

LAPORAN TUGAS PHYTON 2  
MENGOLAH DATA MENJADI GRAFIK



Disusun Oleh:  
Ferdian Yoga Aditama  
03411940000034

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

2020

## I. Data Asli

Pada tugas kali ini, saya menggunakan data jumlah penduduk Kota Malang dari tahun 2011-2020.

Kecamatan di Kota Malang	Jumlah Penduduk Kota Malang(Jiwa)									
	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
Kedungkandang	177260	179512	181834	183927	186068	188175	190274	192316	194341	196298
Sukun	183690	185352	187074	188545	190053	191513	192951	194321	195659	196917
Klojen	105755	105399	105060	104590	104127	103637	103129	102584	102018	101410
Blimbing	173838	174891	175988	176845	177729	178564	179368	180104	180805	181426
Lowokwaru	187948	189373	190847	192066	193321	194521	195692	196793	197859	198839
KOTA MALANG	828491	834527	840803	845973	851298	856410	861414	866118	870682	874890

Sumber: bps Kota Malang

## II. Pengolahan Data Pada Phyton

- `import pandas as pd`
- `iris = pd.read_csv('https://raw.githubusercontent.com/ferdianyoga034/Tugas-Phyton2/master/data%20penduduk%20malang%20bismillah.csv')`
- `iris.head()`

Out [29]:

	Kecamatan di Kota Malang	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
0	Kedungkandang	177260	179512	181834	183927	186068	188175	190274	192316	194341	196298
1	Sukum	183690	185352	187074	188545	190053	191513	192951	194321	195659	196917
2	Klojen	105755	105399	105060	104590	104127	103637	103129	102584	102018	101410
3	Blimbing	173838	174891	175988	176845	177729	178564	179368	180104	180805	181426
4	Lowokwaru	187948	189373	190847	192066	193321	194521	195692	196793	197859	198839

- `iris.tail()`

Out [30]:

	Kecamatan di Kota Malang	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
1	Sukum	183690	185352	187074	188545	190053	191513	192951	194321	195659	196917
2	Klojen	105755	105399	105060	104590	104127	103637	103129	102584	102018	101410
3	Blimbing	173838	174891	175988	176845	177729	178564	179368	180104	180805	181426
4	Lowokwaru	187948	189373	190847	192066	193321	194521	195692	196793	197859	198839
5	Kota Malang	828491	834527	845973	845973	851298	856410	861414	866118	870682	874890

- iris.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6 entries, 0 to 5
Data columns (total 11 columns):
Kecamatan di Kota Malang    6 non-null object
2011                        6 non-null int64
2012                        6 non-null int64
2013                        6 non-null int64
2014                        6 non-null int64
2015                        6 non-null int64
2016                        6 non-null int64
2017                        6 non-null int64
2018                        6 non-null int64
2019                        6 non-null int64
2020                        6 non-null int64
dtypes: int64(10), object(1)
memory usage: 656.0+ bytes
```

- iris.describe()

	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
count	6.000000	6.000000	6.000000	6.000000	6.000000	6.000000	6.000000	6.000000	6.000000	6.000000
mean	276163.666667	278175.666667	281129.333333	281991.000000	283766.000000	285470.000000	287138.000000	288706.000000	290227.333333	291630.000000
std	272283.108926	274329.353455	278553.530421	278222.105907	280040.679299	281790.905389	283507.603262	285127.329776	286702.943073	288163.047558
min	105755.000000	105399.000000	105060.000000	104590.000000	104127.000000	103637.000000	103129.000000	102584.000000	102018.000000	101410.000000
25%	174693.500000	176046.250000	177449.500000	178615.500000	179813.750000	180966.750000	182094.500000	183157.000000	184189.000000	185144.000000
50%	180475.000000	182432.000000	184454.000000	186236.000000	188060.500000	189844.000000	191612.500000	193318.500000	195000.000000	196607.500000
75%	186883.500000	188367.750000	189903.750000	191185.750000	192504.000000	193769.000000	195006.750000	196175.000000	197309.000000	198358.500000
max	828491.000000	834527.000000	845973.000000	845973.000000	851298.000000	856410.000000	861414.000000	866118.000000	870682.000000	874890.000000

- iris.groupby('2011').mean()

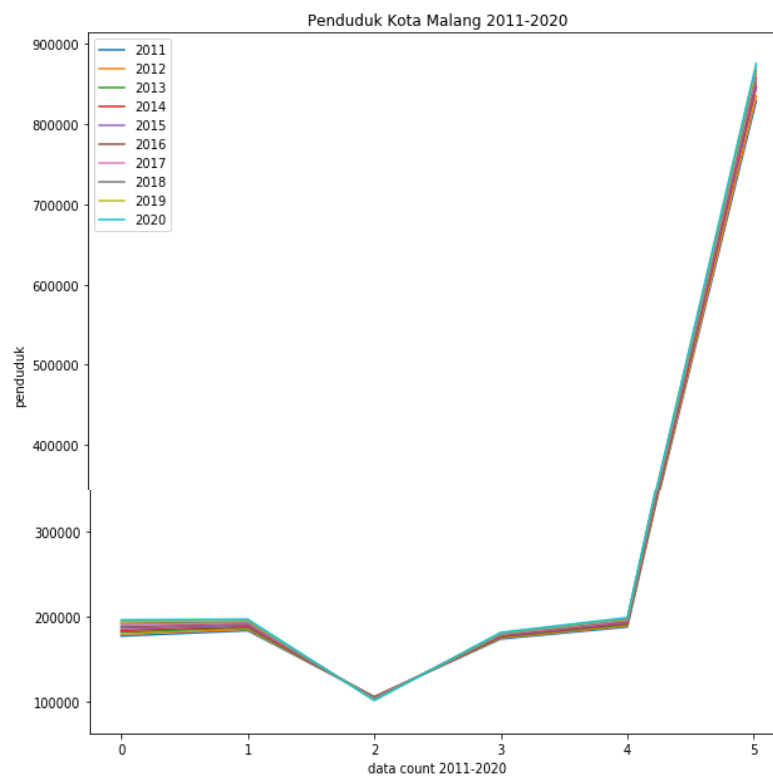
	2012	2013	2014	2015	2016	2017	2018	2019	2020
2011									
105755	105399	105060	104590	104127	103637	103129	102584	102018	101410
173838	174891	175988	176845	177729	178564	179368	180104	180805	181426
177260	179512	181834	183927	186068	188175	190274	192316	194341	196298
183690	185352	187074	188545	190053	191513	192951	194321	195659	196917
187948	189373	190847	192066	193321	194521	195692	196793	197859	198839
828491	834527	845973	845973	851298	856410	861414	866118	870682	874890

- import numpy as np
- import matplotlib.pyplot as plt
- plt.figure(figsize=(10,10))
  - plt.plot(iris['2011'], label='2011')
  - plt.plot(iris['2012'], label='2012')
  - plt.plot(iris['2013'], label='2013')
  - plt.plot(iris['2014'], label='2014')

```

plt.plot(iris['2015'], label='2015')
plt.plot(iris['2016'], label='2016')
plt.plot(iris['2017'], label='2017')
plt.plot(iris['2018'], label='2018')
plt.plot(iris['2019'], label='2019')
plt.plot(iris['2020'], label='2020')
plt.xlabel('data count 2011-2020')
plt.ylabel('penduduk')
plt.title('Penduduk Kota Malang 2011-2020')
plt.legend()
plt.show()

```



### III. Penjelasan Fungsi Pada Script

1. `import pandas as pd` berfungsi untuk memanipulasi data, persiapan data, dan pembersihan data. Penggunaan *as* berarti menggantikan pandas dengan *prefix* pd untuk proses berikutnya.
2. `iris = pd.read_csv('https://raw.githubusercontent.com/ferdianyoga034/Tugas-Phyton2/master/data%20penduduk%20malang%20bismillah.csv')` `iris` berfungsi sebagai nama file yang akan digunakan, `pd.read` berfungsi untuk membaca data dari file yang tertera dalam link tersebut, pembacaan data (tabel) menggunakan format csv.
3. `iris.head()` berfungsi untuk menampilkan data secara *default* untuk 5 data teratas.

4. `iris.tail()` berfungsi untuk menampilkan data secara *default* untuk 5 data terbawah.
5. `iris.info()` berfungsi untuk menampilkan informasi yang berkaitan dengan data yang ada.
6. `iris.describe()` berfungsi untuk mengetahui statistika data untuk data *numeric* seperti *count*, *mean*, *standard deviation*, *maximum*, *minimum*, dan *quartile*.
7. `iris.groupby('2011').mean()` berfungsi untuk mengelompokkan data dari tabel, pada data di atas dikelompokkan setelah dikalikan dengan rata-rata data.
8. `import numpy as np` berfungsi untuk memanggil numpy (membentuk objek N-dimensional array, mirip dengan list pada Python).
9. `import matplotlib.pyplot as plt` penggunaan `as` berfungsi untuk memanggil atau mengaktifkan fungsi, matplotlib berfungsi untuk membuat grafik dari data tabel yang tersedia.
10. `plt.figure(figsize=(10,10))` berfungsi untuk membuat grafik dengan ukuran yang sudah ditentukan, dalam grafik di atas menggunakan ukuran (10,10)
11. `plt.plot(iris['2011'], label='2011')` berfungsi untuk memasukkan data tabel ke dalam grafik yang akan dibuat, misalnya pada poin tersebut berarti memasukkan data pada tahun 2011 untuk diinput ke dalam grafik, kemudian diberi label atau nama (2011).
12. `plt.xlabel('data count 2011-2020')` berfungsi untuk memberi label atau nama pada sumbu-x grafik, pada grafik di atas diberi label (data count 2011-2020).
13. `plt.ylabel('penduduk')` berfungsi untuk memberi label atau nama pada sumbu-y grafik, pada grafik di atas diberi label (penduduk).
14. `plt.title('Penduduk Kota Malang 2011-2020')` berfungsi untuk memberi judul pada grafik agar pembaca mudah memahami, grafik di atas diberi nama (Penduduk Kota Malang 2011-2020).
15. `plt.legend()` berfungsi untuk menampilkan legenda dari grafik yang dibuat, meliputi judul, label, warna, dan keterangan lainnya.
16. `plt.show()` berfungsi untuk menampilkan grafik pada hasil script Python (Window).

#### IV. Penjelasan Grafik

Pada sumbu-x (nomor 0 sampai 4) adalah kecamatan yang ada di Malang, sedangkan nomor 5 adalah jumlah seluruh penduduk di Kota Malang. Dalam grafik juga tertera keterangan warna yang menunjukkan jumlah penduduk pada tahun tertentu. Sumbu-y merupakan jumlah penduduk.

#### V. Alasan Memilih Data Penduduk Kota Malang 2011-2020

Saya memilih data “Jumlah Penduduk Kota Malang Tahun 2011-2020” karena saya berasal dari Kota Malang, selain itu data yang saya peroleh mudah untuk

dipahami dan diolah ke dalam Phyton. Ketika sudah menjadi bentuk grafik, pembaca akan mudah memahami maksud dari grafik tersebut.