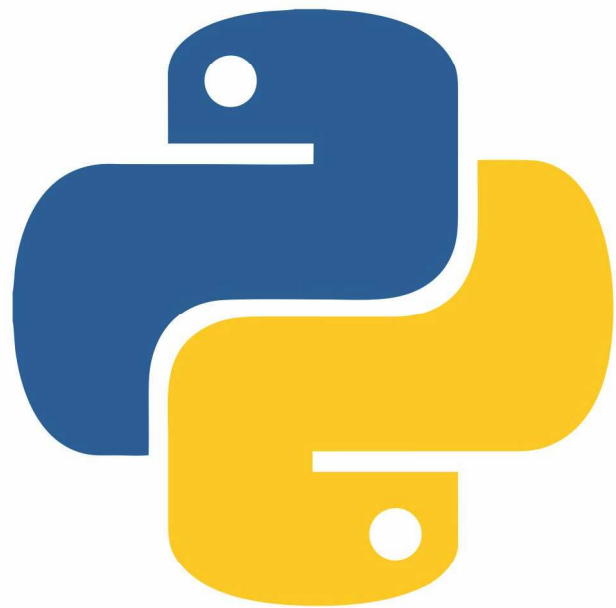


# PYTHON

 DATA SCIENCE

by Kevin Perdana



# PYTHON

## Pandas DataFrame

### 2.1 Pendahuluan

Pandas berasal dari kata Python Data Analysis Library, turunan dari kata Panel Data. Mendukung data multi-dimensi yang artinya elemen-elemen pada data diakses dengan menggunakan 2 buah index. Sedangkan data satu dimensi adalah elemen pada data dapat diakses hanya dengan 1 buah index.

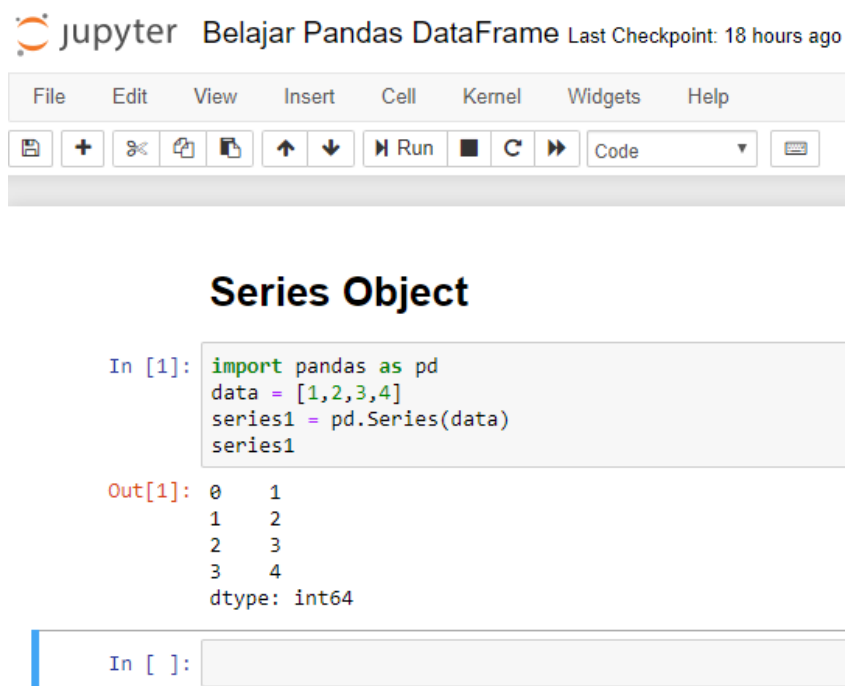
	Age	Location	Name
0	24	New York	John
1	13	Paris	Anna
2	53	Berlin	Peter
3	33	London	Linda

Gambar 4 Data Multi-Dimensi

#### 2.1.1 Series

Series merupakan struktur data dasar dalam Pandas. Series adalah data satu dimensi yang dapat berisi tipe data seperti integer, string, dll. Dan mendukung tipe data sama atau campuran. Contoh series object.

#### 1. Membuat Series



The screenshot shows a Jupyter Notebook titled "Belajar Pandas DataFrame" with a last checkpoint of 18 hours ago. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, navigation, and execution. The main content area displays the following code and output:

```
In [1]: import pandas as pd
data = [1,2,3,4]
series1 = pd.Series(data)
series1
```

```
Out[1]: 0    1
1    2
2    3
3    4
dtype: int64
```

Below the output, there is an input field for the next code cell, labeled "In [ ]:".

Gambar 5 Series Object

# PYTHON

## 2. Cek Tipe Struktur Data Series atau Bukan

```
In [2]: type(series1)
Out[2]: pandas.core.series.Series

In [ ]:
```

Gambar 6 Cek Tipe Struktur Data Series atau Bukan

## 3. Ubah Nama Index

### Ubah Nama Index

```
In [3]: series1 = pd.Series(data, index = ['a', 'b', 'c', 'd'])
series1
Out[3]: a    1
       b    2
       c    3
       d    4
       dtype: int64
```

Gambar 7 Ubah Nama Index

### 2.1.2 DataFrame

DataFrame merupakan array dua dimensi dengan baris dan kolom. Struktur data ini merupakan cara paling standar untuk menyimpan data. Secara sederhana, DataFrame merupakan tabel/data tabular. Setiap kolom pada DataFrame merupakan objek dari Series, dan baris terdiri dari elemen yang ada pada Series. Contoh DataFrame.

#### 1. DataFrame Menggunakan List

### DataFrame

```
In [4]: import pandas as pd
data = [1,2,3,4]
df = pd.DataFrame(data)
df
Out[4]:
```

	0
0	1
1	2
2	3
3	4

Gambar 8 DataFrame Menggunakan List

# PYTHON

## 2. DataFrame Menggunakan Dictionary

```
In [7]: dictionary = {'buah ': ['Apel', 'Jeruk', 'Lemon'], 'jumlah': [10, 5, 12]}  
df = pd.DataFrame(dictionary)  
df
```

```
Out[7]:
```

	buah	jumlah
0	Apel	10
1	Jeruk	5
2	Lemon	12

```
In [ ]:
```

**Gambar 9** DataFrame Menggunakan Dictionary

## 3. DataFrame Menggunakan List dengan Tipe Data Campuran

### DataFrame List Tipe Data Campuran

```
In [2]: import pandas as pd  
data = [['Berti', 90, 85, 95, 90.5],  
        ['Qorygore', 80, 85, 90, 86.6],  
        ['Bimo', 70, 75, 80, 78.5]]  
index = [0, 1, 2]  
kolom = ['Nama', 'Tugas', 'UTS', 'UAS', 'Rata-Rata']  
df = pd.DataFrame(data, index, kolom)  
df
```

```
Out[2]:
```

	Nama	Tugas	UTS	UAS	Rata-Rata
0	Berti	90	85	95	90.5
1	Qorygore	80	85	90	86.6
2	Bimo	70	75	80	78.5

**Gambar 10** DataFrame Menggunakan List dengan Tipe Data Campuran

# PYTHON

## 4. DataFrame Menggunakan List & Dictionary dengan Tipe Data Campuran

```
In [4]: # atau dengan Dictionary
import pandas as pd
nama = ['Berti', 'Qorygore', 'Bimo']
tugas = [90,80,70]
uts = [85,85,75]
uas = [95,90,80]
ratarata = [90.5,86.6,78.5]
df2 = pd.DataFrame({'Nama':nama,'Tugas':tugas,'UTS':uts,'UAS':uas,'Rata-Rata':ratarata})
df2
```

Out[4]:

	Nama	Tugas	UTS	UAS	Rata-Rata
0	Berti	90	85	95	90.5
1	Qorygore	80	85	90	86.6
2	Bimo	70	75	80	78.5

**Gambar 11** DataFrame Menggunakan List & Dict dengan Tipe Data Campuran

# PYTHON

## 2.2 Merge, Join, & Concatenate DataFrame

Ketiga fungsi ini pengertiannya adalah operasi penggabungan. Perbedaannya adalah sebagai berikut.

### 2.2.1 Merge

Merge adalah operasi penggabungan antara DataFrame Objects.

#### 1. Siapkan 2 Data

```
In [19]: # DATA PERTAMA
import pandas as pd
nama = ['Berti', 'Ryndes', 'Arin']
tugas = [95, 90, 75]
jurusan = ['IF', 'SI', 'KA']
df3 = pd.DataFrame({'Nama':nama, 'Tugas':tugas, 'Jurusan':jurusan})
df3
```

```
Out[19]:
```

	Nama	Tugas	Jurusan
0	Berti	95	IF
1	Ryndes	90	SI
2	Arin	75	KA

```
In [20]: # DATA KEDUA
nama = ['Berti', 'Ryndes', 'Rylo']
uts = [85, 84, 70]
jurusan = ['IF', 'SI', 'SI']
df4 = pd.DataFrame({'Nama':nama, 'UTS':uts, 'Jurusan':jurusan})
df4
```

```
Out[20]:
```

	Nama	UTS	Jurusan
0	Berti	85	IF
1	Ryndes	84	SI
2	Rylo	70	SI

Gambar 12 Dua DataFrame untuk di-Merge

#### 2. Inner Merge

```
In [21]: # INNER MERGE
df3.merge(df4)
```

```
Out[21]:
```

	Nama	Tugas	Jurusan	UTS
0	Berti	95	IF	85
1	Ryndes	90	SI	84

Gambar 13 Inner Merge

# PYTHON

## 3. Left Merge

```
In [23]: # LEFT MERGE
df3.merge(df4, on='Nama', how='left')
```

```
Out[23]:
```

	Nama	Tugas	Jurusan_x	UTS	Jurusan_y
0	Berti	95	IF	85.0	IF
1	Ryndes	90	SI	84.0	SI
2	Arin	75	KA	NaN	NaN

**Gambar 14** Left Merge

## 4. Right Merge

```
In [28]: # RIGHT MERGE
df3.merge(df4, on='Nama', how='right')
```

```
Out[28]:
```

	Nama	Tugas	Jurusan_x	UTS	Jurusan_y
0	Berti	95.0	IF	85	IF
1	Ryndes	90.0	SI	84	SI
2	Rylo	NaN	NaN	70	SI

**Gambar 15** Right Merge

## 5. Outer Merge

```
In [29]: # OUTER MERGE
df3.merge(df4, on='Nama', how='outer')
```

```
Out[29]:
```

	Nama	Tugas	Jurusan_x	UTS	Jurusan_y
0	Berti	95.0	IF	85.0	IF
1	Ryndes	90.0	SI	84.0	SI
2	Arin	75.0	KA	NaN	NaN
3	Rylo	NaN	NaN	70.0	SI

**Gambar 16** Outer Merge

# PYTHON

## 2.2.2 Join

Join adalah operasi penggabungan dengan menggunakan index.

### 1. Siapkan 2 Data

```
In [30]: # DATA PERTAMA
nama = ['Berti', 'Ryndes', 'Arin']
tugas = [95, 90, 75]
jurusan = ['IF', 'SI', 'KA']
df3 = pd.DataFrame({'Nama': nama, 'Tugas': tugas, 'Jurusan': jurusan}, index=['L1', 'L2', 'L3'])
df3
```

```
Out[30]:
```

	Nama	Tugas	Jurusan
L1	Berti	95	IF
L2	Ryndes	90	SI
L3	Arin	75	KA

```
In [31]: # DATA KEDUA
nama = ['Berti', 'Ryndes', 'Rylo']
uts = [85, 84, 70]
jurusan = ['IF', 'SI', 'SI']
df4 = pd.DataFrame({'Nama B': nama, 'UTS': uts, 'Jurusan B': jurusan}, index=['L2', 'L3', 'L4'])
df4
```

```
Out[31]:
```

	Nama B	UTS	Jurusan B
L2	Berti	85	IF
L3	Ryndes	84	SI
L4	Rylo	70	SI

Gambar 17 Dua DataFrame untuk di-Join

### 2. Inner Join

```
In [33]: # INNER JOIN
df3.join(df4, how='inner')
```

```
Out[33]:
```

	Nama	Tugas	Jurusan	Nama B	UTS	Jurusan B
L2	Ryndes	90	SI	Berti	85	IF
L3	Arin	75	KA	Ryndes	84	SI

Gambar 18 Inner Join

# PYTHON

## 3. Left Join

```
In [34]: # LEFT JOIN  
df3.join(df4, how='left')
```

Out[34]:

	Nama	Tugas	Jurusan	Nama B	UTS	Jurusan B
L1	Berti	95	IF	NaN	NaN	NaN
L2	Ryndes	90	SI	Berti	85.0	IF
L3	Arin	75	KA	Ryndes	84.0	SI

Gambar 19 Left Join

## 4. Right Join

```
In [35]: # RIGHT JOIN  
df3.join(df4, how='right')
```

Out[35]:

	Nama	Tugas	Jurusan	Nama B	UTS	Jurusan B
L2	Ryndes	90.0	SI	Berti	85	IF
L3	Arin	75.0	KA	Ryndes	84	SI
L4	NaN	NaN	NaN	Rylo	70	SI

Gambar 20 Right Join

## 5. Outer Join

```
In [36]: # OUTER JOIN  
df3.join(df4, how='outer')
```

Out[36]:

	Nama	Tugas	Jurusan	Nama B	UTS	Jurusan B
L1	Berti	95.0	IF	NaN	NaN	NaN
L2	Ryndes	90.0	SI	Berti	85.0	IF
L3	Arin	75.0	KA	Ryndes	84.0	SI
L4	NaN	NaN	NaN	Rylo	70.0	SI

Gambar 21 Outer Join

# PYTHON

## 2.2.3 Concatenate

Concatenate adalah operasi penggabungan objek DataFrame secara vertical.

```
In [42]: pd.concat([df3,df4], sort='False')
```

Out[42]:

	Jurusan	Jurusan B	Nama	Nama B	Tugas	UTS
L1	IF	NaN	Berti	NaN	95.0	NaN
L2	SI	NaN	Ryndes	NaN	90.0	NaN
L3	KA	NaN	Arin	NaN	75.0	NaN
L2	NaN	IF	NaN	Berti	NaN	85.0
L3	NaN	SI	NaN	Ryndes	NaN	84.0
L4	NaN	SI	NaN	Rylo	NaN	70.0

Gambar 22 Concatenate

## 2.3 Pandas DataFrame - Import Data CSV

Pada tahap ini akan dibahas mengenai cara import data CSV ke dalam Panda DataFrame dan mengolah datanya.

### 2.3.1 Import Data CSV

```
In [3]: import pandas as pd

# membaca dataset dan store ke dataframe
sample = pd.read_csv("sampledataok.csv")

# print
sample
```

Out[3]:

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	7000000
1	Statement Prod	L	29	Daily Vlog	120000
2	Arief Muhammad	L	28	Daily Vlog	3000000
3	Annisa Aziza	P	25	Food Travel	600000
4	Sarah Viloid	P	23	Gamer	2000000
5	MLI	L	30	Komedi	800000
6	Chandra Liow	L	26	Sketsa	3000000

Gambar 23 Import Data CSV

# PYTHON

## 2.3.2 Macam - Macam Operasi

Fungsi-fungsi yang dapat dilakukan oleh Pandas DataFrame adalah sebagai berikut.

### 1. head()

```
In [4]: # head(), menampilkan 5 records pertama  
sample.head()
```

Out[4]:

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	7000000
1	Statement Prod	L	29	Daily Vlog	120000
2	Arief Muhammad	L	28	Daily Vlog	3000000
3	Annisa Aziza	P	25	Food Travel	600000
4	Sarah Viloid	P	23	Gamer	2000000

```
In [5]: # menampilkan 2 records pertama  
sample.head(2)
```

Out[5]:

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	7000000
1	Statement Prod	L	29	Daily Vlog	120000

Gambar 24 head()

### 2. tail()

```
In [6]: # tail(), menampilkan 5 records terakhir  
sample.tail()
```

Out[6]:

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
2	Arief Muhammad	L	28	Daily Vlog	3000000
3	Annisa Aziza	P	25	Food Travel	600000
4	Sarah Viloid	P	23	Gamer	2000000
5	MLI	L	30	Komedi	800000
6	Chandra Liow	L	26	Sketsa	3000000

```
In [7]: # menampilkan 3 records terakhir  
sample.tail(3)
```

Out[7]:

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
4	Sarah Viloid	P	23	Gamer	2000000
5	MLI	L	30	Komedi	800000
6	Chandra Liow	L	26	Sketsa	3000000

Gambar 25 tail()

# PYTHON

## 3. shape

```
In [8]: # menampilkan jumlah baris dan kolom pada DataFrame
sample.shape

Out[8]: (7, 5)
```

Gambar 26 shape

## 4. Mean, Median, Standar Deviasi

```
In [9]: # mean adalah rata-rata
# 2.785714e+01 artinya 2.785714 * 10^1 = 2.785714 * 10 jadi Mean-nya adalah 27.85714
# 2.360000e+06 artinya 2.360000 * 10^6 = 2.360000 * 1000000 jadi Mean-nya adalah 2360000
sample.mean()

Out[9]: umur          2.785714e+01
subscriber  2.360000e+06
dtype: float64

In [10]: # median adalah nilai tengah dari data yang telah diurut dari terkecil hingga terbesar
# umur : 23 25 26 28 29 30 34
sample.median()

Out[10]: umur          28.0
subscriber  2000000.0
dtype: float64

In [15]: # standar deviasi
sample.std()

Out[15]: umur          3.625308e+00
subscriber  2.346174e+06
dtype: float64
```

Gambar 27 Mean, Median, Standar Deviasi

# PYTHON

## 5. Max, Min, Count

```
In [16]: # max untuk mencari nilai tertinggi  
sample.max()
```

```
Out[16]: nama_youtuber    Statement Prod  
jenis_kelamin          P  
umur                   34  
kategori               Sketsa  
subscriber              7000000  
dtype: object
```

```
In [17]: # min untuk mencari nilai terendah  
sample.min()
```

```
Out[17]: nama_youtuber    Annisa Aziza  
jenis_kelamin          L  
umur                   23  
kategori               Daily Vlog  
subscriber              120000  
dtype: object
```

```
In [18]: # jumlah non null record pada setiap kolom  
sample.count()
```

```
Out[18]: nama_youtuber    7  
jenis_kelamin          7  
umur                   7  
kategori               7  
subscriber              7  
dtype: int64
```

**Gambar 28** Max, Min, Count

## 6. describe()

```
In [19]: # ringkasan statistik data  
sample.describe()
```

```
Out[19]:
```

	umur	subscriber
count	7.000000	7.000000e+00
mean	27.857143	2.360000e+06
std	3.625308	2.346174e+06
min	23.000000	1.200000e+05
25%	25.500000	7.000000e+05
50%	28.000000	2.000000e+06
75%	29.500000	3.000000e+06
max	34.000000	7.000000e+06

**Gambar 29** describe()

# PYTHON

## 7. Rename dan Drop Kolom

```
In [20]: # rename kolom  
sample = sample.rename(columns={'nama_youtuber': 'Youtuber'})  
sample
```

```
Out[20]:
```

	Youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	7000000
1	Statement Prod	L	29	Daily Vlog	120000
2	Arief Muhammad	L	28	Daily Vlog	3000000
3	Annisa Aziza	P	25	Food Travel	600000
4	Sarah Viloid	P	23	Gamer	2000000
5	MLI	L	30	Komedi	800000
6	Chandra Liow	L	26	Sketsa	3000000

```
In [21]: # drop atau menghilangkan kolom  
sample = sample.drop(columns=['jenis_kelamin'])  
sample
```

```
Out[21]:
```

	Youtuber	umur	kategori	subscriber
0	Raditya Dika	34	Komedi	7000000
1	Statement Prod	29	Daily Vlog	120000
2	Arief Muhammad	28	Daily Vlog	3000000
3	Annisa Aziza	25	Food Travel	600000
4	Sarah Viloid	23	Gamer	2000000
5	MLI	30	Komedi	800000
6	Chandra Liow	26	Sketsa	3000000

**Gambar 30** Rename dan Drop Kolom

# PYTHON

## 8. iloc

```
In [9]: # menampilkan record 1 kolom (umur yang berada di indeks ke-2)
sample.iloc[:,2]
```

```
Out[9]: 0    34
1    29
2    28
3    25
4    23
5    30
6    26
Name: umur, dtype: int64
```

```
In [11]: # menampilkan 3 record pertama dari 1 kolom (umur)
sample.iloc[0:3,2]
```

```
Out[11]: 0    34
1    29
2    28
Name: umur, dtype: int64
```

```
In [12]: # tampilkan kembali semua baris dan kolom
sample.iloc[:,:]
```

```
Out[12]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	7000000
1	Statement Prod	L	29	Daily Vlog	120000
2	Arief Muhammad	L	28	Daily Vlog	3000000
3	Annisa Aziza	P	25	Food Travel	600000
4	Sarah Viloid	P	23	Gamer	2000000
5	MLI	L	30	Komedi	800000
6	Chandra Liow	L	26	Sketsa	3000000

```
In [16]: # menampilkan data dari record ke-3 dan kolom ke-2
sample.iloc[3:,2:]
```

```
Out[16]:
```

	umur	kategori	subscriber
3	25	Food Travel	600000
4	23	Gamer	2000000
5	30	Komedi	800000
6	26	Sketsa	3000000

Gambar 31 iloc

# PYTHON

## 9. loc

```
In [19]: # menampilkan record 1 kolom, dengan menulis nama kolomnya("nama_youtuber")
sample.loc[:, "nama_youtuber"]
```

```
Out[19]: 0    Raditya Dika
1    Statement Prod
2    Arief Muhammad
3     Annisa Aziza
4     Sarah Viloid
5             MLI
6     Chandra Liow
Name: nama_youtuber, dtype: object
```

```
In [23]: # menampilkan record dari indeks ke-0 sampai ke-3 dari kolom "nama_youtuber"
sample.loc[0:3, "nama_youtuber"]
```

```
Out[23]: 0    Raditya Dika
1    Statement Prod
2    Arief Muhammad
3     Annisa Aziza
Name: nama_youtuber, dtype: object
```

```
In [24]: # menampilkan record sampai indeks ke-3 dari kolom nama_youtuber sampai jenis_kelamin
sample.loc[:3, "nama_youtuber": "jenis_kelamin"]
```

```
Out[24]:
```

	nama_youtuber	jenis_kelamin
0	Raditya Dika	L
1	Statement Prod	L
2	Arief Muhammad	L
3	Annisa Aziza	P

Gambar 32 loc

## 10. Mengisi Nilai Sama Untuk 1 Kolom

```
In [25]: # mengisi nilai 1 untuk kolom "subscriber"
sample['subscriber']=1
sample
```

```
Out[25]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	1
1	Statement Prod	L	29	Daily Vlog	1
2	Arief Muhammad	L	28	Daily Vlog	1
3	Annisa Aziza	P	25	Food Travel	1
4	Sarah Viloid	P	23	Gamer	1
5	MLI	L	30	Komedi	1
6	Chandra Liow	L	26	Sketsa	1

Gambar 33 Mengisi Nilai Sama Untuk 1 Kolom

# PYTHON

## 11. Sorting

```
In [31]: # sorting berdasarkan ascending dari kolom "kategori"  
sample.sort_values(by='kategori')
```

```
Out[31]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
1	Statement Prod	L	29	Daily Vlog	1
2	Arief Muhammad	L	28	Daily Vlog	1
3	Annisa Aziza	P	25	Food Travel	1
4	Sarah Viloid	P	23	Gamer	1
0	Raditya Dika	L	34	Komedi	1
5	MLI	L	30	Komedi	1
6	Chandra Liow	L	26	Sketsa	1

```
In [32]: # sorting berdasarkan descending dari kolom "kategori"  
sample.sort_values(by='kategori', ascending=False)
```

```
Out[32]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
6	Chandra Liow	L	26	Sketsa	1
0	Raditya Dika	L	34	Komedi	1
5	MLI	L	30	Komedi	1
4	Sarah Viloid	P	23	Gamer	1
3	Annisa Aziza	P	25	Food Travel	1
1	Statement Prod	L	29	Daily Vlog	1
2	Arief Muhammad	L	28	Daily Vlog	1

Gambar 34 Sorting

## 12. Filter

```
In [35]: # filter record yang "umurnya" Lebih dari 28 dan akan menampilkan status True atau False  
sample['umur'] > 28
```

```
Out[35]: 0    True  
1    True  
2   False  
3   False  
4   False  
5    True  
6   False  
Name: umur, dtype: bool
```

```
In [36]: # filter, kemudian tampilkan record-nya  
filter1 = sample['umur'] > 28  
filterbaru = sample[filter1]  
filterbaru
```

```
Out[36]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
0	Raditya Dika	L	34	Komedi	1
1	Statement Prod	L	29	Daily Vlog	1
5	MLI	L	30	Komedi	1

```
In [55]: # filter dengan 2 parameter (berdasarkan yang umurnya 27 dan kategori Daily Vlog)  
filter2 = (sample['umur'] > 27) & (sample['kategori'] == 'Daily Vlog')  
filterbaru2 = sample[filter2]  
filterbaru2
```

```
Out[55]:
```

	nama_youtuber	jenis_kelamin	umur	kategori	subscriber
1	Statement Prod	L	29	Daily Vlog	120000
2	Arief Muhammad	L	28	Daily Vlog	3000000

Gambar 35 Filter