TAKE HOME TASK

UTS PENGANTAR SAINS DATA

MINI PROJECT

Mhd Iqbal Pratama – 105220042

PROGRAM STUDI ILMU KOMPUTER

FAKULTAS SAINS DAN ILMU KOMPUTER

UNIVERSITAS PERTAMINA

2022

1. Pendahuluan

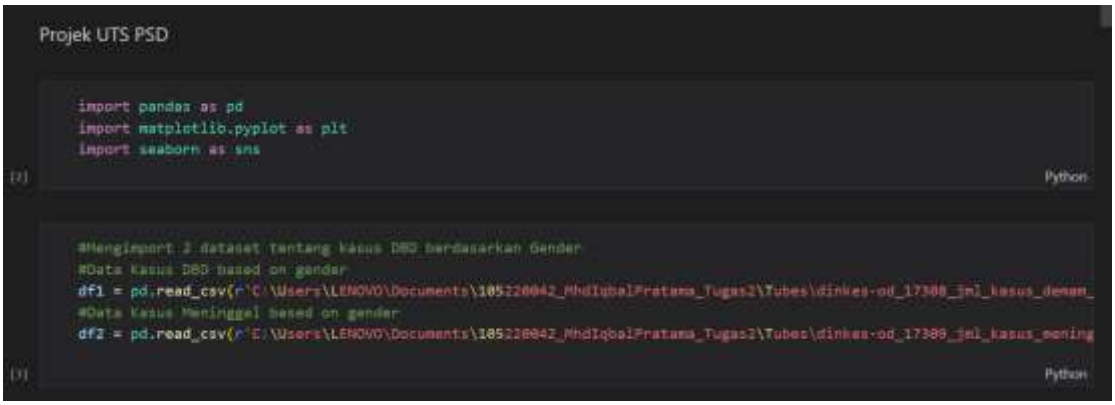
Open data jabar adalah suatu open source web yang dibuat oleh pemerintahan Jawa Barat sebagai bentuk komitmen untuk mewujudkan pemerintahan yang transparan. Pada mini projek, mahasiswa diminta mencari dataset yang tersedia pada web open data jabar. Setelah mendapatkan dataset, mahasiswa diharuskan melakukan visualisasi data untuk menemukan insight dan melakukan beberapa teknik prosesing pada dataset tersebut. Pada mini projek ini, penulis menggunakan 2 buah dataset yang diambil dari open data jabar. Setelah mengambil 2 buah dataset, penulis melakukan merge dataset. Dimana kedua dataset tersebut di gabungkan menjadi satu dataset.

1.1.Daftar Dataset

No	Judul Data	Penyedia/Sumber	Tahun	Tautan
1.	Jumlah Kasus Demam Berdarah Dengue (DBD) Berdasarkan Jenis Kelamin di Jawa Barat	Dinas Kesehatan	2014 - 2020	Tautan1
2.	Jumlah Kasus Meninggal Demam Berdarah Dengue (DBD) Berdasarkan Jenis Kelamin di Jawa Barat	Dinas Kesehatan	2014 - 2020	Tautan2

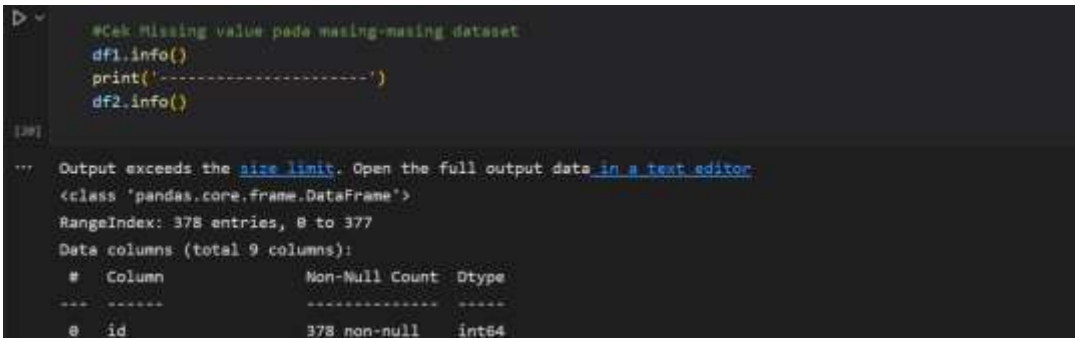
2. Pembahasan

2.1. Import Library dan Dataset



Pada step awal, penulis mengimport library dan dataset yang akan digunakan. Library pandas digunakan untuk membaca dataset, matplotlib digunakan untuk membuat plot, begitu juga dengan seaborn. Variabel df1 dan df2 digunakan untuk menampung dataset yang telah dibaca.

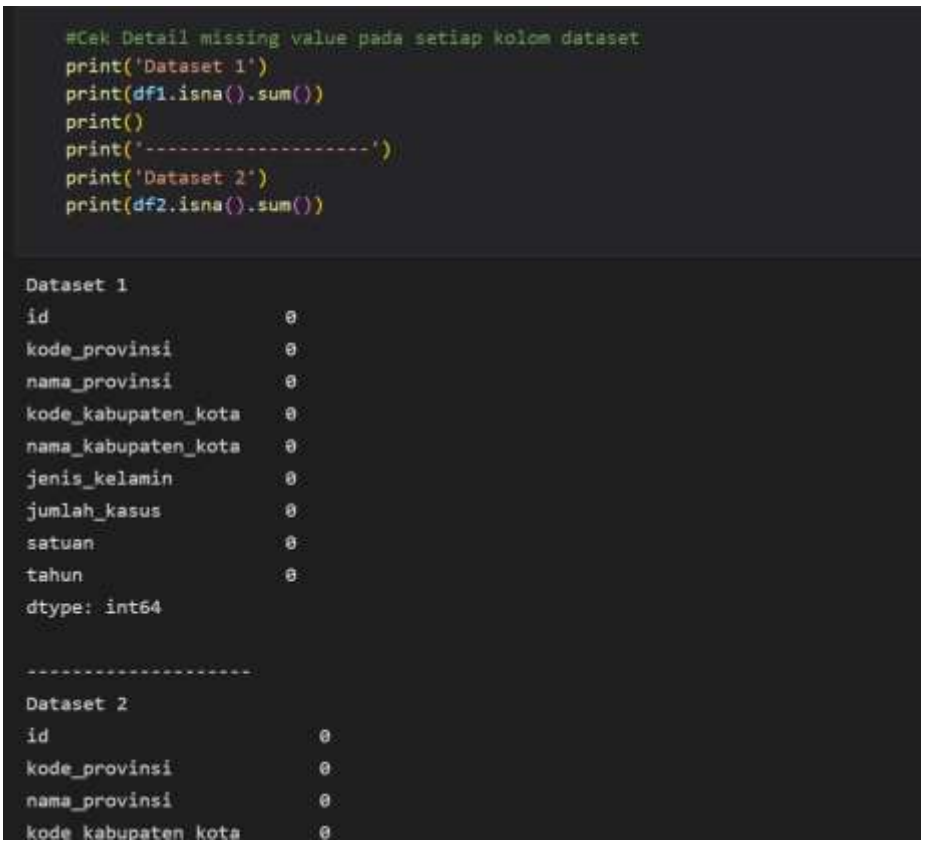
2.2. Mengecek Missing Value di kedua dataset



```
#Cek missing value pada masing-masing dataset
df1.info()
print('----------------------')
df2.info()
```

```
Output exceeds the size limit. Open the full output data in a text editor
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 378 entries, 0 to 377
Data columns (total 9 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   id                  378 non-null    int64
```

Agar lebih meyakinkan kita dapat mengecek null value pada masing-masing kolom dataset.



```
#Cek Detail missing value pada setiap kolom dataset
print('Dataset 1')
print(df1.isna().sum())
print()
print('----------------------')
print('Dataset 2')
print(df2.isna().sum())
```

```
Dataset 1
id                    0
kode_provinsi         0
nama_provinsi         0
kode_kabupaten_kota   0
nama_kabupaten_kota   0
jenis_kelamin         0
jumlah_kasus          0
satuan                0
tahun                 0
dtype: int64

----------------------
Dataset 2
id                    0
kode_provinsi         0
nama_provinsi         0
kode_kabupaten_kota   0
```

2.3. Merge Dataset

```
#Menggabungkan 2 Dataset menjadi variabel data
data = pd.merge(df1, df2, on=['id', 'kode_provinsi', 'nama_provinsi', 'kode_kabupaten_kota', 'nama_kabupaten_kota', '
#Menghapus kolom tahun dan satuan yang terulang
data = data.drop(columns=[col for col in data.columns if 'tahun_x' in col])
data = data.drop(columns=[col for col in data.columns if 'satuan_x' in col])
#Mengganti nama kolom yang satuan_y dan tahun_y menjadi satuan dan tahun
data = data.rename({'satuan_y' : 'satuan'}, axis=1)
data = data.rename({'tahun_y' : 'tahun'}, axis=1)
data
```

Python

	id	kode_provinsi	nama_provinsi	kode_kabupaten_kota	nama_kabupaten_kota	jenis_kelamin	jumlah_kasus	jumlah_kasus
0	1	32	JAWA BARAT	3201	KABUPATEN BOGOR	LAKI-LAKI	915	
1	2	32	JAWA BARAT	3201	KABUPATEN BOGOR	PEREMPUAN	919	
2	3	32	JAWA BARAT	3202	KABUPATEN SUKABUMI	LAKI-LAKI	409	
3	4	32	JAWA BARAT	3202	KABUPATEN SUKABUMI	PEREMPUAN	295	
4	5	32	JAWA BARAT	3203	KABUPATEN CIANJUR	LAKI-LAKI	200	

Dengan fungsi merge pada pandas kita dapat menggabungkan dataset yang similar, dengan menggunakan primary key pada command on=. Lalu drop kolom yang terulang pada dataset. Kemudian rename kolom yang terulang pada dataset.

2.4.Cek Missing Value

```
data.info()
```

Python

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 378 entries, 0 to 377
Data columns (total 10 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   id                    378 non-null    int64
 1   kode_provinsi         378 non-null    int64
 2   nama_provinsi         378 non-null    object
 3   kode_kabupaten_kota   378 non-null    int64
 4   nama_kabupaten_kota   378 non-null    object
 5   jenis_kelamin         378 non-null    object
 6   jumlah_kasus          378 non-null    int64
 7   jumlah_kasus_meninggal 378 non-null   int64
 8   satuan                378 non-null    object
 9   tahun                 378 non-null    int64
dtypes: int64(6), object(4)
memory usage: 32.5+ KB
```

Pada dataset baru, lakukan pengecekan missing value terlebih dahulu sebelum mencari insight. Untuk memastikan kita dapat mengecek null value setiap kolom.

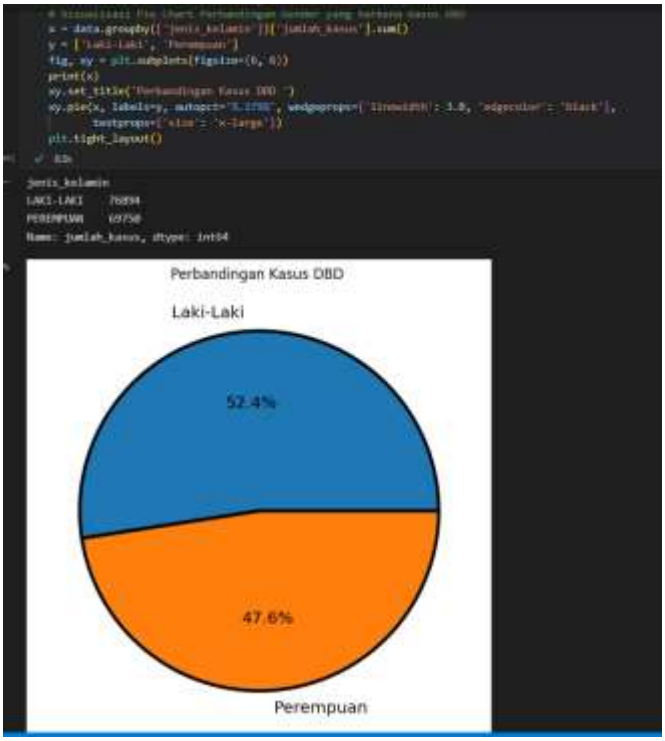
```
print(data.isna().sum())
```

```
id                    0
kode_provinsi         0
nama_provinsi         0
kode_kabupaten_kota   0
nama_kabupaten_kota   0
jenis_kelamin         0
jumlah_kasus          0
jumlah_kasus_meninggal 0
satuan                0
tahun                 0
dtype: int64
```

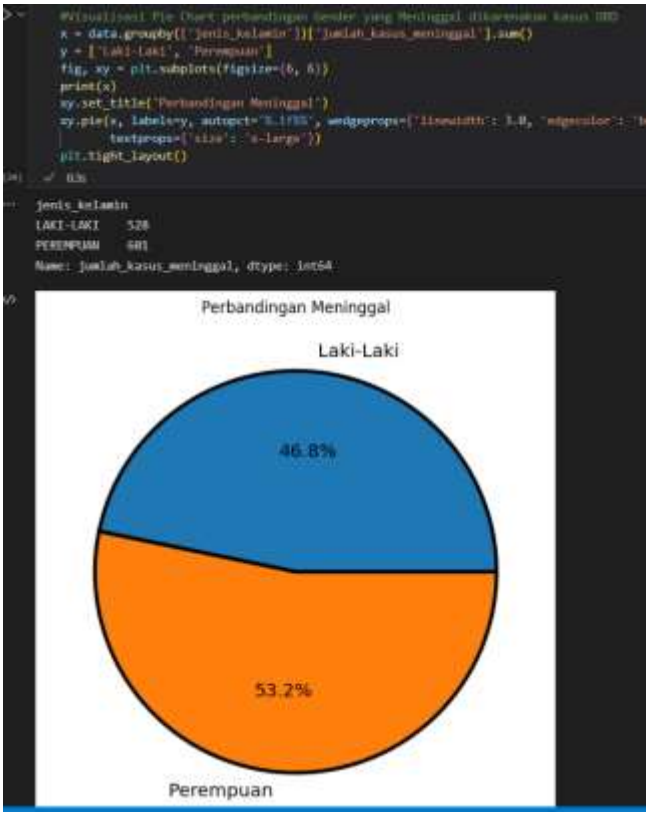
2.5. Visualiasi Data

Didapatkan beberapa insight dari hasil visualiasi data. Insight yang didapat antara lain :

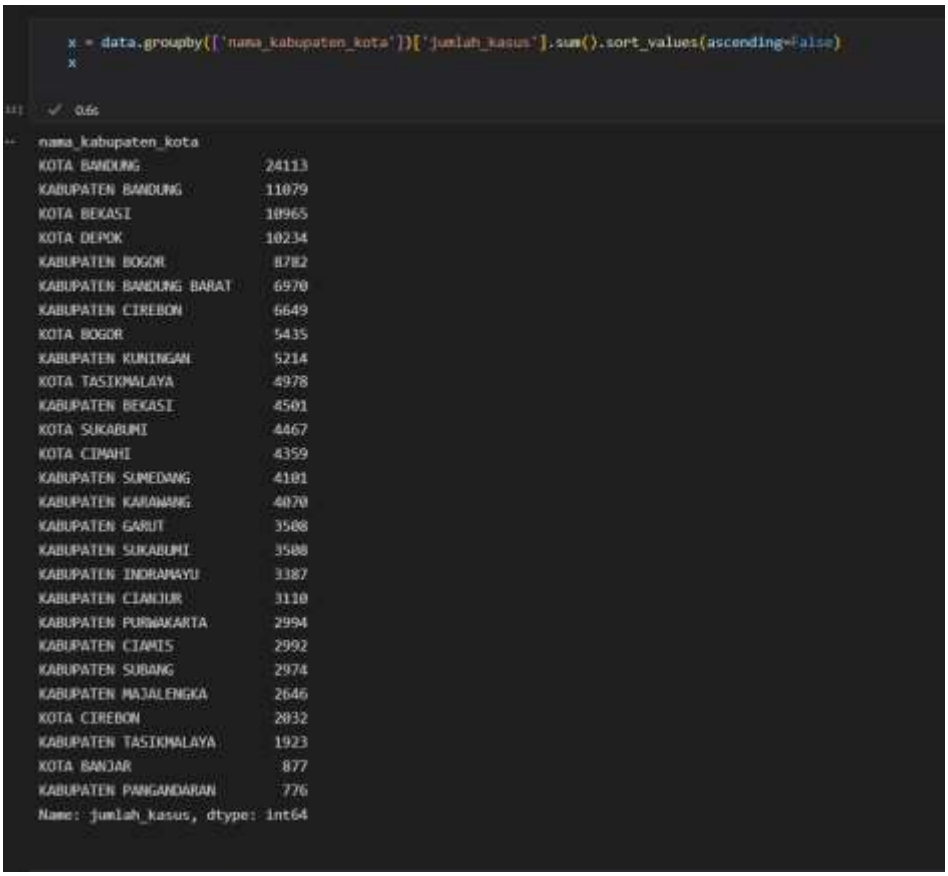
1. Perbandingan gender yang terkena kasus DBD.



2. Perbandingan gender yang meninggal dikarenakan kasus DBD.



3. Kota dengan kasus DBD terbanyak.



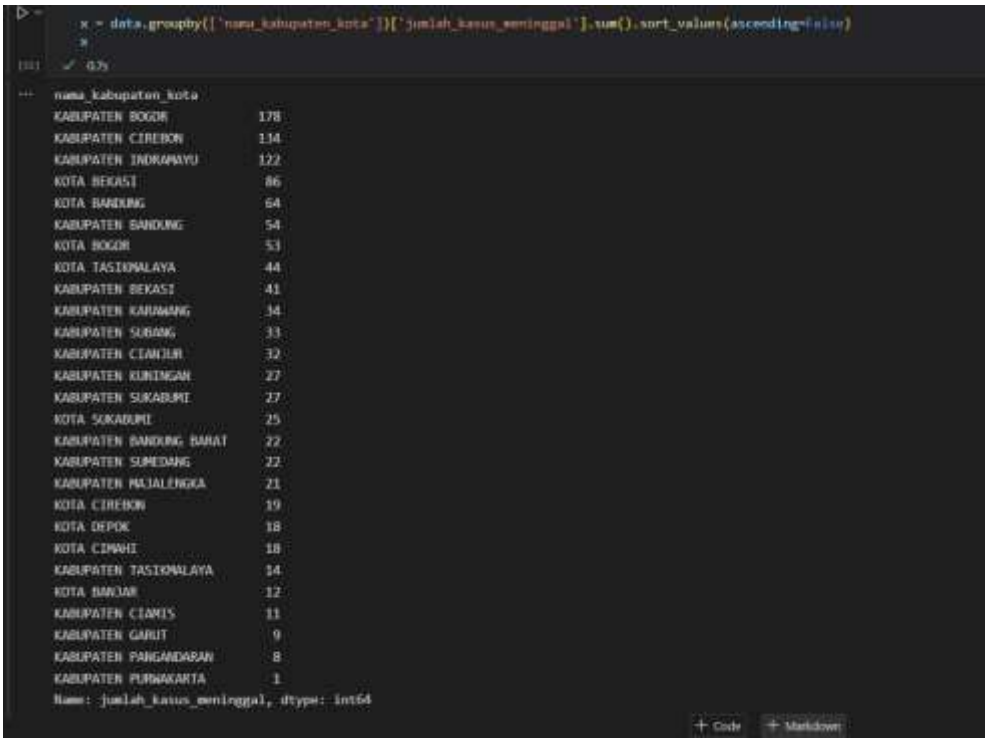
```
x = data.groupby(['nama_kabupaten_kota'])['jumlah_kasus'].sum().sort_values(ascending=False)
x
```

✓ 0.6s

```
nama_kabupaten_kota
KOTA BANDUNG              24113
KABUPATEN BANDUNG         11079
KOTA BEKASI               10965
KOTA DEPOK                10234
KABUPATEN BOGOR            8782
KABUPATEN BANDUNG BARAT    6970
KABUPATEN CIREBON          6649
KOTA BOGOR                 5435
KABUPATEN KUNINGAN         5214
KOTA TASIKMALAYA           4978
KABUPATEN BEKASI           4501
KOTA SUKABUMI              4467
KOTA CIMAHI                4359
KABUPATEN SUMEDANG         4101
KABUPATEN KARAWANG         4070
KABUPATEN GARUT            3508
KABUPATEN SUKABUMI         3508
KABUPATEN INDRAMAYU        3387
KABUPATEN CIANJUR          3110
KABUPATEN PURWAKARTA       2994
KABUPATEN CIAMIS           2992
KABUPATEN SUBANG           2974
KABUPATEN MAJALENGKA       2646
KOTA CIREBON               2032
KABUPATEN TASIKMALAYA      1923
KOTA BANJAR                 877
KABUPATEN PANGANDARAN       776
Name: jumlah_kasus, dtype: int64
```

Dengen menggunakan key sorting, maka kita bisa mendapatkan insight.

4. Kota dengan kasus meninggal dikarenakan DBD terbanyak.



```
x = data.groupby(['nama_kabupaten_kota'])['jumlah_kasus_meninggal'].sum().sort_values(ascending=False)
x
```

✓ 0.7s

```
nama_kabupaten_kota
KABUPATEN BOGOR           178
KABUPATEN CIREBON         114
KABUPATEN INDRAMAYU       122
KOTA BEKASI                86
KOTA BANDUNG               64
KABUPATEN BANDUNG          54
KOTA BOGOR                 53
KOTA TASIKMALAYA           44
KABUPATEN BEKASI           41
KABUPATEN KARAWANG         34
KABUPATEN SUBANG           33
KABUPATEN CIANJUR          32
KABUPATEN KUNINGAN         27
KABUPATEN SUKABUMI         27
KOTA SUKABUMI              25
KABUPATEN BANDUNG BARAT    22
KABUPATEN SUMEDANG         22
KABUPATEN MAJALENGKA       21
KOTA CIREBON               19
KOTA DEPOK                 18
KOTA CIMAHI                18
KABUPATEN TASIKMALAYA      14
KOTA BANJAR                12
KABUPATEN CIAMIS           11
KABUPATEN GARUT             9
KABUPATEN PANGANDARAN       8
KABUPATEN PURWAKARTA        1
Name: jumlah_kasus_meninggal, dtype: int64
```

+ Code + Markdown

Dengen menggunakan key sorting, maka kita bisa mendapatkan insight.

2.6. Prosesing Data

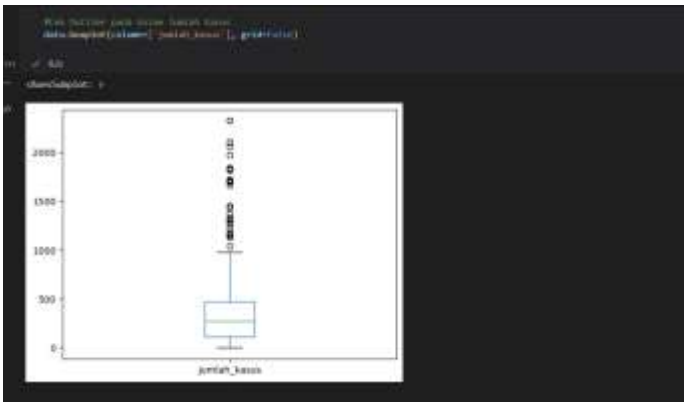
2.6.1. Cek missing value

```
#Cek missing value
data.info()
```

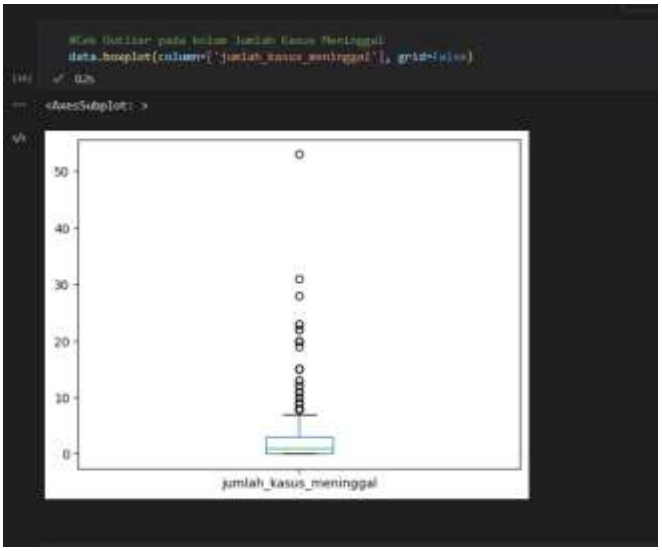
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 378 entries, 0 to 377
Data columns (total 10 columns):
 #   Column                 Non-Null Count  Dtype 
---  ------                 --------------  ----- 
 0   id                     378 non-null    int64 
 1   kode_provinsi          378 non-null    int64 
 2   nama_provinsi          378 non-null    object
 3   kode_kabupaten_kota    378 non-null    int64 
 4   nama_kabupaten_kota    378 non-null    object
 5   jenis_kelamin          378 non-null    object
 6   jumlah_kasus           378 non-null    int64 
 7   jumlah_kasus_meninggal 378 non-null    int64 
 8   satuan                 378 non-null    object
 9   tahun                  378 non-null    int64 
dtypes: int64(6), object(4)
memory usage: 32.5+ KB
```

Seperti yang sudah dilakukan setelah merge dataset, tidak ditemukan adanya missing value.

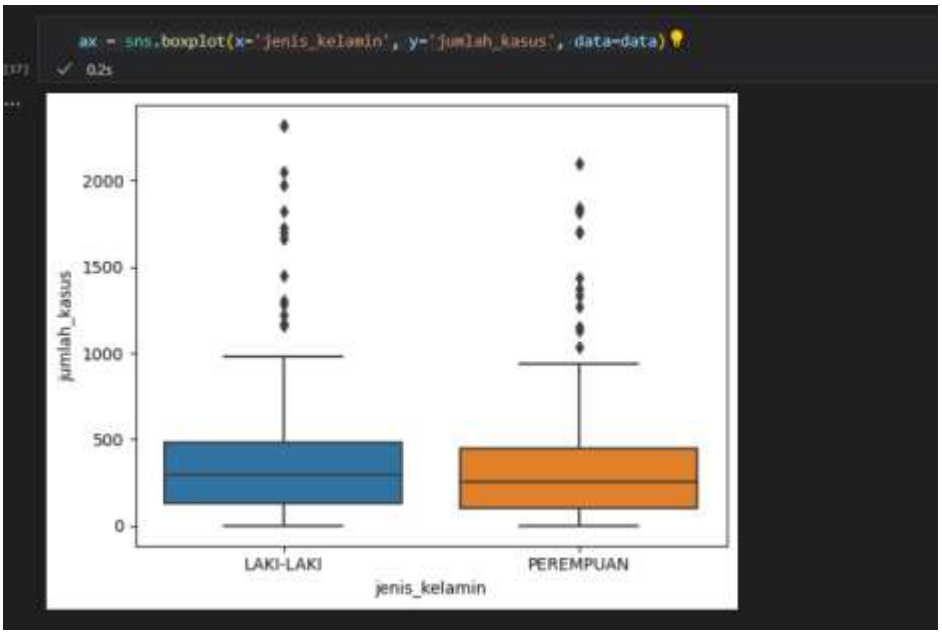
2.6.2. Outlier



Pada kolom jumlah_kasus, dapat diliihat adanya outlier pada persebaran datanya. Begitu pula pada kolom jumlah_kasus_meninggal dibawah ini.



Outlier pada tiap gender yang terkena kasus DBD.



Outlier pada tiap gender yang meninggal dikarenakan kasus DBD.