DataFrame setelah modifikasi
print(april)
```

Output:

```
april['Tinggi Badan'] = pd.to_numeric(april['Tinggi Badan'])
import numpy as np
april['Tinggi Badan'] = april['Tinggi Badan'].apply(lambda x: 'tinggi' if x > 160 else 'pendek')
# Menampilkan DataFrame setelah modifikasi
print(april)
```

No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
5	6	Ariel Febrio Hadi	L	2023	pendek
6	7	Arik Dias Putra	L	2023	pendek
7	8	Aurel regina	P	2023	pendek
8	9	BAHRUDIN IZMI AULIA	L	2023	pendek
9	10	Bambang Aji Wicaksono	L	2023	pendek
10	11	Bidzi Mawfa Ferdiansyah	L	2023	pendek
11	12	Calista Azzahra	P	2023	pendek
12	13	Dafasyah adinata	L	2023	pendek

	Waktu Perjalanan	Wilayah	Tinggal
5	35	TANGGERANG	
6	35	BEKASI	
7	35	BEKASI	
8	35	BEKASI	
9	60	BEKASI	
10	60	JAKARTA	
11	60	JAKARTA	
12	60	JAKARTA	

#### 4. Menambah Variabel

- Buat variabel kolom baru dengan nama “Jurusan” dengan isi baris datanya adalah “Infor20”
- Buat variabel kolom baru kedua dengan nama “Fakultas” dengan isi baris datanya adalah “FTI”

```
import numpy as np

april['Jurusan']='Infor20'
april['Fakultas']='FTI'

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(april)
```

Output:

```
import numpy as np

april['Jurusan']='Infor20'
april['Fakultas']='FTI'

# Menampilkan DataFrame 'nama' setelah penambahan kolom baru
print(april)
```

	No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
5	6	Ariel Febrio Hadi	L	2023		pendek
6	7	Arik Dias Putra	L	2023		pendek
7	8	Aurel regina	P	2023		pendek
8	9	BAHRUDIN IZMI AULIA	L	2023		pendek
9	10	Bambang Aji Wicaksono	L	2023		pendek
10	11	Bidzi Mawfa Ferdiansyah	L	2023		pendek
11	12	Calista Azzahra	P	2023		pendek
12	13	Dafasyah adinata	L	2023		pendek

	Waktu Perjalanan	Wilayah	Tinggal	Jurusan	Fakultas
5	35	TANGGERANG	Infor20	FTI	
6	35	BEKASI	Infor20	FTI	
7	35	BEKASI	Infor20	FTI	
8	35	BEKASI	Infor20	FTI	
9	60	BEKASI	Infor20	FTI	
10	60	JAKARTA	Infor20	FTI	
11	60	JAKARTA	Infor20	FTI	
12	60	JAKARTA	Infor20	FTI	

5. Delete Variabel □ Hapus kolom Fakultas

```
april = april.drop(columns=['Fakultas'])

# Menampilkan DataFrame 'nama'
print(april)
```

```
april = april.drop(columns=['Fakultas'])  
  
# Menampilkan DataFrame 'nama'  
print(april)
```

	No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
5	6	Ariel Febrio Hadi	L	2023		pendek
6	7	Arik Dias Putra	L	2023		pendek
7	8	Aurel regina	P	2023		pendek
8	9	BAHRUDIN IZMI AULIA	L	2023		pendek
9	10	Bambang Aji Wicaksono	L	2023		pendek
10	11	Bidzi Mawfa Ferdiansyah	L	2023		pendek
11	12	Calista Azzahra	P	2023		pendek
12	13	Dafasyah adinata	L	2023		pendek

	Waktu Perjalanan	Wilayah	Tinggal	Jurusan
5	35	TANGGERANG		Infor20
6	35	BEKASI		Infor20
7	35	BEKASI		Infor20
8	35	BEKASI		Infor20
9	60	BEKASI		Infor20
10	60	JAKARTA		Infor20
11	60	JAKARTA		Infor20
12	60	JAKARTA		Infor20

Output:

## 6. Merge Data Frame

- Gabung kolom Nama dan Gender
- Gabung kolom Angkatan dan Tinggi Badan
- Gabungkan 2 variabel kolom gabungan diatas

```
kolom1dan2 = data_april.iloc[:, 0:2]

# Menampilkan DataFrame kolom1dan2
print(kolom1dan2)

# Memilih kolom 1 dan 2 dari DataFrame data_nama
kolom3dan4 = data_april.iloc[:, 2:4]
# Menampilkan DataFrame kolom3dan4
print(kolom3dan4)

#Menggabungkan dua DataFrame
kolom1sd4 = pd.concat([kolom1dan2, kolom3dan4], axis=1)
# Menampilkan DataFrame kolom1sd4
print(kolom1sd4)
```

Output:

```
kolom1dan2 = data_april.iloc[:, 0:2]

# Menampilkan DataFrame kolom1dan2
print(kolom1dan2)
```

No.	Nama Lengkap
0	1 Achmad Muqafi
1	2 Aditya Fajrian Aryadeva
2	3 Almakius Felix Bariq Hekopung
3	4 Andrew Hendrian Nathanael
4	5 April lesa farsilis
5	6 Ariel Febrio Hadi
6	7 Arik Dias Putra
7	8 Aurel regina
8	9 BAHRUDIN IZMI AULIA
9	10 Bambang Aji Wicaksono
10	11 Bidzi Mawfa Ferdiansyah
11	12 Calista Azzahra
12	13 Dafasyah adinata
13	14 Daud Marthinus Paul Jitmau
14	15 Dave Ryano F.M
15	16 Dewanto Maulana Sukarno Putra
16	17 Dyah Kartika Putri (Putri)
17	18 Ersya Ashari
18	19 Fadhil Ibrahim Wira Dharma
19	20 Fairuz Maulidya
20	21 Naswa Aulia
21	22 Nicholas Saputra
22	23 Nicodemus Arjuna Prasetyo
23	24 Noval Susanto
24	25 Novan Andreas Imanuel pasaribu
25	26 Nur Imam
26	27 NUZURUL HAQI FADILLAH
27	28 Partahi daniel
28	29 rachel azzahra putri lukito
29	30 rafaël gala herlambang

```
#Menggabungkan dua DataFrame
kolom1sd4 = pd.concat([kolom1dan2, kolom3dan4], axis=1)

# Menampilkan DataFrame kolom1sd4
print(kolom1sd4)
```

No.	Nama Lengkap	Gender	Angkatan
0	1 Achmad Muqafi	L	2023
1	2 Aditya Fajrian Aryadeva	L	2023
2	3 Almakius Felix Bariq Hekopung	L	2023
3	4 Andrew Hendrian Nathanael	L	2023
4	5 April lesa farsiliis	P	2023
5	6 Ariel Febrio Hadi	L	2023
6	7 Arik Dias Putra	L	2023
7	8 Aurel regina	P	2023
8	9 BAHRUDDIN IZMI AULIA	L	2023
9	10 Bambang Aji Wicaksono	L	2023
10	11 Bidzi Mawfa Ferdiansyah	L	2023
11	12 Calista Azzahra	P	2023
12	13 Dafasyah adinata	L	2023
13	14 Daud Marthinus Paul Jitmau	L	2023
14	15 Dave Ryano F.M	L	2023
15	16 Dewanto Maulana Sukarno Putra	L	2023
16	17 Dyah Kartika Putri (Putri)	P	2023
17	18 Ersas Ashari	L	2023
18	19 Fadhil Ibrahim Wira Dharma	L	2023
19	20 Fairuz Maulidya	L	2023
20	21 Naswa Aulia	P	2023
21	22 Nicholas Saputra	L	2023
22	23 Nicodemus Arjuna Prasetyo	L	2023
23	24 Noval Susanto	L	2023
24	25 Novan Andreas Imanuel pasaribu	L	2023
25	26 Nur Imam	L	2023
26	27 MUZURUL HAQI FADILLAH	L	2023
27	28 Partahi daniel	L	2023
28	29 Rachel azzahra putri lukito	P	2023
29	30 rafael gala herlambang	L	2023

```
# Memilih kolom 1 dan dari DataFrame data_nama  
kolom3dan4 = data_april.iloc[:, 2:4]  
  
# Menampilkan DataFrame kolom3dan4  
print(kolom3dan4)
```

	Gender	Angkatan
0	L	2023
1	L	2023
2	L	2023
3	L	2023
4	P	2023
5	L	2023
6	L	2023
7	P	2023
8	L	2023
9	L	2023
10	L	2023
11	P	2023
12	L	2023
13	L	2023
14	L	2023
15	L	2023
16	P	2023
17	L	2023
18	L	2023
19	L	2023
20	P	2023
21	L	2023
22	L	2023
23	L	2023
24	L	2023
25	L	2023
26	L	2023
27	L	2023
28	P	2023
29	L	2023

## 7. Merge Baris

- Buat gabungan baris 1-5
- Buat gabungan baris 25-30
- Gabungkan 2 variabel diatas

code:

```
import pandas as pd
```

```
# Menggabungkan baris dari dua DataFrame
```

```
baris1sd5 = data_april.iloc[0:5, :]
```

```
baris25sd30 = data_april.iloc[24:31, :]
```

```
baris1sd5dan25sd30 = pd.concat([baris1sd5, baris25sd30])
```

```
# Menampilkan DataFrame baris1sd6
```

```
print(baris1sd5dan25sd30)
```

Output:

```
import pandas as pd

# Menggabungkan baris dari dua DataFrame
baris1sd5 = data_april.iloc[0:5, :]
baris25sd30 = data_april.iloc[24:31, :]
baris1sd5dan25sd30 = pd.concat([baris1sd5, baris25sd30])

# Menampilkan DataFrame baris1sd6
print(baris1sd5dan25sd30)
```

	No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
0	1	Achmad Muqafi	L	2023		165
1	2	Aditya Fajrian Aryadeva	L	2023		165
2	3	Almakius Felix Bariq Hekopung	L	2023		165
3	4	Andrew Hendrian Nathanael	L	2023		165
4	5	April lesa farsilis	P	2023		165
24	25	Novan Andreas Imanuel pasaribu	L	2023		167
25	26	Nur Imam	L	2023		167
26	27	NUZURUL HAQI FADILLAH	L	2023		167
27	28	Partahi daniel	L	2023		167
28	29	rachel azzahra putri lukito	P	2023		170
29	30	rafael gala herlambang	L	2023		170

	Waktu Perjalanan	Wilayah Tinggal
0	35	TANGGERANG
1	35	TANGGERANG
2	35	TANGGERANG
3	35	TANGGERANG
4	35	TANGGERANG
24	40	CIBUBUR
25	40	CIBUBUR
26	30	CIBUBUR
27	30	PALMERAH
28	30	PALMERAH

8. Sort Data Frame □ Lakukan sort berdasarkan waktu perjalanan

```
data_april_sort = data_april.sort_values(by='Waktu Perjalanan')
print(data_april_sort)
```

Output:



```
data_april_sort = data_april.sort_values(by='Waktu Perjalanan')
print(data_april_sort)
```

No.	Nama Lengkap	Gender	Angkatan	Tinggi	Badan \
14	Dave Ryano F.M	L	2023		163
17	Ersa Ashari	L	2023		163
16	Dyah Kartika Putri (Putri)	P	2023		163
15	Dewanto Maulana Sukarno Putra	L	2023		163
13	Daud Marthinus Paul Jitmau	L	2023		163
19	Fairuz Maulidya	L	2023		163
18	Fadhil Ibrahim Wira Dharma	L	2023		163
27	Partahi daniel	L	2023		167
26	NUZURUL HAQI FADILLAH	L	2023		167
28	rachel azzahra putri lukito	P	2023		170
29	rafael gala herlambang	L	2023		170
0	Achmad Muqafi	L	2023		165
1	Aditya Fajrian Aryadeva	L	2023		165
2	Almakius Felix Bariq Hekopung	L	2023		165
8	BAHRUDIN IZMI AULIA	L	2023		160
7	Aurel regina	P	2023		160
6	Arik Dias Putra	L	2023		160
5	Ariel Febrio Hadi	L	2023		160
4	April lesa farsilis	P	2023		165
3	Andrew Hendrian Nathanael	L	2023		165
25	Nur Imam	L	2023		167
24	Novan Andreas Imanuel pasaribu	L	2023		167
23	Noval Susanto	L	2023		159
20	Naswa Aulia	P	2023		159
21	Nicholas Saputra	L	2023		159
22	Nicodemus Arjuna Prasetyo	L	2023		159
10	Bidzi Mawfa Ferdiansyah	L	2023		160
9	Bambang Aji Wicaksono	L	2023		160
12	Dafasyah adinata	L	2023		160
11	Calista Azzahra	P	2023		160

No.	Waktu Perjalanan	Wilayah Tinggal
14	15	JAKARTA
17	15	JAKARTA
16	15	JAKARTA
15	15	JAKARTA
13	15	JAKARTA
19	20	JAKARTA
18	20	JAKARTA
27	30	PALMERAH
26	30	CIBUBUR
28	30	PALMERAH
29	30	PALMERAH
0	35	TANGGERANG
1	35	TANGGERANG

#### 4. File Praktikum

Github Repository:

<https://github.com/lesa04/prob3.git>

## 5. Soal Latihan

Soal:

1. Apa saja kegunaan pengelolaan data pada data frame?
2. Dalam kasus apakah data perlu dihapus?

Jawaban:

1. Pembersihan Data: Data frame memungkinkan pengguna untuk membersihkan data dari nilai yang hilang atau tidak valid, serta mengatasi duplikasi data.

Transformasi Data: Pengguna dapat melakukan transformasi data seperti mengubah format kolom, membuat kolom baru berdasarkan perhitungan dari kolom yang ada, atau menggabungkan data dari beberapa sumber.

Pemfilteran Data: Data frame memungkinkan pengguna untuk memfilter baris berdasarkan kriteria tertentu.

Pengurutan Data: Data frame memungkinkan pengguna untuk mengurutkan data berdasarkan nilai tertentu dalam kolom.

2. Outlier: Data tidak representatif atau merupakan pencilan.

Duplikat: Terdapat data duplikat yang tidak diperlukan.

Tidak Relevan: Data tidak memberikan kontribusi signifikan terhadap analisis.

Tidak Lengkap: Data memiliki banyak nilai yang hilang dan tidak dapat dipulihkan secara akurat.

## 6. Kesimpulan

- a. Dalam pengerjaan praktikum Statistika, ...mengelola data dengan efisien dalam analisis statistik menggunakan data frame melibatkan sorting, penggabungan data, memisahkan data tidak valid, dan transformasi data seperti mengubah format kolom, membuat kolom baru
- b. Kita juga dapat mengetahui... cara memfilter data , kumpulan data dapat dihapus untuk mencegahnya mempengaruhi hasil analisis , mendapatkan data secara akurat dapat meningkatkan pemahaman dan keterampilan analisis .

## 7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	

## 8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	10 Menit	Menarik
2.	Latihan Kedua	20 Menit	Menarik

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang