

Operasi Python Menggunakan Data Analysis Library (Pandas)

Python memiliki 3 library terbaik dalam pengolahan data science. Dalam pengolahan data tugas kali ini menggunakan 2 library ,yaitu Pandas dan Numpy. Dalam pengolahan data dengan python lebih efisiensi,keterbacaan kode dan kecepatan telah membuat python menjadi bahasa pemrograman yang banyak digunakan.Python menjadi pilihan untuk para data scientist dan machine learning engineer untuk mengembangkan model dan berbagai aplikasi terkait data science.berikut penejelasan mengenai Pandas dan Numpy library :

1. Pandas

Pandas kependekan dari Python Data Analysis Library. Pandas merupakan sebuah *Open source package library* dengan lisensi BSD yang menyediakan banyak perkakas untuk kebutuhan data analisis, maipulasi,dan pembersihan data.Pandas mendukung pembacaan dan penulisan data dengan media berupa *Excel Spreadsheet*,CSV,dan SQL yang kemudian akan dijadikan sebagai objek python dengan *rows* dan *columns* yang disebut *data frame* seperti halnya pada tabel statistik.

Ada tiga jenis struktur data di library ini:

1. Series: single-dimensional, array homogeny
2. DataFrame: two-dimensional dengan kolom yang diketik secara heterogen
3. Panel: three-dimensional, array size-mutable

Sebagai contoh,kita menggunakan library Panda Python yang disingkat (pd) dapat digunakan untuk melakukan beberapa perhitungan statistik deskriptif.

```
In [1]: import pandas as pd
```

```
In [10]: data=pd.read_csv('https://raw.githubusercontent.com/Yosiaginting/tugas-python02/master/data%20penduduk%20medan.csv')
```

Adapun hasil raw table data “jumlah penduduk untuk wilayah Medan dengan rentang tahun 2011-2015” yang didapatkan dari GITHUB adalah sebagai berikut:

1	Wilayah	2011	2012	2013	2014	2015
2	Medan Tuntungan	41974	42155	42437	42920	43325
3	Medan Johor	63589	63908	64336	66027	66805
4	Medan Amplas	58186	58612	59004	61444	62674
5	Medan Denai	70459	70627	71100	73754	73914
6	Medan Area	48791	48873	49200	50099	50095
7	Medan Kota	37389	37449	37700	37671	37670
8	Medan Maimun	20206	20243	20379	20567	20577
9	Medan Polonia	27109	27231	27413	28032	28313
10	Medan Baru	21971	22003	22150	20514	20515
11	Medan Selayang	50879	51189	51532	52884	53717
12	Medan Sunggal	57439	57542	57927	58571	58593
13	Medan Helvetia	74065	74308	74805	75845	76273
14	Medan Petisah	32431	32484	32701	32065	32071
15	Medan Barat	36099	36164	36406	36767	36781
16	Medan Timur	56071	56163	56539	56385	56384
17	Medan Perjuangan	48272	48359	48683	48497	48521
18	Medan Tembung	68303	68424	68888	69392	69419
19	Medan Deli	83876	84449	85014	90193	91828

Berikut beberapa Tools yang digunakan untuk melihat/mengecek data:

- ❖ `head(n)` : Berfungsi untuk melihat data sebanyak n pada kolom awal. Dimana data yang ditampilkan berupa data awal dari 0 s.d 4 (memuat 5 data awal dari 0). Berikut Ouput yang didapatkan pada data ini, sebagai berikut:

In [3]:

▶

data.head()

Out[3]:

	Wilayah	2011	2012	2013	2014	2015
0	Medan Tuntungan	41974	42155	42437	42920	43325
1	Medan Johor	63589	63908	64336	66027	66805
2	Medan Amplas	58186	58612	59004	61444	62674
3	Medan Denai	70459	70627	71100	73754	73914
4	Medan Area	48791	48873	49200	50099	50095

- ❖ `tail(n)` : Berfungsi untuk melihat data sebanyak `n` pada kolom akhir. Dimana data yang ditampilkan berupa data akhir dari 17 s.d 21 (data 5 terakhir). berikut Output yang didapatkan pada data ini, sebagai berikut:

```
In [4]: data.tail()
```

Out[4]:

	Wilayah	2011	2012	2013	2014	2015
17	Medan Deli	83876	84449	85014	90193	91828
18	Medan Labuhan	55068	55309	55679	58910	59447
19	Medan Marelan	71786	72645	73131	79180	82115
20	Medan Belawan	46701	46794	47105	49626	49650
21	Medan	1070664	1074929	1082123	1109343	1118687

- ❖ `info()` : Nomor index beserta tipe datanya. Tipe data yang digunakan. Berikut Output yang didapatkan pada data ini, sebagai berikut:

```
In [5]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22 entries, 0 to 21
Data columns (total 6 columns):
Wilayah    22 non-null object
2011       22 non-null int64
2012       22 non-null int64
2013       22 non-null int64
2014       22 non-null int64
2015       22 non-null int64
dtypes: int64(5), object(1)
memory usage: 1.2+ KB
```

- ❖ `describe()` : Menunjukkan rangkuman statistik seperti rata-rata, median, dll pada kolom.

```
In [7]: data.describe()
```

Out[7]:

	2011	2012	2013	2014	2015
count	2.200000e+01	2.200000e+01	2.200000e+01	2.200000e+01	2.200000e+01
mean	9.733309e+04	9.772091e+04	9.837509e+04	1.008494e+05	1.016988e+05
std	2.180792e+05	2.189527e+05	2.204180e+05	2.260328e+05	2.279591e+05
min	2.020600e+04	2.024300e+04	2.037900e+04	2.051400e+04	2.051500e+04
25%	3.853525e+04	3.862550e+04	3.888425e+04	3.898325e+04	3.908375e+04
50%	5.297350e+04	5.324900e+04	5.360550e+04	5.463450e+04	5.505050e+04
75%	6.712450e+04	6.729500e+04	6.775000e+04	6.855075e+04	6.876550e+04
max	1.070664e+06	1.074929e+06	1.082123e+06	1.109343e+06	1.118687e+06

- ❖ Dilanjutkan dengan menggunakan “Data groupby (‘Jumlah penduduk di wilayah Medan dan sekitarnya dalam rentang tahun 2011 sampai dengan 2015’).mean()” yang artinya adalah mengelompokkan data yang diinginkan menjadi data rata-rata sehingga ditambahkan “mean” pada akhir formula. Output yang didapatkan adalah

```
In [8]: data.groupby('2015').mean()
```

Out[8]:

	2011	2012	2013	2014
2015				
20515	21971	22003	22150	20514
20577	20206	20243	20379	20567
28313	27109	27231	27413	28032
32071	32431	32484	32701	32065
36781	36099	36164	36406	36767
37670	37389	37449	37700	37671
43325	41974	42155	42437	42920
48521	48272	48359	48683	48497
49650	46701	46794	47105	49626
50095	48791	48873	49200	50099
53717	50879	51189	51532	52884

Untuk menampilkan sebuah grafik berdasarkan data yang di input, harus mengimpor library lain yang akan mendukung pembentukan grafik yaitu dengan mengimpor data ke matplotlib. Matplotlib juga merupakan bagian dari paket inti SciPy dan ditawarkan di bawah lisensi BSD. Ini adalah library ilmiah Python populer yang digunakan untuk menghasilkan visualisasi yang sederhana dan kuat. Anda dapat menggunakan kerangka kerja Python untuk ilmu data untuk menghasilkan grafik, chart, histogram, dan bentuk dan gambar lain yang kreatif — tanpa perlu khawatir menulis banyak baris kode. Sebagai contoh, mari kita lihat bagaimana perpustakaan Matplotlib dapat digunakan untuk membuat bar chart sederhana.

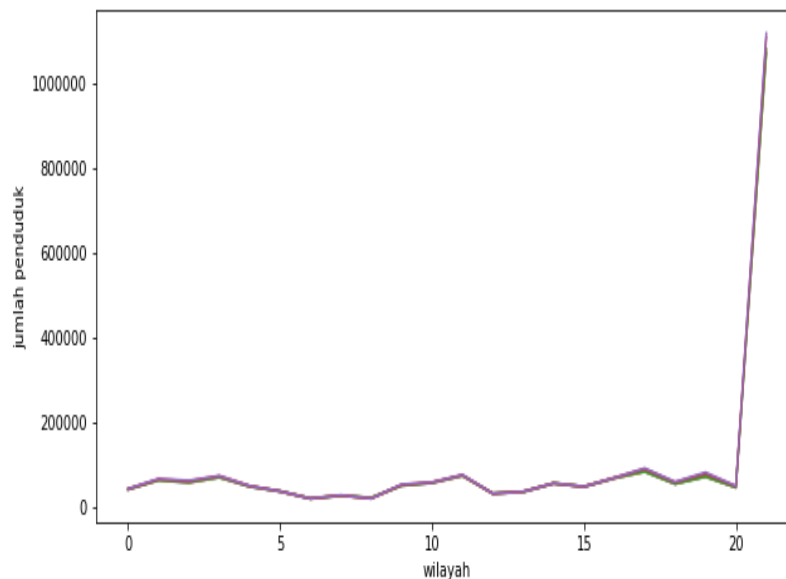
Mari memulainya dengan mengimpor library:

```
In [11]: import matplotlib.pyplot as plt
```

```
In [17]: plt.figure(figsize=(10,5))
plt.plot(data['2011'],label = '2011')
plt.plot(data['2012'],label = '2012')
plt.plot(data['2013'],label = '2013')
plt.plot(data['2014'],label = '2014')
plt.plot(data['2015'],label = '2015')
plt.xlabel('wilayah')
plt.ylabel('jumlah penduduk')
```

Penjelasan dari data yang di input adalah dimama (figsize) menjelaskan mengenai skala dari grafik 2 dimensi yang akan ditampilkan,pada gradik ini menampilkan jumla populasi penduduk di setiap wilayah tersebut.

```
Out[17]: Text(0, 0.5, 'jumlah penduduk')
```



Saya mengambil data tersebut seperti yang idtampilkan pada grafik diatas karena ingin menunjukkan hasil data secara stastik melalui grafik 2 dimensi diatas agar lebih mudah dimengerti oleh pembaca dimana grafik tersebut memuat data hasil pertumbuhan penduduk atau populasi penduduk di wilayah Medan,Sumatera Utara dan sekitarnya. Dari hasil grafik tersebut dapat dilihat adanya pertambahan jumlah penduduk yang meningkat secara signifikat dari tahun 2011 sampai dengan 2015.

