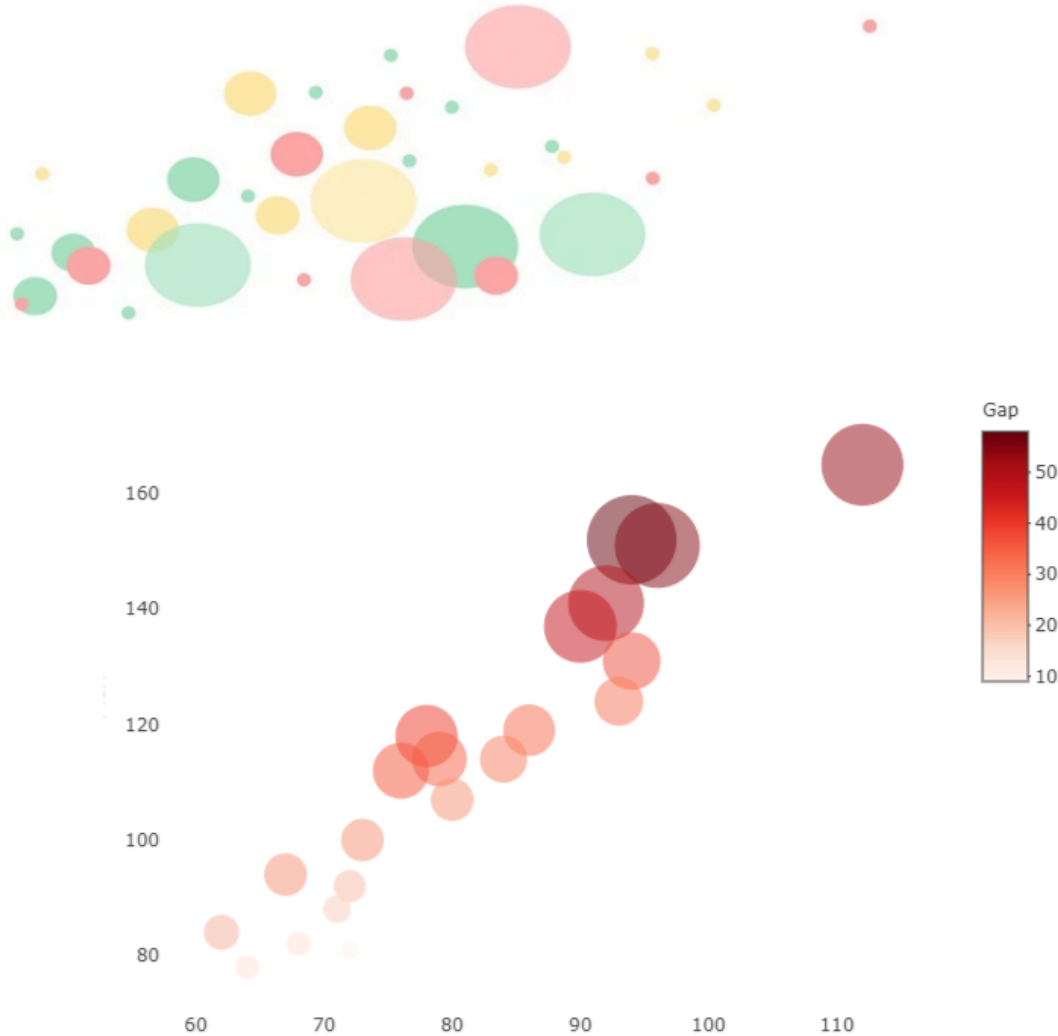


# Visualisasi Data dengan Python

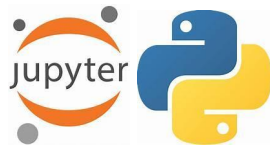


Prana Ugiana Gio (Universitas Sumatera Utara)

Egi Akbar Anagio (Australian National University)



# VISUALISASI DATA DENGAN PYTHON



**Prana Ugiana Gio**

**Diupdate pada 19 Desember 2023**

<https://statcal.com/> | <https://statkomat.com/>

**KATA PENGANTAR**

Alhamdulillah, puji syukur atas kehadiran Allah SWT, karena atas izin-Nya, penulis dapat terus mempertahankan semangat untuk menulis, dan akhirnya dapat menyelesaikan buku ini. Hadirnya buku ini, tidak semata-mata atas usaha penulis sendiri, melainkan atas izin-Nya. Sungguh suatu kebahagiaan bagi penulis bisa berbagi sebagian kecil ilmu pengetahuan milik-Nya melalui buku yang berjudul “**Visualisasi Data dengan Python**”.

Ucapan terima kasih penulis sampaikan kepada semua pihak yang telah membantu dalam rangka penyelesaian buku ini. Penulis menyadari bahwa buku ini tentunya masih perlu perbaikan, sehingga penulis mengharapkan kritik dan saran yang membangun dari para pembaca agar buku ini dapat menjadi lebih baik. Kritik dan saran dapat ditujukan ke alamat email [gioprana89@gmail.com](mailto:gioprana89@gmail.com).

Medan, 19 Desember 2023

Prana Ugiana Gio

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# DAFTAR ISI

**BAB 1 MENGIMPORT & MENCETAK DATA**

**BAB 2 MENGAkses BARIS DAN KOLOM PADA DATAFRAME**

**BAB 3 OPERASI STATISTIKA PADA DATAFRAME**

**BAB 4 MEMBUAT GRAFIK BATANG**

**BAB 5 MEMBUAT GRAFIK GARIS**

**BAB 6 MEMBUAT GRAFIK LINGKARAN**

**BAB 7 MEMBUAT GRAFIK SCATTERPLOT**

**BAB 8 MEMBUAT PETA**

**BAB 9 SEKILAS MEMBUAT APLIKASI BERBASIS WEBSITE DENGAN  
STREAMLIT**



# BAB 1 MENGIMPORT & MENCETAK DATA

---

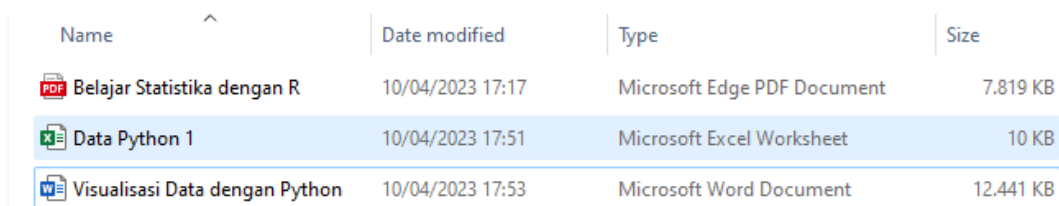
## 1.1 Mengimport Data File Microsoft Excel (.xlsx)




Misalkan diberikan data seperti pada Tabel 1.1.

**Tabel 1.1 Contoh Data**

Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
Andi	Laki-Laki	S1	8.6	6.1
Budi	Laki-Laki	S2	7.8	6.7
Tono	Laki-Laki	S1	7.4	6.2
Roni	Laki-Laki	S1	8	7.7
Bagus	Laki-Laki	S2	8.8	6.3
Amin	Laki-Laki	S1	7.6	5.1
Rini	Perempuan	S2	7.8	6.5
Fini	Perempuan	S1	7.6	5.1
Fina	Perempuan	S1	8.4	5.9
Sari	Perempuan	S2	9	7.4
Arni	Perempuan	S1	7	4.5
Wawan	Laki-Laki	S2	8	5.6
Andri	Laki-Laki	S1	8	5.5
Roy	Laki-Laki	S2	9	7.2
Didi	Laki-Laki	S1	7	6.5
Vivi	Perempuan	S2	8.2	5.8
Tari	Perempuan	S1	7.9	4.4
Cici	Perempuan	S2	6.5	4.5

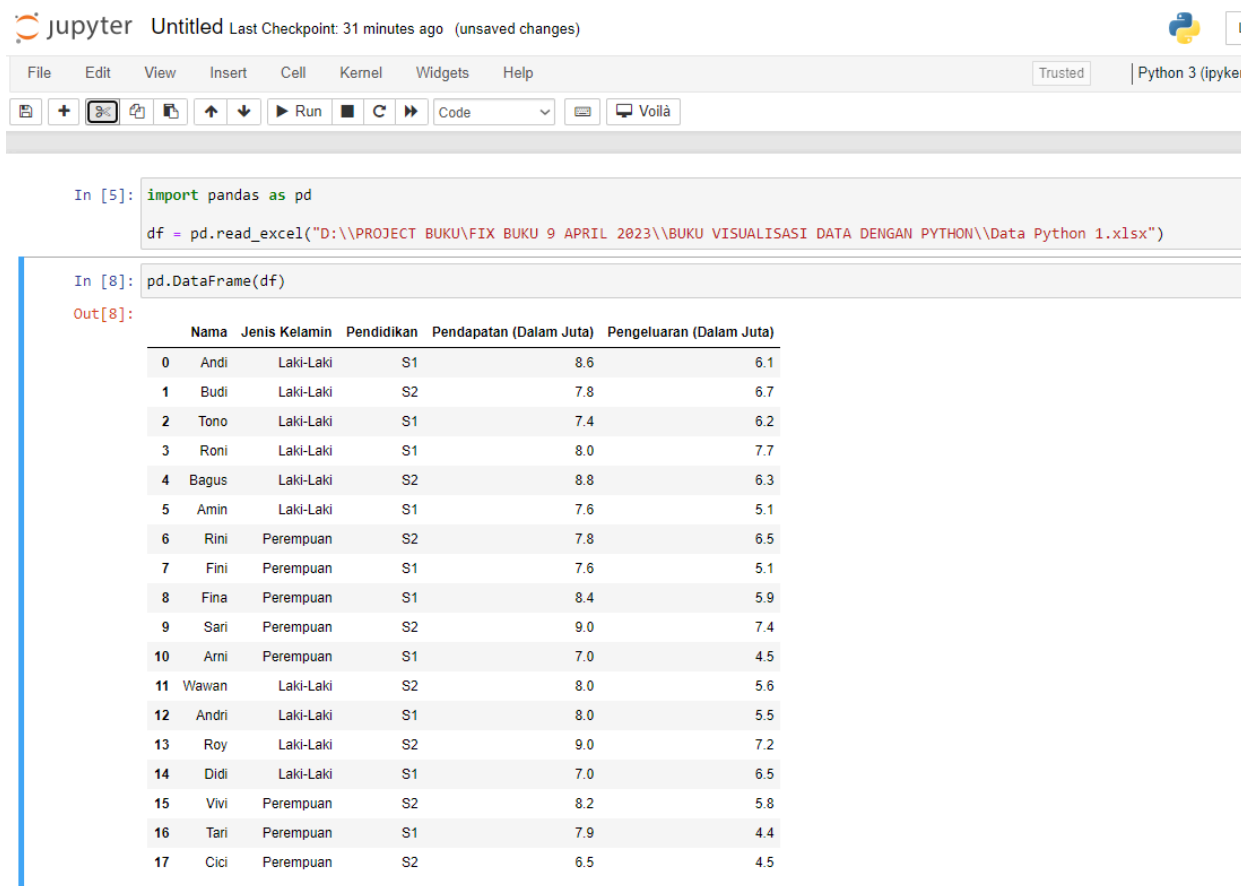
Data pada Tabel 1.1 kemudian disimpan dalam format **.xlsx**, dengan nama **Data Python 1.xlsx** (Gambar 1.1).



Name	Date modified	Type	Size
 Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
 Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
 Visualisasi Data dengan Python	10/04/2023 17:53	Microsoft Word Document	12.441 KB

**Gambar 1.1 Data Python 1.xlsx**

Selanjutnya data pada Tabel 1.1 akan diimport ke dalam Python seperti pada Gambar 1.2.



```
In [5]: import pandas as pd
df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")

In [8]: pd.DataFrame(df)
Out[8]:
```

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

**Gambar 1.2 Import Data Excel**

Pada perintah Python

```
import pandas as pd
```

dapat diartikan mengimport pustaka atau library **pandas** dan memberi nama atau *nickname* **pd**. Kemudian perintah Python

```
df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")
```

berarti membaca data dengan nama **Data Python 1.xlsx** pada alamat **D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON**.

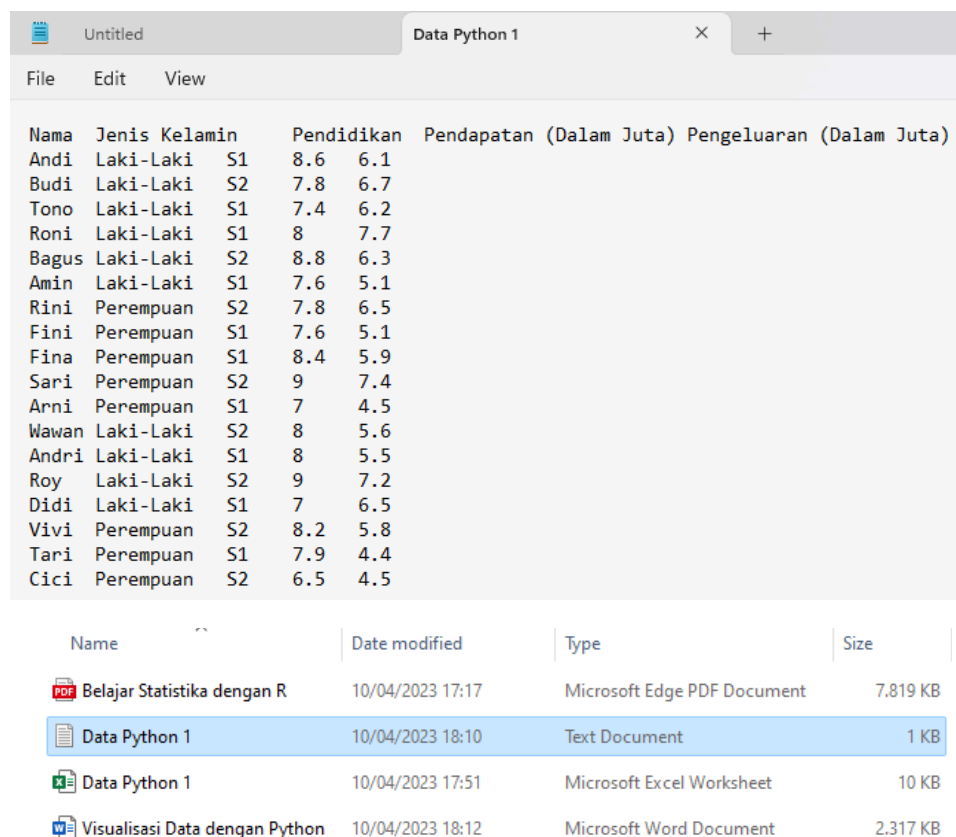
Data Microsoft Excel yang dibaca diberi nama **df**. Perintah Python

**pd.DataFrame(df)**

berarti menyajikan data **df** dalam bentuk *data frame*, seperti pada Gambar 1.2.

## 1.2 Mengimport Data File Text Document (.txt)

Misalkan data pada Tabel 1.1 disimpan dalam bentuk *text document*, yakni **Data Python 1.txt**, seperti pada Gambar 1.3.



The image shows a text editor window titled 'Data Python 1' containing a table with 5 columns: Nama, Jenis Kelamin, Pendidikan, Pendapatan (Dalam Juta), and Pengeluaran (Dalam Juta). Below the table is a file explorer showing a list of files, with 'Data Python 1' (Text Document, 1 KB) highlighted.

Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
Andi	Laki-Laki	S1	8.6	6.1
Budi	Laki-Laki	S2	7.8	6.7
Tono	Laki-Laki	S1	7.4	6.2
Roni	Laki-Laki	S1	8	7.7
Bagus	Laki-Laki	S2	8.8	6.3
Amin	Laki-Laki	S1	7.6	5.1
Rini	Perempuan	S2	7.8	6.5
Fini	Perempuan	S1	7.6	5.1
Fina	Perempuan	S1	8.4	5.9
Sari	Perempuan	S2	9	7.4
Arni	Perempuan	S1	7	4.5
Wawan	Laki-Laki	S2	8	5.6
Andri	Laki-Laki	S1	8	5.5
Roy	Laki-Laki	S2	9	7.2
Didi	Laki-Laki	S1	7	6.5
Vivi	Perempuan	S2	8.2	5.8
Tari	Perempuan	S1	7.9	4.4
Cici	Perempuan	S2	6.5	4.5

Name	Date modified	Type	Size
Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
<b>Data Python 1</b>	<b>10/04/2023 18:10</b>	<b>Text Document</b>	<b>1 KB</b>
Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
Visualisasi Data dengan Python	10/04/2023 18:12	Microsoft Word Document	2.317 KB

**Gambar 1.3**

Selanjutnya data **Data Python 1.txt** diimport ke dalam Python seperti Gambar 1.4. Pada Gambar 1.4, perintah Python

```
df = pd.read_table('D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.txt')
```

berarti membaca data dengan nama **Data Python 1.txt** pada alamat **D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI DATA DENGAN PYTHON**. Data yang dibaca diberi nama **df**. Perintah Python

**pd.DataFrame(df)**

berarti menyajikan data **df** dalam bentuk *data frame*, seperti pada Gambar 1.4.

```
In [14]: df = pd.read_table('D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI DATA DENGAN PYTHON\Data Python 1.txt')
pd.DataFrame(df)
```

Out[14]:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

**Gambar 1.4**

### 1.3 Mengimport Data File Comma Separated Values (.csv)

Misalkan data pada Tabel 1.1 disimpan dalam bentuk *comma separated values*, yakni **Data Python 1.csv**, seperti pada Gambar 1.5. Selanjutnya data **Data Python 1.csv** diimport ke dalam Python seperti Gambar 1.6. Pada Gambar 1.6, perintah Python

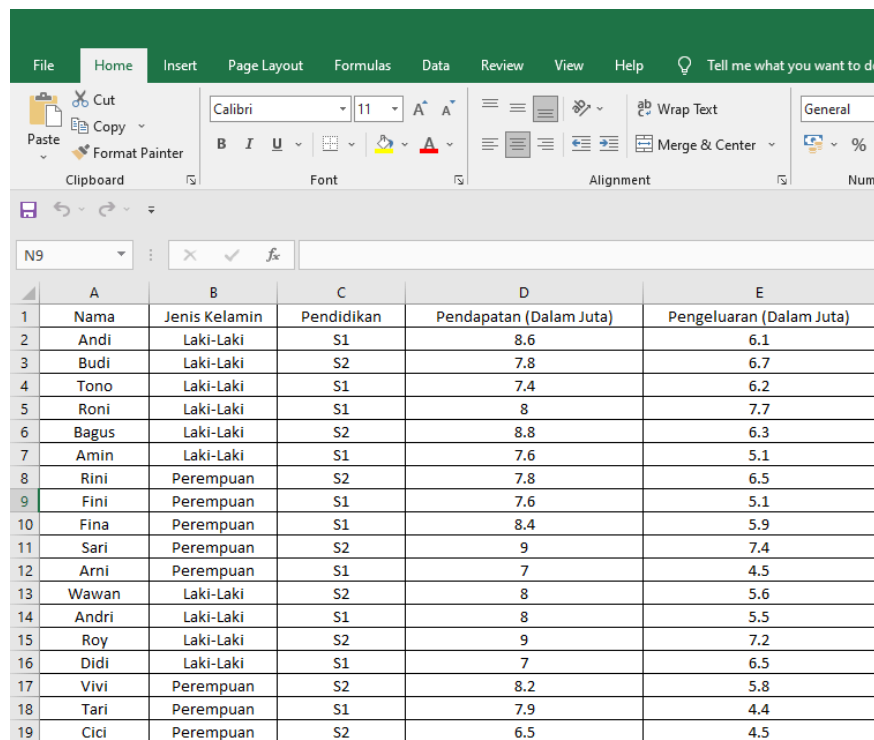
```
df = pd.read_csv('D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI DATA DENGAN PYTHON\Data Python 1.csv', sep = ';')
```

berarti membaca data dengan nama **Data Python 1.csv** pada alamat **D:\\PROJECT BUKUFIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON.**

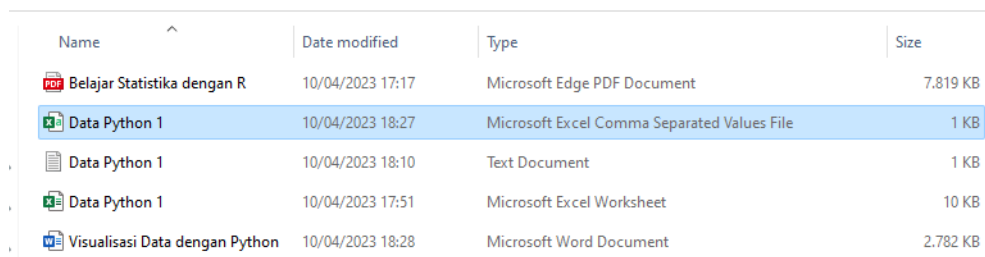
Data yang dibaca diberi nama **df**. Perintah Python

**pd.DataFrame(df)**

berarti menyajikan data **df** dalam bentuk *data frame*, seperti pada Gambar 1.6.



	A	B	C	D	E
1	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
2	Andi	Laki-Laki	S1	8.6	6.1
3	Budi	Laki-Laki	S2	7.8	6.7
4	Tono	Laki-Laki	S1	7.4	6.2
5	Roni	Laki-Laki	S1	8	7.7
6	Bagus	Laki-Laki	S2	8.8	6.3
7	Amin	Laki-Laki	S1	7.6	5.1
8	Rini	Perempuan	S2	7.8	6.5
9	Fini	Perempuan	S1	7.6	5.1
10	Fina	Perempuan	S1	8.4	5.9
11	Sari	Perempuan	S2	9	7.4
12	Arni	Perempuan	S1	7	4.5
13	Wawan	Laki-Laki	S2	8	5.6
14	Andri	Laki-Laki	S1	8	5.5
15	Roy	Laki-Laki	S2	9	7.2
16	Didi	Laki-Laki	S1	7	6.5
17	Vivi	Perempuan	S2	8.2	5.8
18	Tari	Perempuan	S1	7.9	4.4
19	Cici	Perempuan	S2	6.5	4.5



Name	Date modified	Type	Size
Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
Data Python 1	10/04/2023 18:27	Microsoft Excel Comma Separated Values File	1 KB
Data Python 1	10/04/2023 18:10	Text Document	1 KB
Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
Visualisasi Data dengan Python	10/04/2023 18:28	Microsoft Word Document	2.782 KB

**Gambar 1.5**

In [17]:

```
df = pd.read_csv("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.csv", sep = ';')
pd.DataFrame(df)
```

Out[17]:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

Gambar 1.6

### 1.4 Mencetak Data ke Dalam Microsoft Excel (.xlsx)

Data yang telah diimport ke dalam Python, dapat dicetak ke dalam Microsoft Excel. Perhatikan Gambar 1.7. Misalkan data pada Gambar 1.7 ingin dicetak ke dalam Excel, dengan nama file *cetakdata.xlsx*, seperti pada Gambar 1.8.

Jupyter Untitled1 Last Checkpoint: 25 minutes ago (unsaved changes)

Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [1]:

```
import pandas as pd
df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")
pd.DataFrame(df)
```

Out[1]:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

Gambar 1.7

Name	Date modified	Type	Size
Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
cetakdata	11/04/2023 00:48	Microsoft Excel Worksheet	6 KB
Data Python 1	10/04/2023 18:27	Microsoft Excel Comma Separated Values File	1 KB
Data Python 1	10/04/2023 18:10	Text Document	1 KB
Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
Visualisasi Data dengan Python	11/04/2023 00:51	Microsoft Word Document	3.235 KB

	A	B	C	D	E	F
1	<b>Nama</b>	<b>Jenis Kelamin</b>	<b>Pendidikan</b>	<b>Pendapatan (Dalam Juta)</b>	<b>Pengeluaran (Dalam Juta)</b>	
2	Andi	Laki-Laki	S1	8.6	6.1	
3	Budi	Laki-Laki	S2	7.8	6.7	
4	Tono	Laki-Laki	S1	7.4	6.2	
5	Roni	Laki-Laki	S1	8	7.7	
6	Bagus	Laki-Laki	S2	8.8	6.3	
7	Amin	Laki-Laki	S1	7.6	5.1	
8	Rini	Perempuan	S2	7.8	6.5	
9	Fini	Perempuan	S1	7.6	5.1	
10	Fina	Perempuan	S1	8.4	5.9	
11	Sari	Perempuan	S2	9	7.4	
12	Arni	Perempuan	S1	7	4.5	
13	Wawan	Laki-Laki	S2	8	5.6	
14	Andri	Laki-Laki	S1	8	5.5	
15	Roy	Laki-Laki	S2	9	7.2	
16	Didi	Laki-Laki	S1	7	6.5	
17	Vivi	Perempuan	S2	8.2	5.8	
18	Tari	Perempuan	S1	7.9	4.4	
19	Cici	Perempuan	S2	6.5	4.5	

**Gambar 1.8**

Berikut perintah Python (Gambar 1.9) untuk mencetak data pada Gambar 1.7 ke dalam Excel.

```
df.to_excel(r'D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI  
DATA DENGAN PYTHON\cetakdata.xlsx', index=False)
```

Jupyter Untitled1 Last Checkpoint: 31 minutes ago (autosaved) Python 3 (ipykernel)

File Edit View Insert Cell Kernel Widgets Help Trusted Voilà

```
pd.DataFrame(df)
```

Out[1]:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

```
In [7]: df.to_excel(r'D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI DATA DENGAN PYTHON\cetakdata.xlsx', index=False)
```

In [ ]:

Gambar 1.9

### 1.5 Mencetak Data ke Dalam Comma Separated Values (.csv)

Misalkan data pada Gambar 1.7 ingin dicetak ke dalam jenis *file* .csv, dengan nama **cetakdata.csv**, seperti pada Gambar 1.10. Berikut disajikan perintah Python.

```
df.to_csv(r'D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU VISUALISASI  
DATA DENGAN PYTHON\cetakdata.csv', index=False, sep = ';')
```

Name	Date modified	Type	Size
Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
cetakdata	11/04/2023 00:57	Microsoft Excel Comma Separated Values File	1 KB
cetakdata	11/04/2023 00:53	Microsoft Excel Worksheet	10 KB
Data Python 1	10/04/2023 18:27	Microsoft Excel Comma Separated Values File	1 KB
Data Python 1	10/04/2023 18:10	Text Document	1 KB
Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
Visualisasi Data dengan Python	11/04/2023 01:03	Microsoft Word Document	3.691 KB

Gambar 1.10

## 1.6 Mencetak Data ke Dalam Text Document (.txt)

Misalkan data pada Gambar 1.7 ingin dicetak ke dalam jenis *file* .txt, dengan nama **cetakdata.txt**, seperti pada Gambar 1.11. Berikut disajikan perintah Python.

```
df.to_csv(r'D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI  
DATA DENGAN PYTHON\\cetakdata.txt', index=False, sep = '\t')
```

The screenshot shows a Jupyter Notebook interface with the following components:

- Header:** "jupyter Untitled1 Last Checkpoint: 20 hours ago (unsaved changes)"
- Menu Bar:** File, Edit, View, Insert, Cell, Kernel, Widgets, Help
- Toolbar:** Includes icons for file operations, a "Run" button, and a "Voilà" button.
- Output:** A table with 18 rows and 6 columns. The columns are: Index, Nama, Jenis Kelamin, Pendidikan, Pendapatan (Dalam Juta), and Pengeluaran (Dalam Juta). The data is as follows:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5
- Code Cell:** A code cell with the following text:

```
In [5]: df.to_csv(r'D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\cetakdata.txt',  
index=False, sep = '\t')
```

Name	Date modified	Type	Size
Belajar Statistika dengan R	10/04/2023 17:17	Microsoft Edge PDF Document	7.819 KB
cetakdata	11/04/2023 00:57	Microsoft Excel Comma Separated Values File	1 KB
cetakdata	11/04/2023 21:52	Text Document	1 KB
cetakdata	11/04/2023 00:53	Microsoft Excel Worksheet	10 KB
Data Python 1	10/04/2023 18:27	Microsoft Excel Comma Separated Values File	1 KB
Data Python 1	10/04/2023 18:10	Text Document	1 KB
Data Python 1	10/04/2023 17:51	Microsoft Excel Worksheet	10 KB
Visualisasi Data dengan Python	11/04/2023 21:52	Microsoft Word Document	3.984 KB

The image shows a Notepad window with the title 'cetakdata'. The window contains a table with the following data:

Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
Andi	Laki-Laki	S1	8.6	6.1
Budi	Laki-Laki	S2	7.8	6.7
Tono	Laki-Laki	S1	7.4	6.2
Roni	Laki-Laki	S1	8.0	7.7
Bagus	Laki-Laki	S2	8.8	6.300000000000001
Amin	Laki-Laki	S1	7.6	5.1
Rini	Perempuan	S2	7.8	6.5
Fini	Perempuan	S1	7.6	5.1
Fina	Perempuan	S1	8.4	5.9
Sari	Perempuan	S2	9.0	7.4
Arni	Perempuan	S1	7.0	4.5
Wawan	Laki-Laki	S2	8.0	5.6
Andri	Laki-Laki	S1	8.0	5.5
Roy	Laki-Laki	S2	9.0	7.2
Didi	Laki-Laki	S1	7.0	6.5
Vivi	Perempuan	S2	8.2	5.8
Tari	Perempuan	S1	7.9	4.4
Cici	Perempuan	S2	6.5	4.5

**Gambar 1.11**

# BAB 2 MENGAKSES BARIS DAN KOLOM PADA DATAFRAME

---

## 2.1 Mengakses Data berdasarkan Kolom dan Baris

Perhatikan Gambar 2.1. Berdasarkan Gambar 2.1, ingin ditampilkan data pada variabel **Nama**, seperti pada Gambar 2.2. Maka perintah Python sebagai berikut.

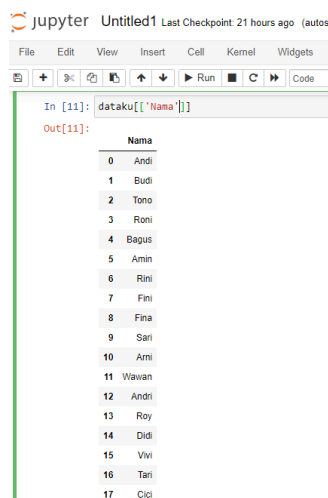
`dataku[['Nama']]`

```
In [6]: import pandas as pd
df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")
dataku = pd.DataFrame(df)
dataku
```

Out[6]:

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Ami	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

Gambar 2.1



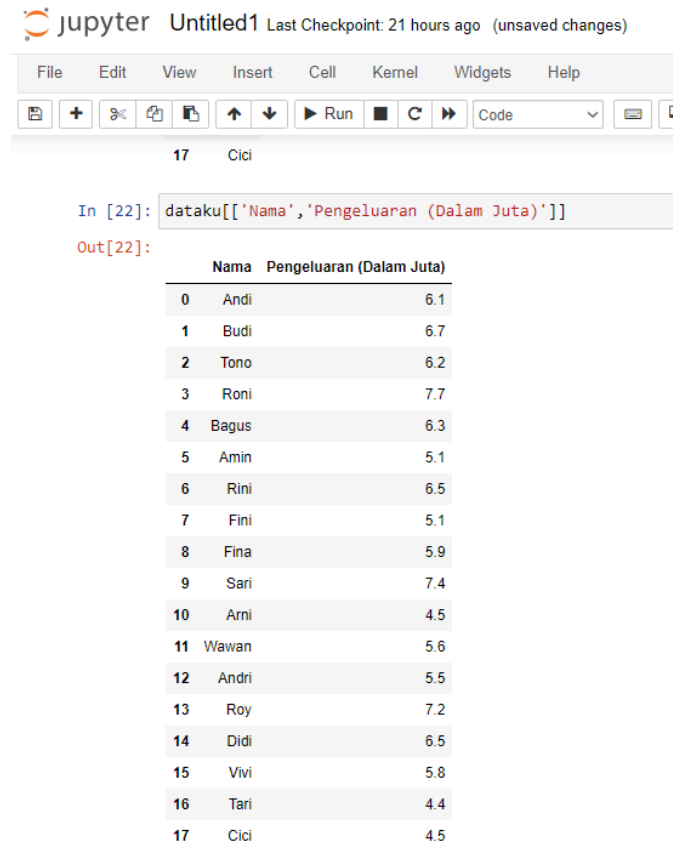
```
In [11]: dataku[['Nama']]
Out[11]:
      Nama
0      Andi
1      Budi
2      Tono
3      Roni
4      Bagus
5      Amin
6      Rini
7      Fini
8      Fina
9      Sari
10     Ami
11     Wawan
12     Andri
13     Roy
14     Didi
15     Vivi
16     Tari
17     Cici
```

Gambar 2.2

Perintah Python

```
dataku[['Nama','Pengeluaran (Dalam Juta)']]
```

menampilkan data variabel nama dan pengeluaran, seperti pada Gambar 2.3.



**Gambar 2.3**

Perhatikan Gambar 2.4. Pada Gambar 2.4, perintah Python

```
dataku.loc[:, ['Jenis Kelamin', 'Nama']]
```

menyajikan data variabel jenis kelamin dan nama.

```
In [48]: dataku.loc[:, ['Jenis Kelamin', 'Nama']]
```

```
Out[48]:
```

	Jenis Kelamin	Nama
0	Laki-Laki	Andi
1	Laki-Laki	Budi
2	Laki-Laki	Tono
3	Laki-Laki	Roni
4	Laki-Laki	Bagus
5	Laki-Laki	Amin
6	Perempuan	Rini
7	Perempuan	Fini
8	Perempuan	Fina
9	Perempuan	Sari
10	Perempuan	Arni
11	Laki-Laki	Wawan
12	Laki-Laki	Andri
13	Laki-Laki	Roy
14	Laki-Laki	Didi
15	Perempuan	Vivi
16	Perempuan	Tari
17	Perempuan	Cici

**Gambar 2.4**

```
In [49]: dataku.loc[[0,3], ['Jenis Kelamin', 'Nama']]
```

```
Out[49]:
```

	Jenis Kelamin	Nama
0	Laki-Laki	Andi
3	Laki-Laki	Roni

**Gambar 2.5**

Perintah Python pada Gambar 2.5

```
dataku.loc[[0,3], ['Jenis Kelamin', 'Nama']]
```

untuk menampilkan data pada variabel jenis kelamin dan nama, pada indeks ke-0 (baris ke-1) dan indeks ke-3 (baris ke-4). Perintah Python pada Gambar 2.6

```
dataku.iloc[:, [1, 3]]
```

untuk menampilkan data pada variabel jenis kelamin (kolom indeks ke-1) dan variabel pendapatan (kolom indeks ke-3).

```
In [66]: dataku.iloc[:, [1, 3]]
```

Out[66]:

	Jenis Kelamin	Pendapatan (Dalam Juta)
0	Laki-Laki	8.6
1	Laki-Laki	7.8
2	Laki-Laki	7.4
3	Laki-Laki	8.0
4	Laki-Laki	8.8
5	Laki-Laki	7.6
6	Perempuan	7.8
7	Perempuan	7.6
8	Perempuan	8.4
9	Perempuan	9.0
10	Perempuan	7.0
11	Laki-Laki	8.0
12	Laki-Laki	8.0
13	Laki-Laki	9.0
14	Laki-Laki	7.0
15	Perempuan	8.2
16	Perempuan	7.9
17	Perempuan	6.5

**Gambar 2.6**

Perintah Python pada Gambar 2.7

```
dataku.iloc[[0,17], [0, 1, 3]]
```

untuk menampilkan data pada variabel nama (kolom indeks ke-0), jenis kelamin (kolom indeks ke-1) dan variabel pendapatan (kolom indeks ke-3), pada baris ke-1 (indeks baris ke-0) dan baris ke-18 (indeks baris ke-17).

```
In [70]: dataku.iloc[[0,17], [0, 1, 3]]
```

Out[70]:

	Nama	Jenis Kelamin	Pendapatan (Dalam Juta)
0	Andi	Laki-Laki	8.6
17	Cici	Perempuan	6.5

**Gambar 2.7**

## 2.2 Seleksi Data

Perhatikan Gambar 2.8. Berdasarkan Gambar 2.8, ingin ditampilkan data khusus pada responden dengan Pendidikan S1. Berikut perintah Python.

```
dataku[dataku["Pendidikan"] == 'S1']
```

```
In [80]: dataku[dataku["Pendidikan"] == 'S1']
```

```
Out[80]:
```

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
5	Amin	Laki-Laki	S1	7.6	5.1
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
10	Arni	Perempuan	S1	7.0	4.5
12	Andri	Laki-Laki	S1	8.0	5.5
14	Didi	Laki-Laki	S1	7.0	6.5
16	Tari	Perempuan	S1	7.9	4.4

```
In [81]: dataku[dataku["Pendidikan"] == 'S1'].shape
```

```
Out[81]: (10, 5)
```

**Gambar 2.8**

Pada Gambar 2.8, perintah Python

```
dataku[dataku["Pendidikan"] == 'S1'].shape
```

untuk mengetahui jumlah baris dan kolom. Diketahui jumlah baris sebanyak 10 baris, sementara jumlah kolom sebanyak 5 kolom.

Perhatikan Gambar 2.9. Berdasarkan Gambar 2.9, ingin ditampilkan data khusus pada responden dengan Pendidikan S1 dan jenis kelamin laki-laki. Berikut perintah Python.

```
dataku[(dataku["Pendidikan"] == 'S1') & (dataku["Jenis Kelamin"] == 'Laki-Laki')]
```

```
In [82]: dataku[(dataku["Pendidikan"] == 'S1') & (dataku["Jenis Kelamin"] == 'Laki-Laki')]
```

```
Out[82]:
```

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
5	Amin	Laki-Laki	S1	7.6	5.1
12	Andri	Laki-Laki	S1	8.0	5.5
14	Didi	Laki-Laki	S1	7.0	6.5

**Gambar 2.9**

# BAB 3 OPERASI STATISTIKA PADA DATAFRAME

## 3.1 Menghitung Nilai Minimum, Maksimum, Rata-Rata dan Standar Deviasi

Diberikan data seperti pada Gambar 3.1. Berdasarkan data pada Gambar 3.1, ingin dihitung nilai minimum, maksimum, rata-rata dan standar deviasi dari pendapatan dan pengeluaran. Perhatikan Gambar 3.2.

The screenshot shows a Jupyter Notebook interface with the following components:

- Header:** jupyter BUKU PYTHON Last Checkpoint: Yesterday at 12:24 AM (unsaved changes)
- Menu Bar:** File, Edit, View, Insert, Cell, Kernel, Widgets, Help
- Toolbar:** Includes icons for file operations, a 'Run' button, and a 'Voilà' button.
- Code Cell (In [84]):**

```
import pandas as pd

df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")

dataku = pd.DataFrame(df)

dataku
```
- Output Cell (Out[84]):** A preview of a DataFrame with 18 rows and 6 columns. The columns are: Nama, Jenis Kelamin, Pendidikan, Pendapatan (Dalam Juta), and Pengeluaran (Dalam Juta).

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

Gambar 3.1

```

In [85]: dataku[['Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']].mean()
Out[85]: Pendapatan (Dalam Juta)    7.922222
         Pengeluaran (Dalam Juta)   5.944444
         dtype: float64

In [87]: dataku[['Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']].max()
Out[87]: Pendapatan (Dalam Juta)    9.0
         Pengeluaran (Dalam Juta)   7.7
         dtype: float64

In [88]: dataku[['Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']].min()
Out[88]: Pendapatan (Dalam Juta)    6.5
         Pengeluaran (Dalam Juta)   4.4
         dtype: float64

In [90]: dataku[['Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']].std()
Out[90]: Pendapatan (Dalam Juta)    0.690742
         Pengeluaran (Dalam Juta)   0.985317
         dtype: float64

```

**Gambar 3.2**

Berdasarkan Gambar 3.2:

- ⇒ Diketahui nilai minimum dari pendapatan adalah 6.5, sementara nilai maksimum adalah 9. Rata-rata pendapatan adalah 7.922222, dengan standar deviasi 0.690742.
- ⇒ Diketahui nilai minimum dari pengeluaran adalah 4.4, sementara nilai maksimum adalah 7.7. Rata-rata pengeluaran adalah 5.944444, dengan standar deviasi 0.985317.

```

In [93]: dat = dataku[['Jenis Kelamin', 'Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']]
         dat.groupby('Jenis Kelamin').mean()
Out[93]:
              Pendapatan (Dalam Juta)  Pengeluaran (Dalam Juta)
Jenis Kelamin
Laki-Laki                8.02                6.2900
Perempuan                7.80                5.5125

In [94]: dat = dataku[['Pendidikan', 'Pendapatan (Dalam Juta)', 'Pengeluaran (Dalam Juta)']]
         dat.groupby('Pendidikan').mean()
Out[94]:
              Pendapatan (Dalam Juta)  Pengeluaran (Dalam Juta)
Pendidikan
S1                7.7500                5.70
S2                8.1375                6.25

```

**Gambar 3.3**

Gambar 3.3 disajikan perhitungan nilai rata-rata dari pendapatan dan pengeluaran berdasarkan jenis kelamin dan pendidikan.

- ⇒ Diketahui rata-rata pendapatan pada laki-laki adalah 8.02, sementara pada perempuan adalah 7.80.
- ⇒ Diketahui rata-rata pengeluaran pada laki-laki adalah 7.80, sementara pada perempuan adalah 5.5125.

### 3.2 Distribusi Frekuensi

Berdasarkan data pada Gambar 3.1, ingin dihitung jumlah laki-laki dan jumlah perempuan, jumlah responden dengan pendidikan S1 dan S2, serta tabulasi silang. Hasil tersebut diperlihatkan pada Gambar 3.4.

```
In [110]: dat = dataku[['Pendidikan', 'Jenis Kelamin']]
          dat.value_counts()

Out[110]: Pendidikan  Jenis Kelamin
          S1          Laki-Laki      6
                     Perempuan      4
          S2          Laki-Laki      4
                     Perempuan      4
          dtype: int64
```

---

```
In [111]: dat = dataku[['Pendidikan']]
          dat.value_counts()

Out[111]: Pendidikan
          S1          10
          S2           8
          dtype: int64
```

---

```
In [112]: dat = dataku[['Jenis Kelamin']]
          dat.value_counts()

Out[112]: Jenis Kelamin
          Laki-Laki      10
           Perempuan      8
          dtype: int64
```

**Gambar 3.4**

Berdasarkan Gambar 3.4:

- ⇒ Diketahui jumlah responden dengan jenis kelamin laki-laki sebanyak 10, sementara perempuan sebanyak 8.
- ⇒ Diketahui jumlah responden dengan pendidikan S1 sebanyak 10, sementara S2 sebanyak 8.

## BAB 4 MEMBUAT GRAFIK BATANG

---

### 4.1 Latihan 1: Membuat Grafik Batang Frekuensi dengan Plotly

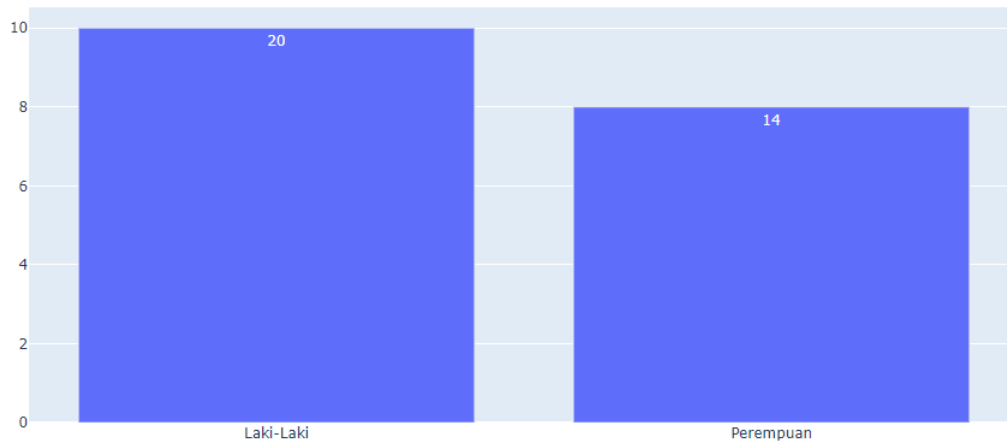
Diberikan data seperti pada Gambar 4.1. Berdasarkan data pada Gambar 4.1, ingin disajikan grafik batang frekuensi berdasarkan jenis kelamin, seperti pada Gambar 4.2.

```
In [113]: import pandas as pd
df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\Data Python 1.xlsx")
dataku = pd.DataFrame(df)
dataku
```

```
Out[113]:
```

	Nama	Jenis Kelamin	Pendidikan	Pendapatan (Dalam Juta)	Pengeluaran (Dalam Juta)
0	Andi	Laki-Laki	S1	8.6	6.1
1	Budi	Laki-Laki	S2	7.8	6.7
2	Tono	Laki-Laki	S1	7.4	6.2
3	Roni	Laki-Laki	S1	8.0	7.7
4	Bagus	Laki-Laki	S2	8.8	6.3
5	Amin	Laki-Laki	S1	7.6	5.1
6	Rini	Perempuan	S2	7.8	6.5
7	Fini	Perempuan	S1	7.6	5.1
8	Fina	Perempuan	S1	8.4	5.9
9	Sari	Perempuan	S2	9.0	7.4
10	Arni	Perempuan	S1	7.0	4.5
11	Wawan	Laki-Laki	S2	8.0	5.6
12	Andri	Laki-Laki	S1	8.0	5.5
13	Roy	Laki-Laki	S2	9.0	7.2
14	Didi	Laki-Laki	S1	7.0	6.5
15	Vivi	Perempuan	S2	8.2	5.8
16	Tari	Perempuan	S1	7.9	4.4
17	Cici	Perempuan	S2	6.5	4.5

Gambar 4.1



**Gambar 4.2**

Berdasarkan Gambar 4.2, diketahui jumlah laki-laki sebanyak 20 responden, sementara jumlah perempuan sebanyak 14 responden. Gambar 4.3 disajikan perintah Python.

```
In [148]: dat = dataku[['Jenis Kelamin']]
        hasil = dat.value_counts()
        kategori = dataku['Jenis Kelamin'].value_counts().index

In [149]: hasil
Out[149]: Jenis Kelamin
Laki-Laki      10
Perempuan      8
dtype: int64

In [150]: kategori
Out[150]: Index(['Laki-Laki', 'Perempuan'], dtype='object')

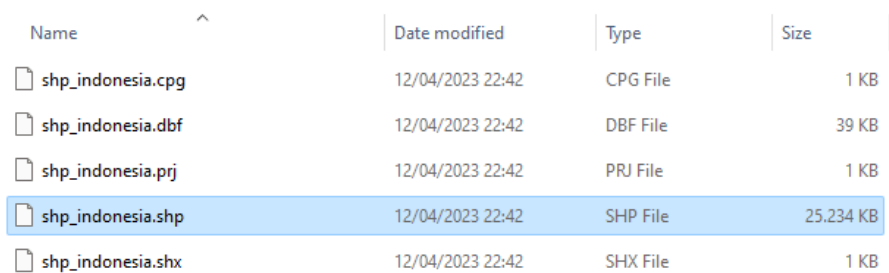
In [152]: import plotly.graph_objects as go
        fig = go.Figure([go.Bar(x = kategori, y = hasil,
                                text=y,
                                textposition='auto')])
        fig.show()
```

**Gambar 4.3**

## BAB 8 MEMBUAT PETA DENGAN GEOPANDAS

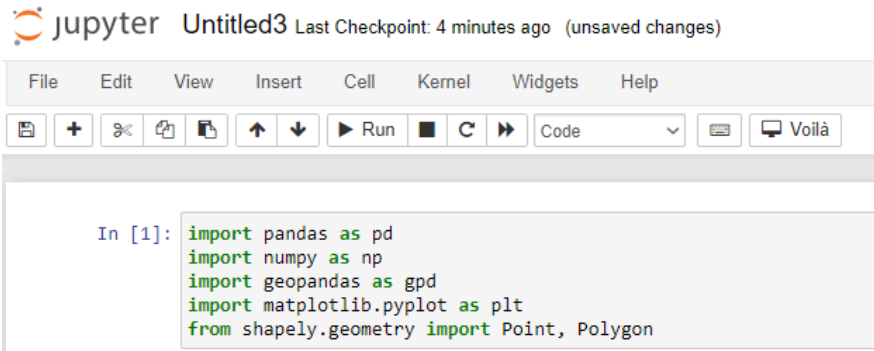
---

Pada bagian ini, kita akan berlatih membuat peta pada provinsi yang ada di Indonesia menggunakan paket atau *package* **geopandas**. Pertama, telah disediakan *file* **.shp**, dengan nama **shp\_indonesia.shp** (Gambar 8.1). Gambar 8.2 mengimport beberapa paket yang digunakan seperti **pandas**, **numpy**, **geopandas**, **matplotlib.pyplot**, dan seterusnya.



Name	Date modified	Type	Size
shp_indonesia.cpg	12/04/2023 22:42	CPG File	1 KB
shp_indonesia.dbf	12/04/2023 22:42	DBF File	39 KB
shp_indonesia.prj	12/04/2023 22:42	PRJ File	1 KB
shp_indonesia.shp	12/04/2023 22:42	SHP File	25.234 KB
shp_indonesia.shx	12/04/2023 22:42	SHX File	1 KB

**Gambar 8.1**

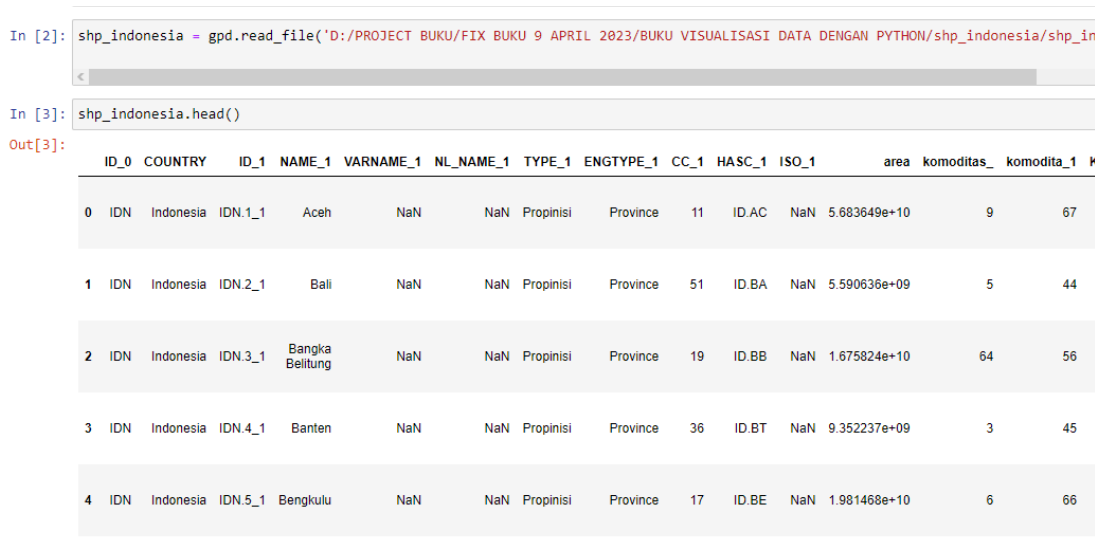


**Gambar 8.2**

Pada Gambar 8.3 diperlihatkan membaca *file shp\_indonesia.shp* pada alamat

**D:/PROJECT BUKU/FIX BUKU 9 APRIL 2023/BUKU VISUALISASI DATA  
DENGAN PYTHON/shp\_indonesia**

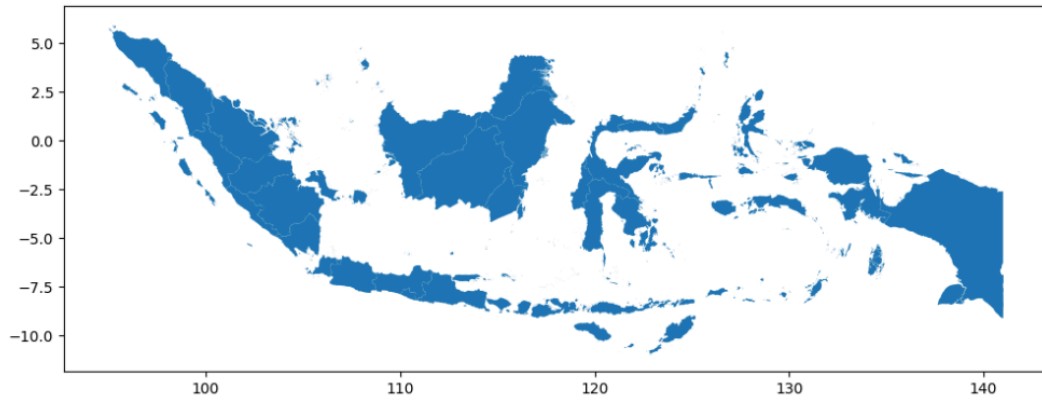
Gambar 8.4 diperlihatkan tampilan peta wilayah Indonesia. Gambar 8.5 diperlihatkan peta, dengan skala warna berdasarkan nilai *latitude*. Semakin berwarna kuning pertanda nilai *latitude* semakin tinggi, sementara semakin berwarna biru gelap pertanda nilai *latitude* semakin rendah atau minus. Gambar 8.6 diperlihatkan tampilan peta berdasarkan kluster. Terdapat 3 kluster, yakni kluster A, B dan C.



**Gambar 8.3**

```
In [11]: shp_indonesia.plot(figsize=(12,8))
```

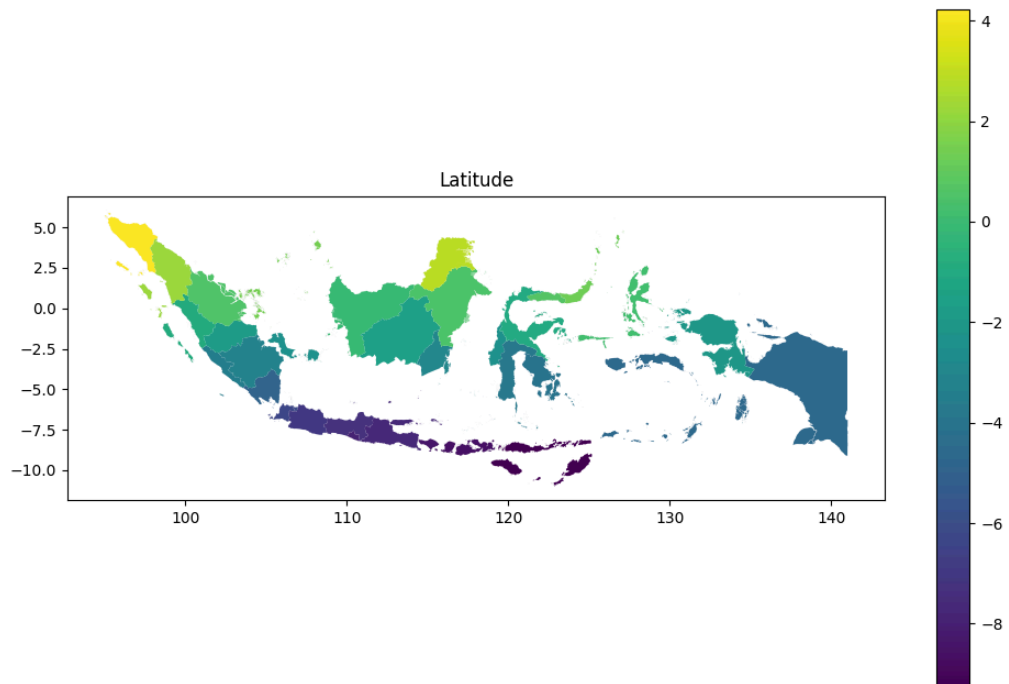
```
Out[11]: <Axes: >
```



**Gambar 8.4**

```
In [28]: shp_indonesia.plot('latitude', legend=True,figsize=(12,8))  
plt.title('Latitude')
```

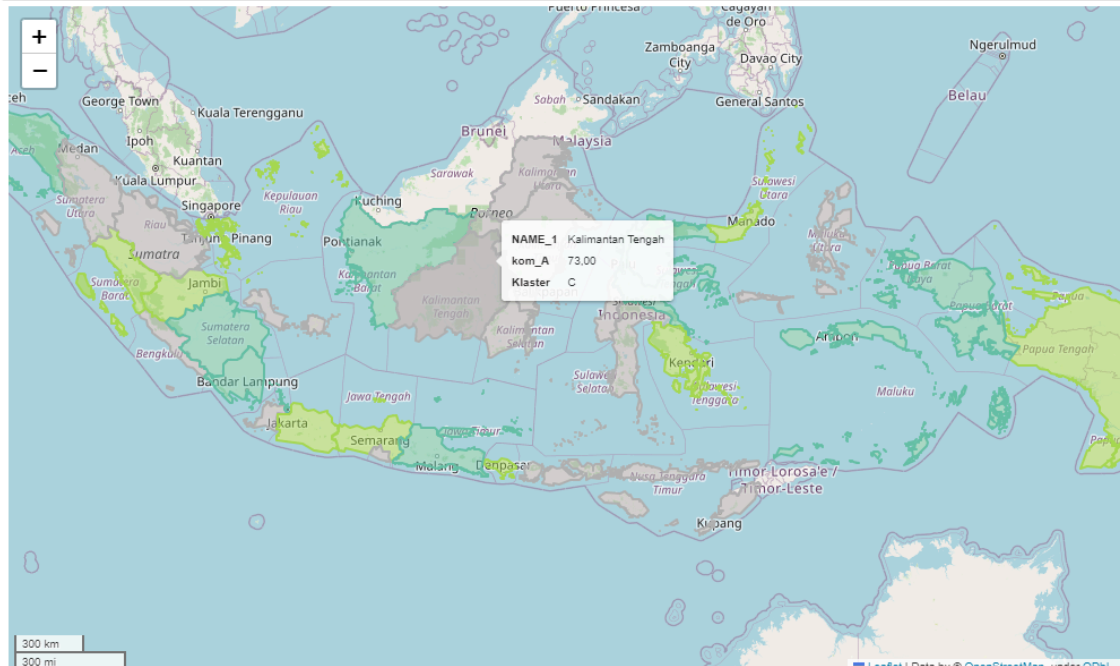
```
Out[28]: Text(0.5, 1.0, 'Latitude')
```



**Gambar 8.5**

```
In [44]: shp_indonesia.explore(column='Klaster',  
                                cmap='Set2',  
                                legend=False,  
                                tooltip=['NAME_1', 'kom_A', 'Klaster'])
```

Out[44]:



Gambar 8.6

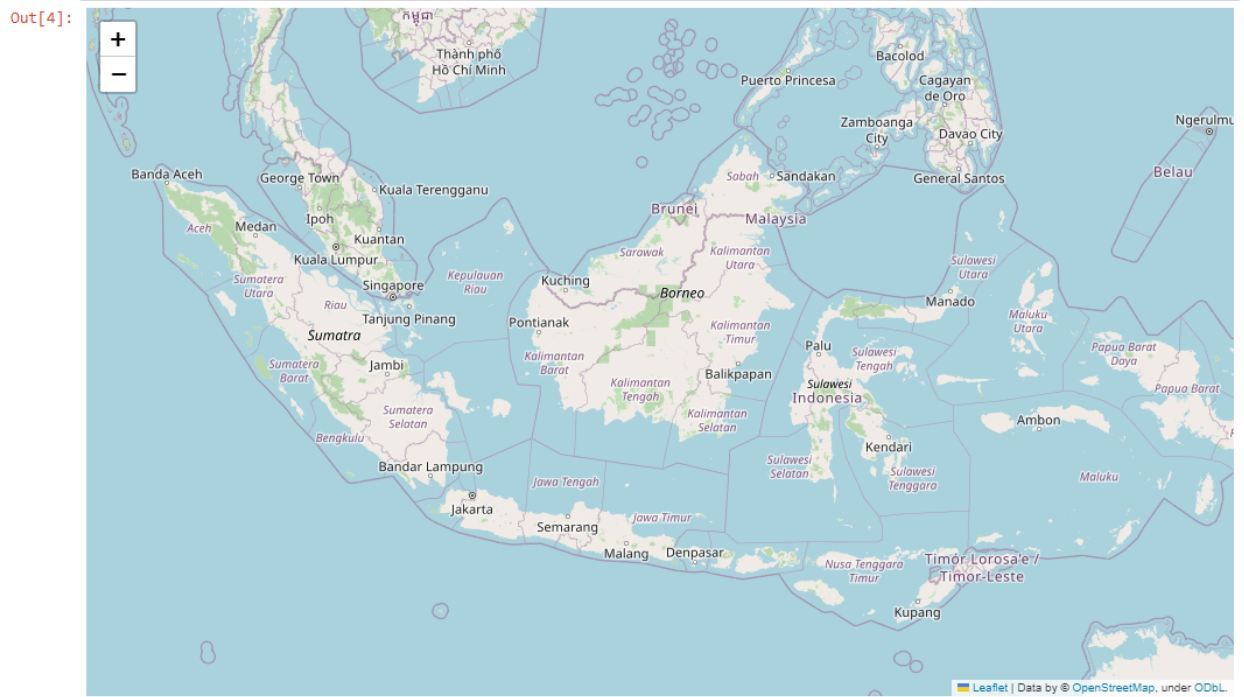
## BAB 9 MEMBUAT PETA DENGAN FOLIUM

**folium** juga merupakan salah satu *library* dalam Python yang dapat digunakan untuk membuat peta. Pada bagian ini, akan disajikan berbagai tampilan peta menggunakan *library folium*. Perhatikan Gambar 9.1. Pada Gambar 9.1 diperlihatkan perintah Python untuk menampilkan wilayah Indonesia, dengan nilai *latitude* -0.789275 dan nilai *longitude* 113.921327. Adapun *library* atau pustaka yang digunakan adalah **folium** dan **pandas**. Gambar 9.2 disajikan gambar peta dengan menampilkan *marker* pada wilayah Aceh dan Medan. Gambar 9.3 disajikan garis yang menghubungkan antara wilayah Aceh dan Padang, serta garis yang menghubungkan antara wilayah Medan dan Padang.

```
In [4]: import folium
import pandas as pd

m = folium.Map(location = [-0.789275, 113.921327],
                zoom_start = 5)

m
```



Gambar 9.1

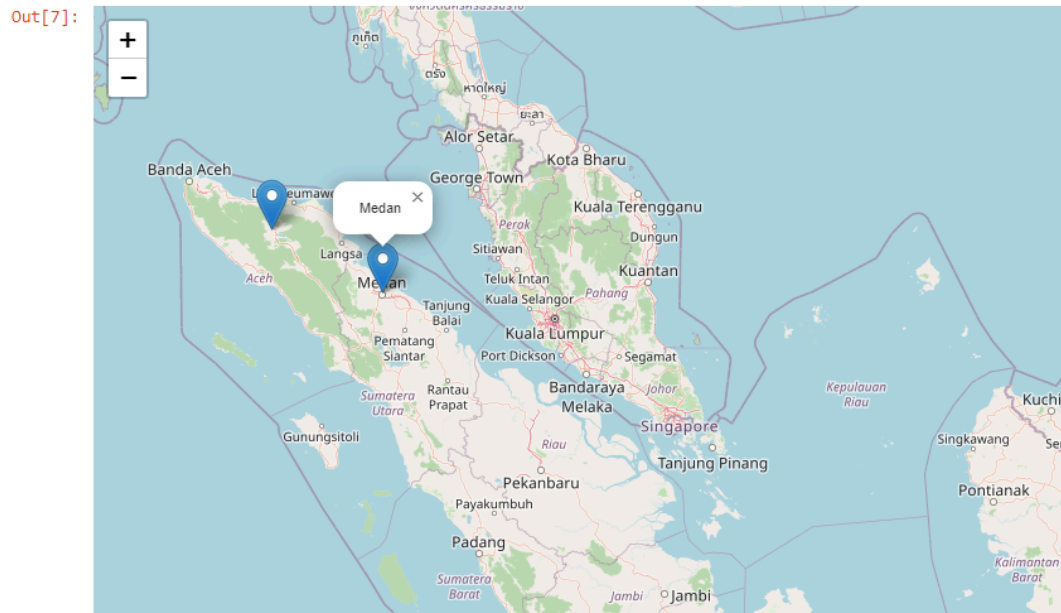
```
In [7]: import folium
import pandas as pd

m = folium.Map(location = [-0.789275, 113.921327],
                zoom_start = 5)

folium.Marker([4.695135, 96.749397],
              popup = 'Aceh').add_to(m)

folium.Marker([3.597031, 98.678513],
              popup = 'Medan').add_to(m)

m
```



**Gambar 9.2**

```

In [10]: import folium
import pandas as pd

m = folium.Map(location = [-0.789275, 113.921327],
                zoom_start = 5)

folium.Marker([4.695135, 96.749397],
              popup = 'Aceh').add_to(m)

folium.Marker([3.597031, 98.678513],
              popup = 'Medan').add_to(m)

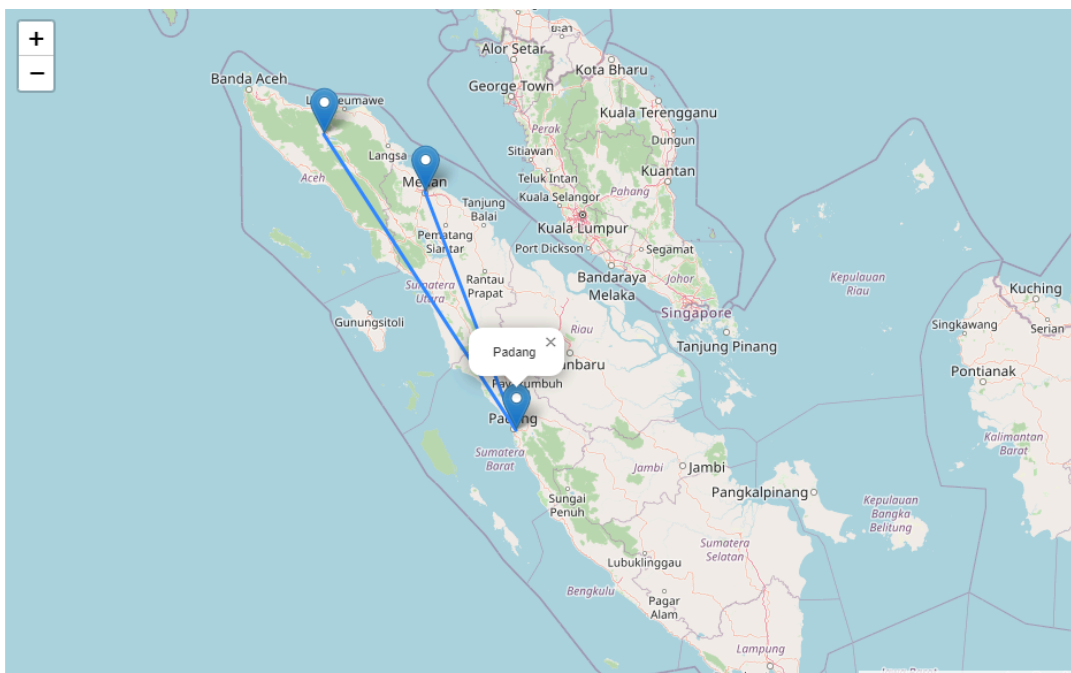
folium.Marker([-0.94708, 100.41718],
              popup = 'Padang').add_to(m)

folium.PolyLine(locations = [(4.695135, 96.749397), (-0.94708, 100.41718)],
                 line_opacity = 0.5).add_to(m)

folium.PolyLine(locations = [(3.597031, 98.678513), (-0.94708, 100.41718)],
                 line_opacity = 0.5).add_to(m)

m

```



**Gambar 9.3**

## BAB 8 MEMBUAT PETA


---

### 8.1 Latihan 1: Membuat Peta Dunia

Gambar 8.1 diperlihatkan gambar peta dunia. Pada Gambar 8.1, untuk membuat peta, digunakan Pustaka atau *library* `ipyleaflet`.

```
In [181]: from ipyleaflet import Map
          lat, lon = 18, 95
          m = Map(center=(lat,lon), zoom=3)
          m.layout.height="550px"
          m

Out[181]:
```

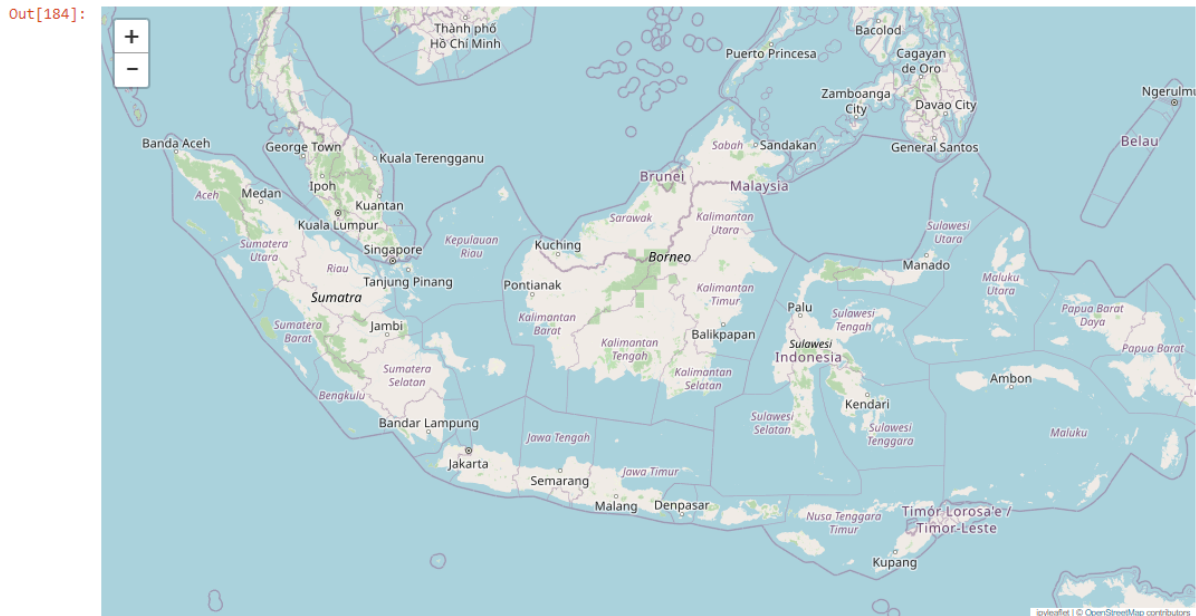


**Gambar 8.1**

### 8.2 Latihan 2: Longitude & Latitude Indonesia

Gambar 8.2 diperlihatkan gambar peta Indonesia. Pada Gambar 8.2, nilai *latitude* adalah `-0.789275`, dan nilai *longitude* adalah `113.921327`, untuk menunjukkan wilayah Indonesia.

```
In [184]: from ipyleaflet import Map
lat, lon = -0.789275, 113.921327
m = Map(center=(lat,lon), zoom=5)
m.layout.height="550px"
m
```



**Gambar 8.2**

### 8.3 Latihan 3: Membuat Marker pada Peta

Gambar 8.3 diperlihatkan peta yang menampilkan *marker* pada wilayah Aceh, Medan dan Padang. Perintah Python

```
df = pd.read_excel("D:\PROJECT BUKU\FIX BUKU 9 APRIL 2023\BUKU
VISUALISASI DATA DENGAN PYTHON\data long lat.xlsx")
```

bertujuan membaca data *longitude* dan *latitude* untuk wilayah Aceh, Medan dan Padang. Selanjut perintah Python

```
from ipyleaflet import Map
from ipyleaflet import Marker
from ipyleaflet import basemaps
```

berarti akan digunakan fungsi **Map**, **Marker** dan **basemaps** pada Pustaka atau *library* **ipyleaflet**.

```
In [194]: import pandas as pd

df = pd.read_excel("D:\\PROJECT BUKU\\FIX BUKU 9 APRIL 2023\\BUKU VISUALISASI DATA DENGAN PYTHON\\data long lat.xlsx")

dataku = pd.DataFrame(df)

dataku
```

```
Out[194]:
```

	Wilayah	Latitude	Longitude
0	Aceh	4.695135	96.749397
1	Medan	3.597031	98.678513
2	Padang	-0.947083	100.417181

```
In [196]: from ipyleaflet import Map

from ipyleaflet import Marker

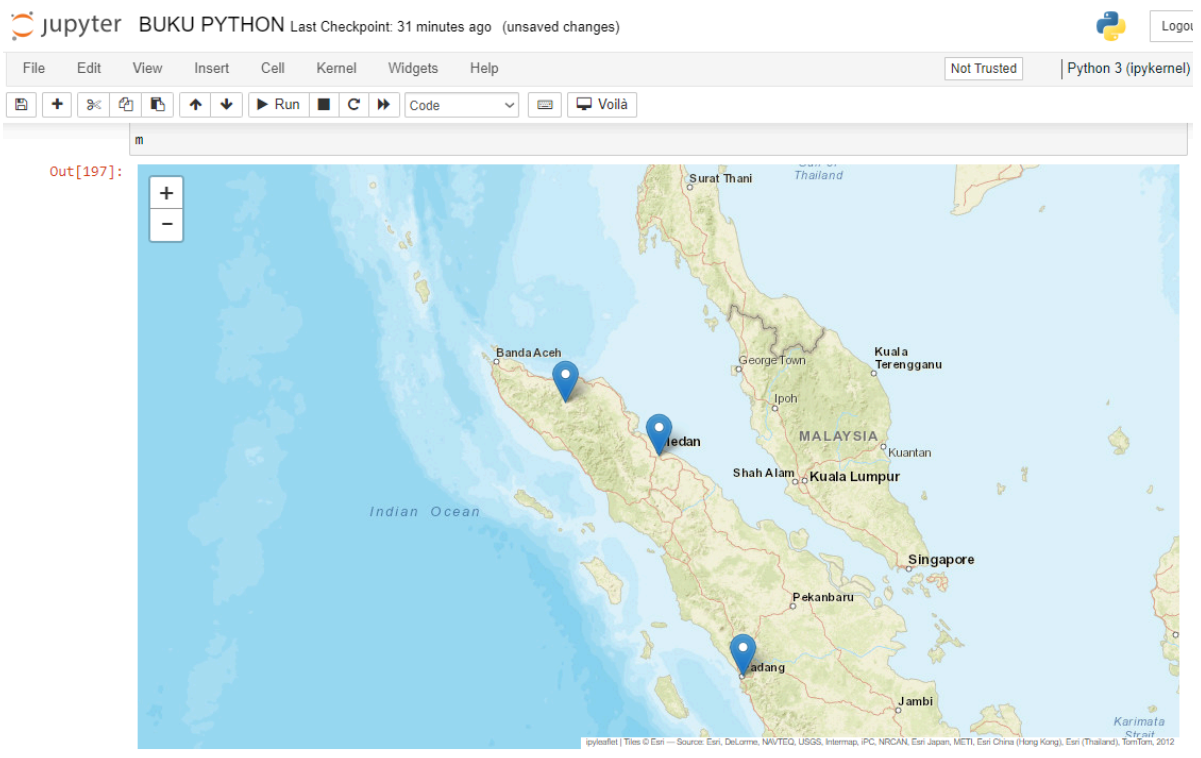
from ipyleaflet import basemaps

m = Map(center=(3.597031, 98.678513), zoom=6, basemap = basemaps.Esri.WorldStreetMap)

for name, lon, lat in dataku[["Wilayah", "Longitude", "Latitude"]].values:
    marker = Marker(location=(lat, lon), draggable=False, title=name, alt=name)
    m.add_layer(marker)

m.layout.height="550px"

m
```



Gambar 8.3

## **BAB 9 SEKILAS MEMBUAT APLIKASI BERBASIS WEBSITE DENGAN STREAMLIT**

---

### **4.1 Latihan 1: Membuat Grafik Batang Frekuensi dengan Plotly**

Diberikan data seperti pada Gambar 4.1. Berdasarkan data pada Gambar 4.1, ingin disajikan grafik batang frekuensi berdasarkan jenis kelamin, seperti pada Gambar 4.2.

## Referensi Bacaan