

Laporan Praktikum Pertemuan 9

Data Science Lanjut

Dibuat Oleh

Nama : Muhamad Faisal Halim
NIM : 19.240.0163
Kelas : -
Mata Kuliah : Data Science Lanjut

Mahasiswa Pertukaran Mahasiswa.
Universitas Muhammadiyah Kalimantan Timur
~ STMIK Widya Pratama Pekalongan

Note

Data pada praktikum ini disamakan dengan data yang ada pada contoh yang diberikan di Openlearning UMKT.

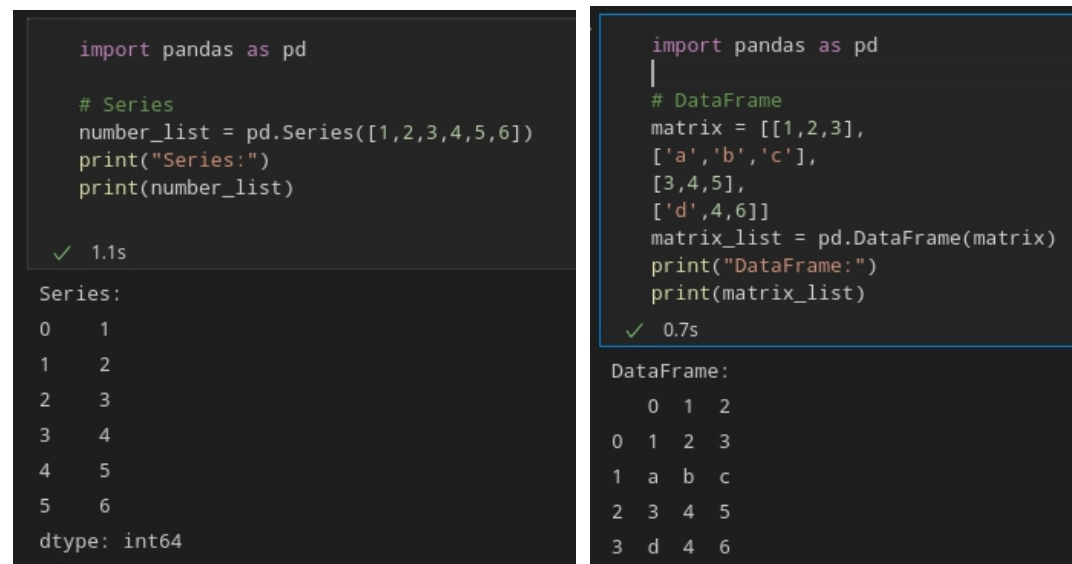
Materi dan Praktikum

title : **Manipulasi Data Dengan Pandas**

Pandas adalah sebuah library di Python yang berlisensi BSD dan open source yang menyediakan struktur data dan analisis data yang mudah digunakan. Pandas biasa digunakan untuk membuat tabel, mengubah dimensi data, mengecek data, dan lain sebagainya. Struktur data dasar pada Pandas dinamakan DataFrame, yang memudahkan kita untuk membaca sebuah file dengan banyak jenis format seperti file .txt, .csv, dan .tsv. Fitur ini akan menjadikannya table dan juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan teknik lainnya yang terdapat pada SQL.

Dapat disimpulkan, bahwa Pandas merupakan library analisis data yang diperlukan untuk membersihkan data mentah ke dalam sebuah bentuk yang bisa untuk diolah.

1. Series: satu kolom bagian dari tabel dataframe yang merupakan 1 dimensional numpy array sebagai basis data nya, terdiri dari 1 tipe data (integer, string, float, dll).
2. DataFrame: gabungan dari Series, berbentuk rectangular data yang merupakan tabel spreadsheet itu sendiri (karena dibentuk dari banyak Series, tiap Series biasanya punya 1 tipe data, yang artinya 1 dataframe bisa memiliki banyak tipe data).



The image shows two side-by-side screenshots of Jupyter Notebook code cells. The left cell demonstrates creating a Pandas Series, and the right cell demonstrates creating a Pandas DataFrame.

```
import pandas as pd

# Series
number_list = pd.Series([1,2,3,4,5,6])
print("Series:")
print(number_list)
```

✓ 1.1s

Series:

0	1
1	2
2	3
3	4
4	5
5	6

dtype: int64

```
import pandas as pd

# DataFrame
matrix = [[1,2,3],
['a','b','c'],
[3,4,5],
['d',4,6]]
matrix_list = pd.DataFrame(matrix)
print("DataFrame:")
print(matrix_list)
```

✓ 0.7s

DataFrame:

	0	1	2
0	1	2	3
1	a	b	c
2	3	4	5
3	d	4	6

Pandas dataframe dan series memiliki banyak sekali attribut, dan pada pembelajaran ini ada beberapa yang sudah di pakah pada sub bab sebelumnya. untuk tahu lebih mengenai attribut apa yang ada bisa dilihat pada link berikut

Dataframe : <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html>

Series : <https://pandas.pydata.org/docs/reference/series.html>

<code>DataFrame.info()</code>	Metode ini mencetak informasi tentang DataFrame termasuk indeks dtype dan kolom, nilai non-null dan penggunaan memori
<code>DataFrame.shape</code>	Mencetak tuple yang mewakili dimensi DataFrame
<code>DataFrame.dtype</code>	Mencetak tipe data di tiap kolom yang ada
<code>DataFrame.astype()</code>	digunakan untuk convert tipe data berdasarkan tipe data seperti float, int, str, numpy.float dll.
<code>DataFrame.copy()</code>	digunakan untuk duplikasi data yang tersimpan didalam sebuah variable dan disimpan di variable baru.
<code>Series.to_list()</code>	Merubah series menjadi list.
<code>Series.unique()</code>	Mengembalikan nilai unik dari suatu kolom dalam bentuk array
<code>Series.index()</code>	The index (axis labels) of the Series
<code>DataFrame.index()</code>	Indeks (label baris) dari DataFrame
<code>DataFrame.column</code>	Mengetahui kolom apa saja dari dataframe.
<code>DataFrame.loc[]</code> <code>Series.loc[]</code>	Akses grup baris dan kolom berdasarkan label atau array boolean
<code>DataFrame.iloc[]</code> <code>Series.iloc[]</code>	Akses grup baris dan kolom berdasarkan index kolom atau index
<code>DataFrame.size</code>	Kembalikan int yang mewakili jumlah elemen dalam objek ini.

dari daftar diatas masih banyak lagi yang belum bisa saya sebutkan.

Membuat Dataframe Dan Series

Untuk membuat DataFrame atau Series bisa dari berbagai macam tipe data di python, seperti list, dictionary, maupun numpy array.

pada praktik dibawah ini saya akan mencoba membuat DataFrame dan Series dari list di python.

```
# dataframe dari list
listku = [
    [1,"a"],
    [2,"b"],
    [3,"c"],
]
index = [0,1,2]
cols = ['float', 'char']

dataframe = pd.DataFrame(listku, index=index, columns=cols)
print(dataframe)
```

✓ 0.7s

	float	char
0	1	a
1	2	b
2	3	c

```
import pandas as pd

cth_list = [1,2,"a","b",3,"c"]
seriesku = pd.Series(cth_list)
print(seriesku)
```

✓ 0.5s

0	1
1	2
2	a
3	b
4	3
5	c

dtype: object

Menggabungkan Dataset atau Series

pada pandas ada beberapa metode menggabungkan series atau dataframe, misal merge, concat, append, join.

berikut adalah contoh menggabungkan series dengan append

```
import pandas as pd

data1 = pd.Series([1,2,3])
data2 = pd.Series([6,7,8])
|
appended = data1.append(data2)
print(appended)
```

✓ 0.6s

0	1
1	2
2	3
0	6
1	7
2	8

dtype: int64

Dataset I/O

Terdapat sangat banyak file yang bisa dibaca dengan pandas, namun pada kebanyakan kasus ada beberapa file yang sering digunakan seperti. CSV, TSV, Excel, Google BigQuery, SQL Query, JSON. dan hasil output yang diberikan dapat berupa series ataupun dataframe.

Pada dasarnya tipe data CSV dan TSV itu sama, hanya berbeda metode pemisahan data pada kedua file itu, jika CSV menggunakan comma dan TSV menggunakan tab.

dengan menggunakan fungsi `Pandas.read_csv()` kita dapat dengan mudah membaca file csv, dengan sparator pemisah comma “,” namun kita juga dapat merubah value dari sparaator tersebut, misal diubah menjadi “;”, “|”, atau tab “\t” tergantung pemisah pada file CSV atau kita, karena pada beberapa kasus terdapat file CSV yang pemisahannya menggunakan “;”.

<code>pandas.read_csv()</code>	Membaca File CSV atau TSV
<code>pandas.read_excel()</code>	Membaca File Excel
<code>pandas.read_json()</code>	Membaca File Json

```
import pandas as pd
# File CSV
df_csv = pd.read_csv("https://dqlab-dataset.s3-ap-southeast-1.amazonaws.com/sample_csv.csv")
print(df_csv.head(3))
```

✓ 0.4s

	order_id	order_date	customer_id	city	province	product_id	\
0	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P0648	
1	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P3826	
2	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P1508	

```
import pandas as pd
# File TSV
df_tsv = pd.read_csv("https://dqlab-dataset.s3-ap-southeast-1.amazonaws.com/sample_tsv.tsv", sep='\t')
print(df_tsv.head(3))
```

✓ 0.8s

	order_id	order_date	customer_id	city	province	product_id	\
0	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P0648	
1	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P3826	
2	1612339	2019-01-01	18055	Jakarta Selatan	DKI Jakarta	P1508	

Dalam praktik dibidang data scientist ada kalanya kita harus membuat dataset. dan pandas udah menyediakan fungsi yang dapat kita gunakan.

<code>.to_csv()</code>	Tulis objek ke file nilai yang dipisahkan koma (csv)
<code>.to_json()</code>	Ubah objek menjadi string JSON. Catatan NaN's dan None akan dikonversi menjadi null dan objek datetime akan dikonversi ke stempel waktu UNIX.
<code>.to_excel()</code>	Tulis objek ke lembar Excel. Untuk menulis satu objek ke file .xlsx Excel, Anda hanya perlu menentukan nama file target.
<code>.to_sql()</code>	Menulis catatan yang disimpan dalam DataFrame ke database SQL.