



Presented by M. Taufiq Hariadi  
Binar Academy – Wave 7

# EXPLORATION OF TWEET FROM INDONESIAN NETIZEN

API FOR TEXT CLEANSING AND DATA ANALYSIS REPORTS

- Data Preprocessing
- Data Analysis
- Data Visualization
- RESTFUL API

## LATAR BELAKANG

Intro

Apakah kamu salah satu orang yang menggunakan media sosial atau biasa disebut dengan 'Netizen' khusus nya pada platform sosial media Twitter? Jika iya, maka kamu adalah salah satu dari 24 Juta 'Netizen' Indonesia (Sumber: katadata.co.id).

Pada Penelitian ini saya akan melakukan eksplorasi data tweet netizen Indonesia, Sebagai seorang data scientist saya menggunakan bahasa pemrograman seperti Python dan Query SQL serta Teknologi seperti Jupyter Notebook, Visual Studio Code. Adapun yang akan dilakukan diantaranya Data Cleansing, Descriptive Analysis, dan Data Visualization.



## RUMUSAN MASALAH

1. Metode apa yang digunakan dalam proses data cleansing, dan data apa yang di bersihkan?
2. Berapa Nilai Mean, Median, Modus, Kuartil dan Interquartile Range, Outliers, Variance, Standard Deviation, Skewness dan kurtois dari Tweet yang telah di bersihkan pada dataframe?
3. Kata apa yang paling sering muncul pada tweet yang telah di bersihkan?
4. Ada berapa banyak 'kata\_alay' dan 'kata\_kasar' pada dataframe?
5. Berapa jumlah karakter dan kata pada tweet yang telah di kelompokkan sebagai 'kata\_alay' dan 'kata\_kasar'?



## TUJUAN PENELITIAN

1. Melakukan Data cleansing tweet netizen dari file data.csv
2. Menampilkan nilai Mean, Median, Modus, Kuartil dan Interquartile Range, Outliers, Variance, Standard Deviation, Skewness dan kurtois dari Tweet yang telah di bersihkan pada dataframe
3. Melihat Kata apa yang paling sering muncul pada tweet yang telah di bersihkan
4. Melihat berapa banyak jumlah 'kata\_alay' dan 'kata\_kasar' pada dataframe
5. Melihat berapa jumlah karakter dan kata pada tweet yang telah di kelompokkan sebagai 'kata\_alay' dan 'kata\_kasar'



## LIST DATA

"data.csv"

Data ini berisi Tweet yang terdapat kolom dengan berbagai kategori tweet, yaitu ujaran kebencian atau Hatespeech (HS), kata kasar (Abusive), dan group Hatespeech lainnya. '0' menunjukkan 'Tidak' dan '1' menunjukkan 'Iya'. Data pada kolom 'Tweet' ini akan dibersihkan.

"Abusive.csv"

Data ini berisi kata-kata kasar 'Abusive'. Data ini akan menjadi kamus untuk melakukan pengelompokan dari tweet netizen. Apakah Termasuk 'Abusive' atau 'Non Abusive'

"data.csv"

Data ini berisi kalimat alay dan kalimat yang normal. Data ini akan dijadikan kamus untuk mengganti kalimat dari tweet netizen pada 'data.csv' yang tergolong kalimat 'alay' menjadi kalimat yang normal.

# METODE PENELITIAN

Data Preprocessing  
(Data Cleansing)  
menggunakan Pandas,  
RegEx & NLTK

Data Analysis  
menggunakan  
Deskriptif Statistik  
(Univariate & Bivariate)

Data Visualization  
menggunakan Pandas,  
Matplotlib & Word  
Cloud

## Tools:



## Library:



## Rest API:



## IMPOR DATA & LIHAT INFO DATA

Impor library dan impor file 'data.csv' sebagai dataframe (df)

```
# Impor Library
import pandas as pd
import numpy as np
import re
import matplotlib as plt
import seaborn as sns
pd.options.display.max_colwidth = 2000
✓ 1.5s
```

```
# impor file "data.csv" kedalam dataframe (df)
df = pd.read_csv('data.csv', encoding='latin-1')
✓ 0.1s
```

# Menampilkan 5 dataframe teratas  
df.head()

✓ 0.0s

	Tweet	HS	Abusive	HS_Individual	HS_Group	HS_Religion	HS_Race	HS_Physical	HS_Gender	HS_Other	HS_Weak	HS_Moderate	HS_Strong
0	- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego !!!'	1	1	1	0	0	0	0	0	1	1	0	0
1	RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'	0	1	0	0	0	0	0	0	0	0	0	0
2	41. Kadang aku berfikir, kenapa aku tetap percaya pada Tuhan padahal aku selalu jatuh berkali-kali. Kadang aku merasa Tuhan itu ninggalkan aku sendirian. Ketika orangtuaku berencana berpisah, ketika kakakku lebih memilih jadi Kristen. Ketika aku anak ter	0	0	0	0	0	0	0	0	0	0	0	0
3	USER USER AKU ITU AKU\nKU TAU MATAMU SIPIT TAPI DILIAH DARI MANA ITU AKU'	0	0	0	0	0	0	0	0	0	0	0	0
4	USER USER Kaum cebong kapir udah keliatan dongoknya dari awal tambah dongok lagi hahahah'	1	1	0	1	1	0	0	0	0	0	1	0



## IMPOR DATA & LIHAT INFO DATA

### Menampilkan Info Dataframe

```
# Menampilkan info dari dataframe  
df.info()  
✓ 0.0s  
  
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 13169 entries, 0 to 13168  
Data columns (total 13 columns):  
 # Column Non-Null Count Dtype  
---  
 0 Tweet 13169 non-null object  
 1 HS 13169 non-null int64  
 2 Abusive 13169 non-null int64  
 3 HS_Individual 13169 non-null int64  
 4 HS_Group 13169 non-null int64  
 5 HS_Religion 13169 non-null int64  
 6 HS_Race 13169 non-null int64  
 7 HS_Physical 13169 non-null int64  
 8 HS_Gender 13169 non-null int64  
 9 HS_Other 13169 non-null int64  
 10 HS_Weak 13169 non-null int64  
 11 HS_Moderate 13169 non-null int64  
 12 HS_Strong 13169 non-null int64  
dtypes: int64(12), object(1)
```

df.info untuk melihat Nama Kolom Tipe Data dari dataframe

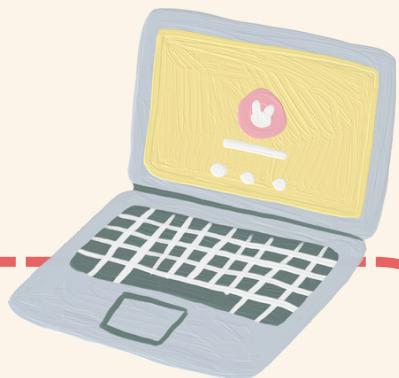
```
# Melihat jumlah value yang kosong  
df.isna().sum()  
✓ 0.0s  
  
Tweet 0  
HS 0  
Abusive 0  
HS_Individual 0  
HS_Group 0  
HS_Religion 0  
HS_Race 0  
HS_Physical 0  
HS_Gender 0  
HS_Other 0  
HS_Weak 0  
HS_Moderate 0  
HS_Strong 0  
dtype: int64
```

df.isna().sum() untuk melihat jumlah value yang kosong dari dataframe

```
df.duplicated().sum()  
✓ 0.0s  
125  
  
df = df.drop_duplicates()  
✓ 0.0s  
  
df.duplicated().sum()  
✓ 0.0s  
0
```

df.duplicated().sum() untuk melihat jumlah value yang duplikat dari dataframe.

df.drop\_duplicates() untuk menghapus data duplikat

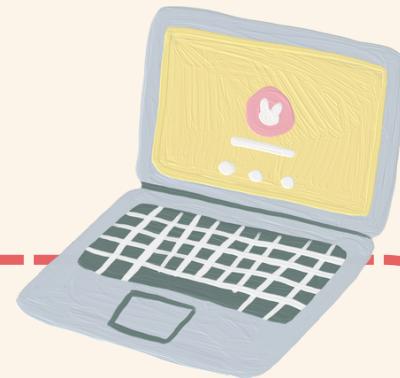


## IMPOR DATA & LIHAT INFO DATA

Impor file 'new\_kamusalay.csv' sebagai dataframe (df\_kamus)

```
# impor file new_kamusAlay.csv kedalam dataframe (df_kamus)
df_kamus = pd.read_csv('new_kamusalay.csv', encoding='latin-1', names=['Alay', 'Normal'])

✓ 0.0s
```



Melihat nama kolom dan tipe data : Melihat list data 5 teratas

```
• df_kamus.info()

✓ 0.0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15167 entries, 0 to 15166
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
--- 
 0   Alay     15167 non-null   object 
 1   Normal   15167 non-null   object 
dtypes: object(2)
memory usage: 237.1+ KB
```

```
df_kamus.head()

✓ 0.0s
```

	Alay	Normal
0	anakjakartaasikasik	anak jakarta asyik asyik
1	pakcikdahtua	pak cik sudah tua
2	pakcikmudalagi	pak cik muda lagi
3	t3tapjokowi	tetap jokowi
4	3x	tiga kali

Melihat value yang kosong

```
df_kamus.isna().sum()

✓ 0.0s

Alay      0
Normal    0
dtype: int64
```

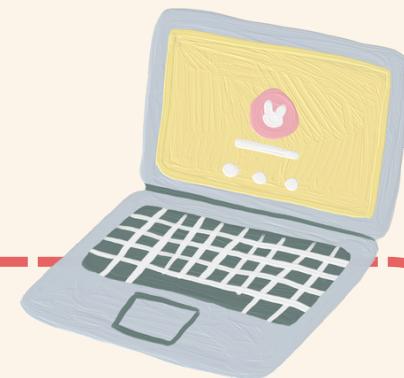
## IMPOR DATA & LIHAT INFO DATA

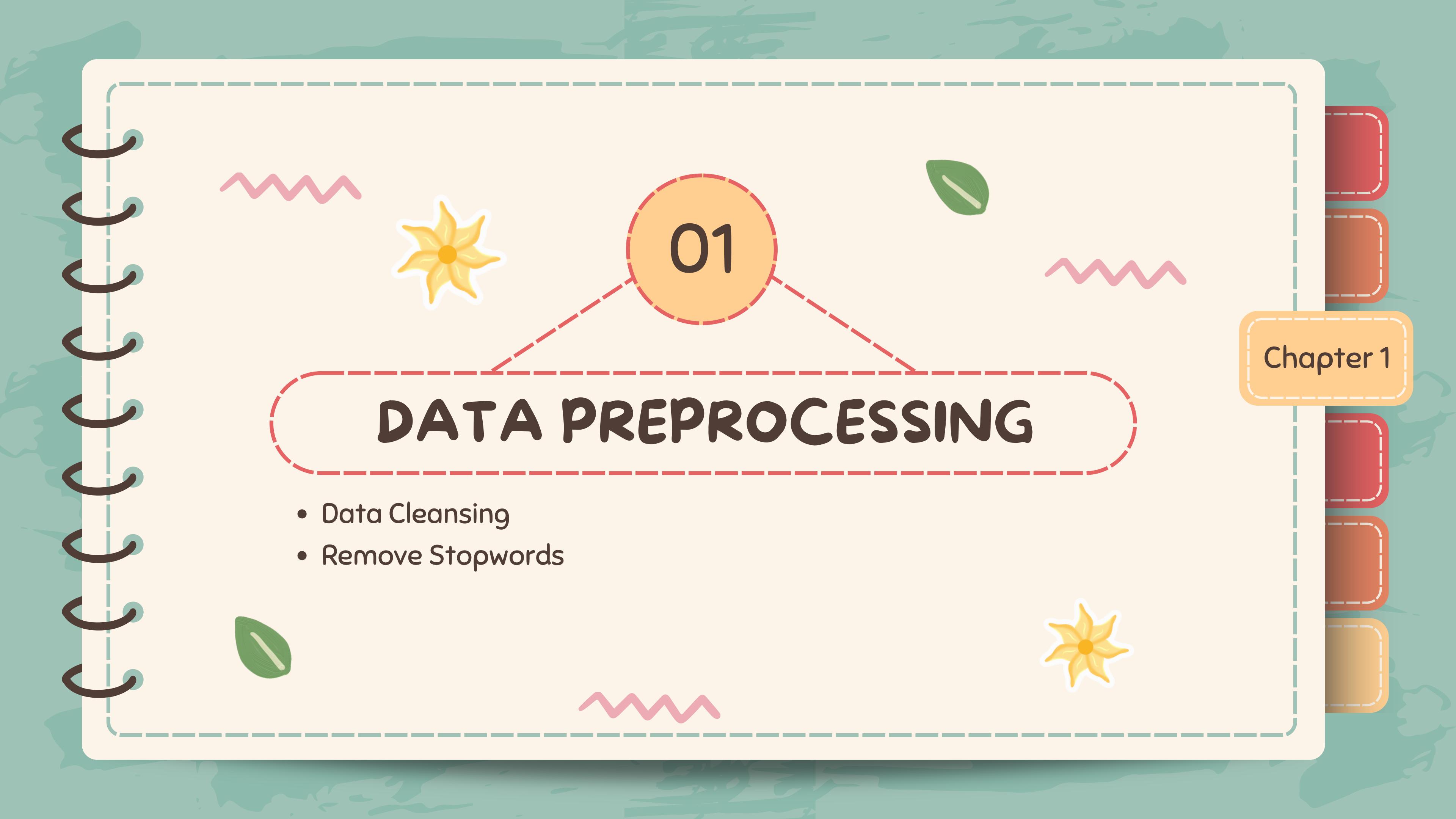
Impor file 'abusive.csv' sebagai dataframe (df\_abusive)

```
# impor file abusive.csv kedalam dataframe (df_kamus)
df_abusive = pd.read_csv('abusive.csv', encoding='latin-1')
df_abusive['label'] = 'Abusive'
df_abusive
✓ 0.1s
```

	ABUSIVE	label
0	alay	Abusive
1	ampas	Abusive
2	buta	Abusive
3	keparat	Abusive
4	anjing	Abusive
...	...	...
120	rezim	Abusive
121	sange	Abusive
122	serbet	Abusive
123	sipit	Abusive
124	transgender	Abusive

125 rows × 2 columns





01

# DATA PREPROCESSING

- Data Cleansing
- Remove Stopwords

Chapter 1

## DATA PREPROCESSING

Data Preprocessing adalah teknik yang digunakan untuk mengubah data mentah menjadi data yang sesuai dengan kebutuhan yang dapat berguna dan juga efisien untuk dianalisis, sehingga keputusan bisnis menjadi lebih tepat.

Adapun library yang digunakan dalam melakukan data preprocessing ini antara lain:

- 1.RegEx
- 2.Pandas
- 3.NLTK

Saya akan mengganti setiap kata yang terdapat pada dataframe 'df' kolom 'Tweet' dengan kata yang terdapat pada dataframe 'df\_kamus' kolom 'Normal'



## DATA PREPROCESSING

Pandas merupakan library yang digunakan untuk memproses data yang meliputi data cleansing, data manipulation dan data analysis.

Regular Expression (RegEx) merupakan library yang digunakan untuk mengolah data teks. Kita dapat mencari, mengubah dan memisahkan setiap karakter dari sebuah data teks.

Saya akan menggunakan RegEx untuk mengolah data teks, dan Pandas untuk melakukan data Cleansing.

# DATA CLEANSING | .[RegEx]\*

```
def cleansing(text):
    text = text.lower()
    pola_1 = r'#[^\s]+'
    text = re.sub(pola_1, '', text)

    pola_2 = r'@[^\s]+'
    text = re.sub(pola_2, '', text)

    pola_3 = r'(user|retweet|\t|\r|url|xd|orang|kalo)'
    text = re.sub(pola_3, '', text)

    pola_4 = r'\b\w{1,3}\b'
    text = re.sub(pola_4, '', text)

    pola_5 = r'[\,@\*\_\-\!\!;?\.\"]\((\{\}\<\>+\%\$\^#\/\`~\|\&\|)'
    text = re.sub(pola_5, ' ', text)

    pola_6 = r'\\[a-z0-9]{1,5}'
    text = re.sub(pola_6, '', text)

    pola_7 = r'[\x00-\x7f]'
    text = re.sub(pola_7, '', text)
```

Mengubah kalimat menjadi huruf kecil

Menghapus hastag

Menghapus mention

Menghapus user, retweet, \t, \r, url, xd, orang, kalo

Menghapus single character

Menghapus tanda baca, operasi matematika, dll.

Menghapus emoji

Menghapus karakter yang bukan termasuk ASCII

# DATA CLEANSING | .[RegEx]\*

```
pola_8 = r'(https|https:)'  
text = re.sub(pola_8, '', text)
```

Menghapus url yang diawali dengan http atau https

```
pola_9 = r'[\\\n]\\n[''  
text = re.sub(pola_9, '', text)
```

Menghapus karakter '\', '[, ]'

```
pola_10 = r'\bwk\w+'  
text = re.sub(pola_10, '', text)
```

Menghapus 'wkwkwk'

```
pola_11 = r'\d+'  
text = re.sub(pola_11, '', text)
```

Menghapus digit karakter

```
pola_12 = r'(\u0-9A-Fa-f)+'  
text = re.sub(pola_12, '', text)
```

Menghapus karakter yang bukan termasuk ASCII

```
pola_13 = r'(\s+|\n)'  
text = re.sub(pola_13, ' ', text)
```

Menghapus spasi yang berlebih

```
text = text.rstrip()  
text = text.lstrip()  
return text
```

Menghapus spasi pada kalimat pertama dan terakhir

```
def replaceThreeOrMore(text):  
    pattern = re.compile(r"(.)\1{1,}", re.DOTALL)  
    return pattern.sub(r"\1\1", text)
```

```
df['clean_tweet'] = df['Tweet'].apply(cleansing)  
df['clean_tweet'] = df['clean_tweet'].apply(replaceThreeOrMore)  
df[['clean_tweet']]
```

Menerapkan function 'cleansing' & 'replaceThreeOrMore'

Menghapus tiga atau lebih pengulangan karakter termasuk newlines

## DATAFRAME SEBELUM & SESUDAH DI BERSIHKAN

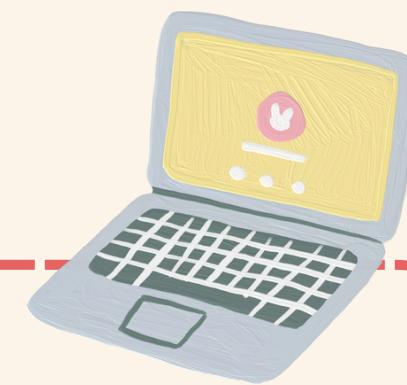
### Menampilkan Hasil dari Data Cleansing

```
df[['Tweet']].head()
✓ 0.1s
Python
Tweet
0 - disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego !!!'
1 RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'
2 41. Kadang aku berfikir, kenapa aku tetap percaya pada Tuhan padahal aku selalu jatuh berkali-kali. Kadang aku merasa Tuhan itu ninggalkan aku sendirian. Ketika orangtuaku berencana berpisah, ketika kakakku lebih memilih jadi Kristen. Ketika aku anak ter
3 USER USER AKU ITU AKU\n\nKU TAU MATAMU SIPIT TAPI DILIAH DARI MANA ITU AKU'
4 USER USER Kaum cebong kapir udah keliatan dongoknya dari awal tambah dongok lagi hahahah'
```

Sebelum dibersihkan menggunakan  
RegEx dan Pandas

```
df[['clean_tweet']]
✓ 0.0s
Python
clean_tweet
0 disaat semua cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego
1 siapa yang telat ngasih edan sarap bergaul dengan cigax jifla calis sama siapa licew juga
2 kadang berfikir kenapa tetap percaya pada tuhan padahal selalu jatuh berkali kali kadang merasa tuhan ninggalkan sendirian ketika tuaku berencana berpisah ketika kakakku lebih memilih jadi kristen ketika anak
3 matamu sipit tapi diliat dari mana
4 kaum cebong kapir udah keliatan dongoknya dari awal tambah dongok lagi hahahah'
```

Setelah dibersihkan menggunakan  
RegEx dan Pandas



## DATAFRAME SEBELUM & SESUDAH DI BERSIHKAN

Replace kata dengan kamus 'new\_kamusalay.csv' kolom 'clean\_tweet'

```
dictionary = dict(zip(df_kamus['anakjakartaasikasik'], df_kamus['anak jakarta asik asyik']))
df['clean_tweet'] = df['clean_tweet'].apply(lambda x: " ".join([dictionary.get(w, w) for w in x.split()]))
```

✓ 0.1s

Python

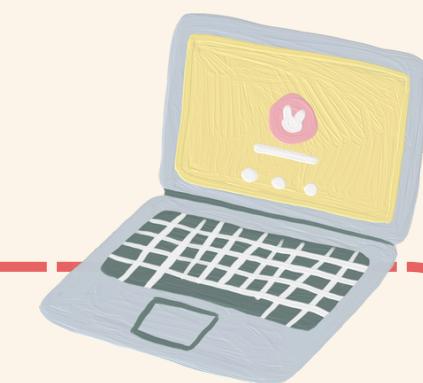
```
df
```

✓ 0.0s

Python

	Tweet	HS	Abusive	HS_Individual	HS_Group	HS_Religion	HS_Race	HS_Physical	HS_Gender	HS_Other	HS_Weak	HS_Moderate	HS_Strong	clean_tweet
0	- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego !! !'	1	1	1	0	0	0	0	0	1	1	0	0	di saat semua cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego
1	RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'	0	1	0	0	0	0	0	0	0	0	0	0	siapa yang telat memberi edan sarap bergaul dengan cigax jifla calis sama siapa licew juga

1. Gabungkan kedua kolom 'tweet' dari datafrane 'df\_kamus' kedalam dictionary dan kolom 'Alay' sebagai key, dan kolom 'Normal' sebagai value.
2. Kemudian kata dari kolom 'Tweet' yang ada pada kolom 'Alay' akan diganti dengan kata yang ada pada kolom 'Normal'.



## MEMBUKTIKAN KATA ALAY YANG DIGANTIKAN OLEH KATA NORMAL



```
df[df['clean_tweet'].str.contains("menteri hukum dan hak asasi manusia")]  
✓ 0.1s
```

	Tweet	clean_tweet	tweet_tokenized	clean_tweet_without_stopwords	total_char_wo_stopwords	total_word_wo_stopwords	label
1373	PRESIDEN DIHINA. Kapolri, Menkumham, Mendikbud, Menhan, Abu Janda, Anshor, Banser. Semua terdiam. Ajaib.'	presiden dihina kepala kepolisian republik indonesia menteri hukum dan hak asasi manusia menteri pendidikan dan kebudayaan menhan janda ansar barisan serba guna semua terdiam ajaib	[presiden, dihina, kepala, kepolisian, republik, indonesia, menteri, hukum, dan, hak, asasi, manusia, menteri, pendidikan, dan, kebudayaan, menhan, janda, ansar, barisan, serba, guna, semua, terdiam, ajaib]	presiden dihina kepala kepolisian republik indonesia menteri hukum hak asasi manusia menteri pendidikan kebudayaan menhan janda ansar barisan serba terdiam ajaib	161	21	Abusive

Membuktikan setiap kata pada 'Tweet' yang terdapat pada df\_kamus['anakjakartaasikasik'] sudah diganti (replaced) oleh kata pada df\_kamus['anak jakarta asyik asyik']

## DATA CLEANSING UNTUK TOKENIZE DAN MENGHAPUS 'STOPWORDS'



Melakukan Data Cleansing menggunakan library NLTK untuk 'TOKENIZE' dan menghapus 'STOPWORDS', adapun definisinya sebagai berikut:

### 1. Tokenize

Pemrosesan data untuk membagi teks yang berupa kalimat, paragraf atau dokumen, menjadi token-token atau bagian-bagian tertentu.

### 2. Stopword

Dalam dunia pemrograman seperti NLP (Natural Language Processing), stopwords merupakan kata yang diabaikan dalam pemrosesan dan biasanya disimpan di dalam stop lists. Stop list ini berisi daftar kata umum yang mempunyai fungsi tapi tidak mempunyai arti. Tujuan utama dalam penerapan proses stopwords ini adalah mengurangi jumlah kata dalam sebuah dataframe yang nantinya akan berpengaruh dalam kecepatan dan performa NLP.

## DATA CLEANSING UNTUK TOKENIZE DAN MENGHAPUS 'STOPWORDS'

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
nltk.download('stopwords')
nltk.corpus.stopwords.words('indonesian')

# tokenize
df['tweet_tokenized'] = df['clean_tweet'].apply(lambda x: word_tokenize(x))

# Melakukan define stopwords yang dipakai
indo_stop_words = stopwords.words("indonesian")
more_stopword = ['dengan', 'ia', 'bahwa', 'oleh', 'udah', 'gitu', 'pake', 'sampe',
'cuma', 'bikin', 'kayak', 'bilang', 'trus', 'mulu', 'haha', 'wkwk', 'emang', 'bener']
indo_stop_words.extend(more_stopword)

# Membuat Kolom 'tweet_without_stopwords' pada dataframe 'df'
df['clean_tweet_without_stopwords'] = df['tweet_tokenized'].apply(lambda x: ' '.join([word for word in x if word not in indo_stop_words]))
```

✓ 3.7s



## HASIL DARI DATA PREPARATION

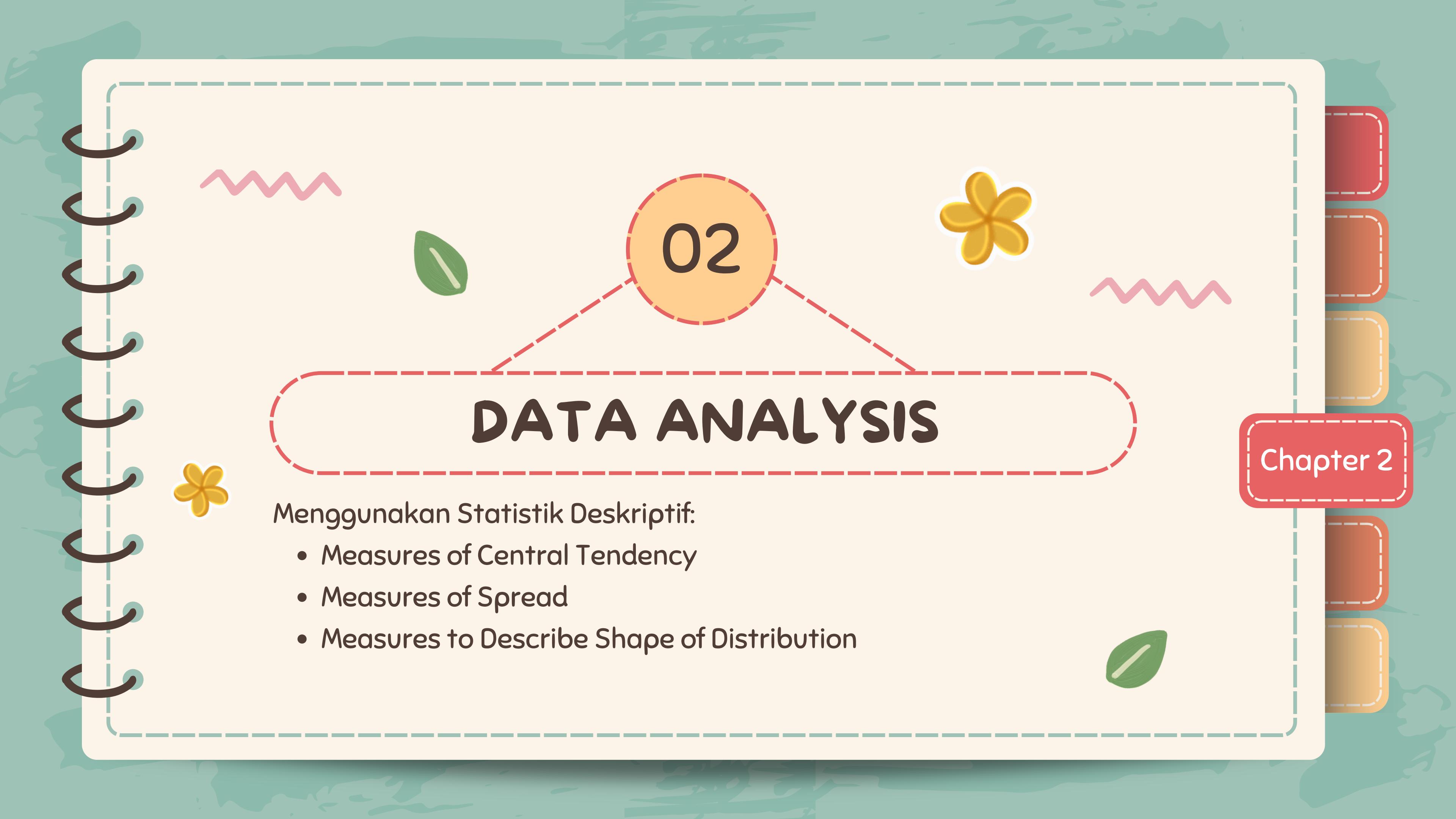
```
df_abusive['label'] = 'Abusive'  
df_abusive  
✓ 0.1s
```

	ABUSIVE	label
0	alay	Abusive
1	ampas	Abusive
2	buta	Abusive
3	keparat	Abusive
4	anjing	Abusive
...	...	...
120	rezim	Abusive
121	sange	Abusive
122	serbet	Abusive
123	sipit	Abusive
124	transgender	Abusive
125 rows × 2 columns		



	Tweet	clean_tweet	tweet_tokenized	clean_tweet_without_stopwords	label
0	- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego !!!'	di saat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego	[di, saat, semua, cowok, berusaha, melacak, perhatian, lantas, remehkan, perhatian, kasih, khusus, basic, cowok, bego]	cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego	Abusive
1	RT USER: USER siapa yang telat ngasih tau elu? edan sarap bergaul dengan cigax jifla calis sama siapa noh licew juga'	siapa yang telat memberi edan sarap bergaul dengan cigax jifla calis sama siapa licew juga	[siapa, yang, telat, memberi, edan, sarap, bergaul, dengan, cigax, jifla, calis, sama, siapa, licew, juga]	telat edan sarap bergaul cigax jifla calis licew	Abusive





02

# DATA ANALYSIS

Menggunakan Statistik Deskriptif:

- Measures of Central Tendency
- Measures of Spread
- Measures to Describe Shape of Distribution

Chapter 2

## MEASURES OF CENTRAL TENDENCY

- Membuat kolom baru berisi jumlah karakter dari tweet yang telah dibersihkan

```
df['total_char_wo_stopwords'] = df['clean_tweet_without_stopwords'].apply(len)  
✓ 0.0s
```

- Membuat kolom baru berisi jumlah kata dari tweet yang telah dibersihkan

```
df['total_word_wo_stopwords'] = df['clean_tweet_without_stopwords'].apply(lambda sent: len(sent.split()))  
✓ 0.0s
```

## MEASURES OF CENTRAL TENDENCY

- Kolom yang berisi jumlah karakter dan jumlah kata dari tweet yang dibersihkan telah dibuat:

```
df.head()
```

✓ 0.0s

	clean_tweet	tweet_tokenized	clean_tweet_without_stopwords	total_char_wo_stopwords	total_word_wo_stopwords
	di saat semua cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego	[di, saat, semua, cowok, berusaha, melacak, perhatian, lantas, remehkan, perhatian, kasih, khusus, basic, cowok, bego]	cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego	88	12

# MEASURES OF CENTRAL TENDENCY

Measures of Central Tendency adalah hal terpenting untuk melihat jumlah data, ada beberapa cara, diantaranya Mean, Median dan Mode

- Mean

```
mean_total_char = round(df["total_char_wo_stopwords"].mean(), 2)
print("Mean of total char : {}".format(mean_total_char))
✓ 0.0s
Mean of total char : 61.04

mean_total_word = round(df["total_word_wo_stopwords"].mean(), 2)
print("Mean of total word : {}".format(mean_total_word))
✓ 0.0s
Mean of total word : 8.41
```

- Median

```
df.median()
✓ 0.0s
C:\Users\lenovo\AppData\Local\Temp\i
df.median()

total_char_wo_stopwords      52.0
total_word_wo_stopwords      7.0
```

- Mode

```
df['total_char_wo_stopwords'].mode()
✓ 0.0s
0    11
Name: total_char_wo_stopwords, dtype: int64

df['total_word_wo_stopwords'].mode()
✓ 0.1s
0    2
Name: total_word_wo_stopwords, dtype: int64
```

## MEASURES OF SPREAD

Measures of spread merupakan pendekatan yang dilakukan untuk melihat persebaran data, ada beberapa cara diantaranya yaitu Range, Quartile, Interquartile, Variance & Standard Deviation

- Range

```
range_total_word = df.total_word_wo_stopwords.max() - df.total_word_wo_stopwords.min()
range_total_word
✓ 0.0s
45

range_total_char = df.total_char_wo_stopwords.max() - df.total_char_wo_stopwords.min()
range_total_char
✓ 0.0s
338
```

Range adalah cara untuk menemukan selisih antara nilai terbesar dan nilai terkecil dalam data.

$$(\text{Nilai terbesar} - \text{Nilai terkecil})$$

# MEASURES OF SPREAD

- Quartile

```
# Find quartile 1  
q1 = df.total_char_wo_stopwords.quantile(0.25)  
  
# Find quartile 2  
q2 = df.total_char_wo_stopwords.quantile(0.5)  
  
# Find quartile 3  
q3 = df.total_char_wo_stopwords.quantile(0.75)  
✓ 0.0s  
  
    q1  
✓ 0.0s  
27.0  
  
    q2  
✓ 0.0s  
52.0  
  
    q3  
✓ 0.0s  
85.0
```

Quartil yakni yang membagi data menjadi empat bagian dengan jumlah yang kurang lebih sama.

Q1: Nilai tengah antara nilai terkecil dengan median

Q2 : Nilai tengah atau Median

Q3 : Nilai tengah antara median dengan nilai terbesar

- Interquartile

```
# Find Interquartile Range  
iqr = q3 - q1  
iqr  
✓ 0.0s  
58.0
```

Interquartile adalah selisih antara persentil ke-75 dan persentil ke-25. Dengan kata lain, IQR adalah kuartil ketiga (Q3) dikurangi kuartil pertama (Q1).

# MEASURES OF SPREAD

- Outlier

Outlier adalah data yang menyimpang secara ekstrim dari rata-rata sekumpulan data yang ada

```
# lower limit  
lower_limit = q1-1.5*iqr  
  
# upper limit  
upper_limit = q3+1.5*iqr
```

✓ 0.1s

```
lower_limit
```

✓ 0.1s

-60.0

```
upper_limit
```

✓ 0.1s

172.0

```
print("Lower Limit 'total_char_wo_stopwords':", lower_limit)  
print("Minimum Value", p0)  
if lower_limit < p0:  
    print("Nothing outlier from lower limit")  
else:  
    print("Lower case have outlier")
```

✓ 0.1s

```
Lower Limit 'total_char_wo_stopwords': -60.0  
Minimum Value 0  
Nothing outlier from lower limit
```

```
print("Upper Limit 'total_char_wo_stopwords':", upper_limit)  
print("Maximum Value", p0)  
if upper_limit > p100:  
    print("Nothing outlier from upper limit")  
else:  
    print("Upper limit have outlier")
```

✓ 0.1s

```
Upper Limit 'total_char_wo_stopwords': 172.0  
Maximum Value 0  
Upper limit have outlier
```

## MEASURES TO DESCRIBE SHAPE OF DISTRIBUTION

- Variance

```
df.var()
✓ 0.1s
C:\Users\lenovo\AppData\Local\Temp\ipykernel_105\105.ipynb:1: UserWarning: This function is deprecated and will be removed in a future version. Please use .var() instead.
df.var()

total_char_wo_stopwords    1849.547903
total_word_wo_stopwords   31.263087
HS                           0.244094
Abusive                      0.236493
HS_Individual                 0.197752
HS_Group                      0.128656
HS_Religion                   0.056833
HS_Race                        0.041302
HS_Physical                    0.024078
HS_Gender                      0.022764
HS_Other                        0.203409
HS_Weak                         0.190805
HS_Moderate                     0.113238
HS_Strong                       0.034879
dtype: float64
```

- Standard Deviation

```
df.std()
✓ 0.1s
C:\Users\lenovo\AppData\Local\Temp\ipykernel_105\105.ipynb:1: UserWarning: This function is deprecated and will be removed in a future version. Please use .std() instead.
df.std()

total_char_wo_stopwords    43.006370
total_word_wo_stopwords   5.591340
HS                           0.494059
Abusive                      0.486305
HS_Individual                 0.444693
HS_Group                      0.358686
HS_Religion                   0.238397
HS_Race                        0.203229
HS_Physical                    0.155171
HS_Gender                      0.150879
HS_Other                        0.451009
HS_Weak                         0.436812
HS_Moderate                     0.336509
HS_Strong                       0.186758
dtype: float64
```

- Skewness

```
df.skew()
✓ 0.1s
C:\Users\lenovo\AppData\Local\Temp\ipykernel_105\105.ipynb:1: UserWarning: This function is deprecated and will be removed in a future version. Please use .skew() instead.
df.skew()

total_char_wo_stopwords    1.029861
total_word_wo_stopwords   0.952958
HS                           0.311631
Abusive                      0.478368
HS_Individual                 1.028331
HS_Group                      1.942718
HS_Religion                   3.687795
HS_Race                        4.496498
HS_Physical                    6.127256
HS_Gender                      6.319885
HS_Other                        0.957487
HS_Weak                         1.114289
HS_Moderate                     2.198351
HS_Strong                       4.967777
dtype: float64
```

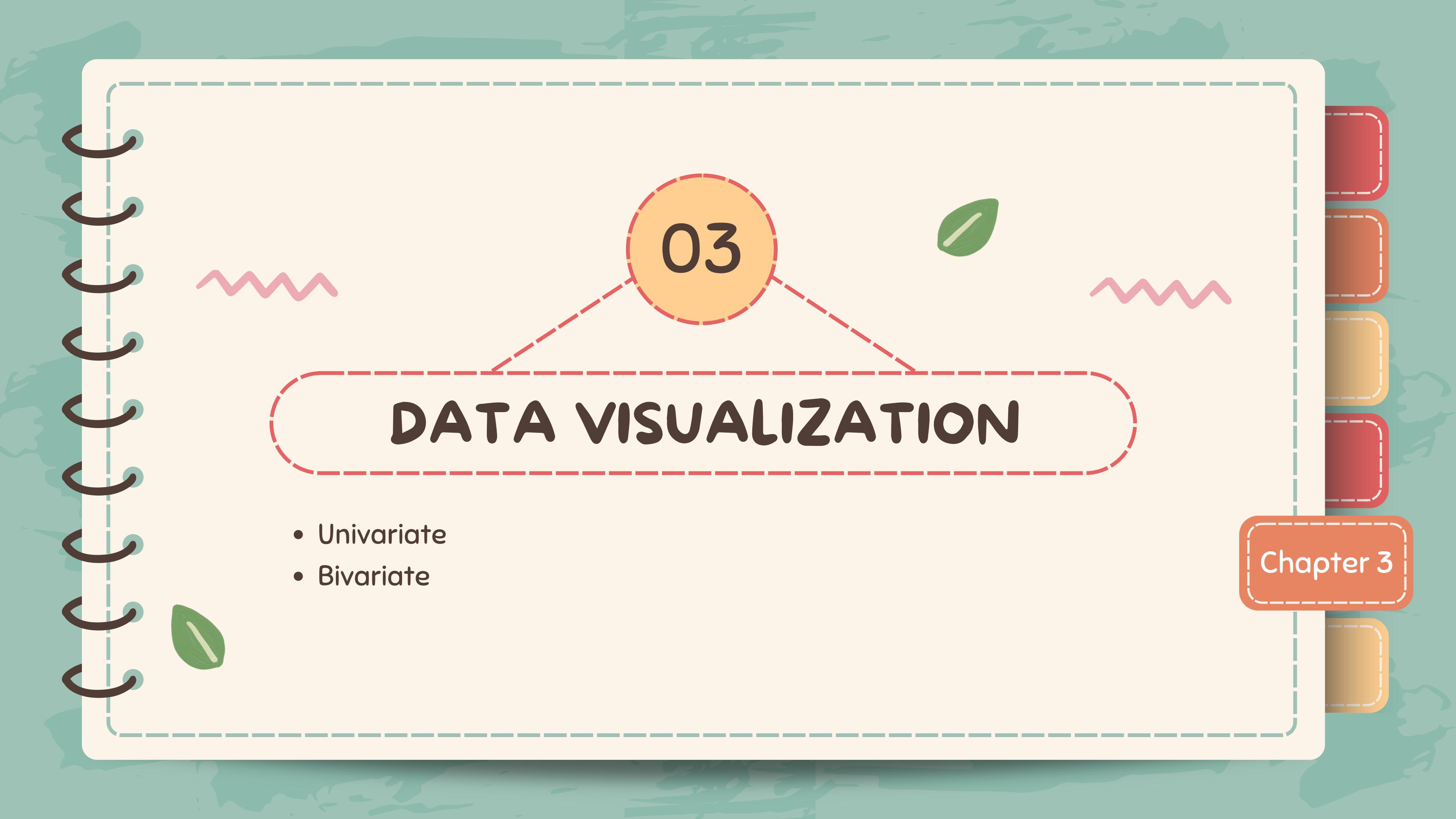
- Kurtosis

```
df.kurtosis()
✓ 0.1s
C:\Users\lenovo\AppData\Local\Temp\ipykernel_105\105.ipynb:1: UserWarning: This function is deprecated and will be removed in a future version. Please use .kurtosis() instead.
df.kurtosis()

total_char_wo_stopwords    1.013789
total_word_wo_stopwords   0.733213
HS                           -1.903178
Abusive                      -1.771435
HS_Individual                 -0.942681
HS_Group                      1.774426
HS_Religion                   11.601608
HS_Race                        18.221291
HS_Physical                    35.548712
HS_Gender                      37.946761
HS_Other                        -1.083385
HS_Weak                         -0.758476
HS_Moderate                     2.833181
HS_Strong                       22.682291
dtype: float64
```

Skewness total\_char\_wo\_stopwords dan total\_word\_wo\_stopwords lebih dari 0, artinya skewness bernilai positif

Kurtosis total\_char\_wo\_stopwords dan total\_word\_wo\_stopwords kurang dari 3, artinya platykurtic yang cenderung menghasilkan nilai outlier lebih sedikit.



03

# DATA VISUALIZATION

- Univariate
- Bivariate

Chapter 3

# UNIVARIATE ANALYSIS

- Wordcloud:

Melihat 10 kata paling sering  
muncul dari tweet netizen  
Indonesia menggunakan library  
Wordcloud:



## UNIVARIATE ANALYSIS

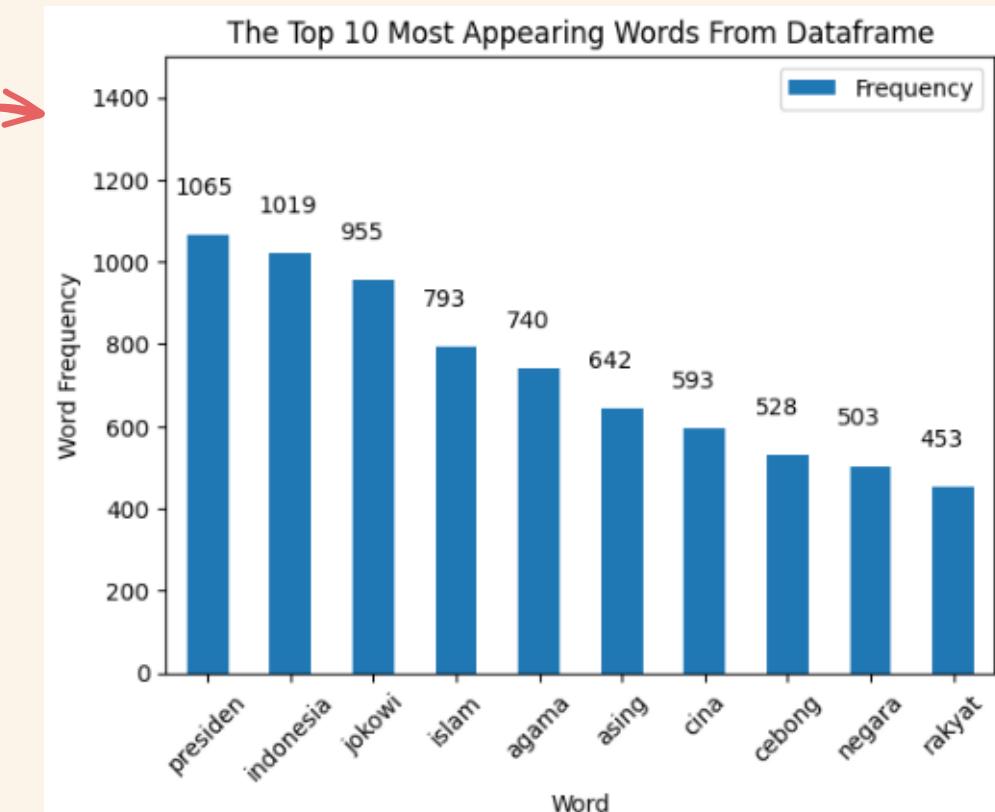
- Wordcloud:

Melihat 10 kata paling sering muncul dari tweet netizen Indonesia menggunakan barchart dari library Matplotlib:

```
df_freq = pd.DataFrame(freq_kata.most_common(10), columns = ['Word', 'Frequency'])

fig = plt.figure(figsize=(10,8))
df_freq.plot.bar(x='Word', y='Frequency')
plt.xlabel('Word')
plt.ylabel('Word Frequency')
plt.xticks(rotation=45)
plt.title('The Top 10 Most Appearing Words From Dataframe')
plt.ylim(0, 1500)
for i in range(len(df_freq)):
    plt.text(x = i-0.4, y = df_freq.loc[i,'Frequency']+100, s = df_freq.loc[i,'Frequency'], size = 10)
plt.savefig("word_distribution.png", bbox_inches = 'tight', dpi=300)
plt.show()

✓ 1.2s
```



# UNIVARIATE ANALYSIS

Berapa Banyak Kata Alay pada  
df['clean\_tweet\_wo\_stopwords'] Berdasarkan  
Kata pada df\_kamus['anakjakartaasikasik']?

```
clean_tweet_words = set(df['clean_tweet_without_stopwords'].str.split().explode())
kamus_words = set(df_kamus['anakjakartaasikasik'])

matching_words = clean_tweet_words.intersection(kamus_words)

print('Number of Alay Words:', len(matching_words))
✓ 0.2s

Number of Alay Words: 133
```

```
total_word_wo_stopwords = df['total_word_wo_stopwords'].sum()
fig, ax = plt.subplots()

x = ['Clean Tweet', 'Kata Alay']
y = [total_word_wo_stopwords-len(matching_words), len(matching_words)]

ax.bar(x,y)
plt.xlabel('Kategori')
plt.ylabel('Total Kata')
plt.ylim(0, 125000)
plt.xticks(rotation=0)
plt.title("Perbandingan Total Kata dari Clean Tweet dengan Kata Alay")
for i in range(len(y)):
    ax.annotate(str(y[i]), (x[i], y[i]), ha='center')
plt.savefig("alay_words.png", bbox_inches = 'tight', dpi=300)
plt.show()
✓ 0.8s
```



```
total_word_wo_stopwords = round(((df['total_word_wo_stopwords'].sum() - len(matching_words)) / df['total_word_wo_stopwords'].sum()) * 100,2)
alay_words = round((len(matching_words) / df['total_word_wo_stopwords'].sum()) * 100, 2)
fig, ax = plt.subplots()

x = ['Clean Tweet', 'Kata Alay']
y = [total_word_wo_stopwords, alay_words]

ax.bar(x,y)
plt.xlabel('Kategori')
plt.ylabel('Persentase (%)')
plt.ylim(0, 200)
plt.xticks(rotation=0)
plt.title("Perbandingan Total Kata dari Clean Tweet dengan Kata Alay")
for i in range(len(y)):
    ax.annotate(str(y[i]), (x[i], y[i]), ha='center')
plt.savefig("alay_words_percent.png", bbox_inches = 'tight', dpi=300)
plt.show()
✓ 1.3s
```



# UNIVARIATE ANALYSIS

Perbandingan Total Tweet dari kolom df['clean\_tweet\_wo\_stopwords'] Berdasarkan Label

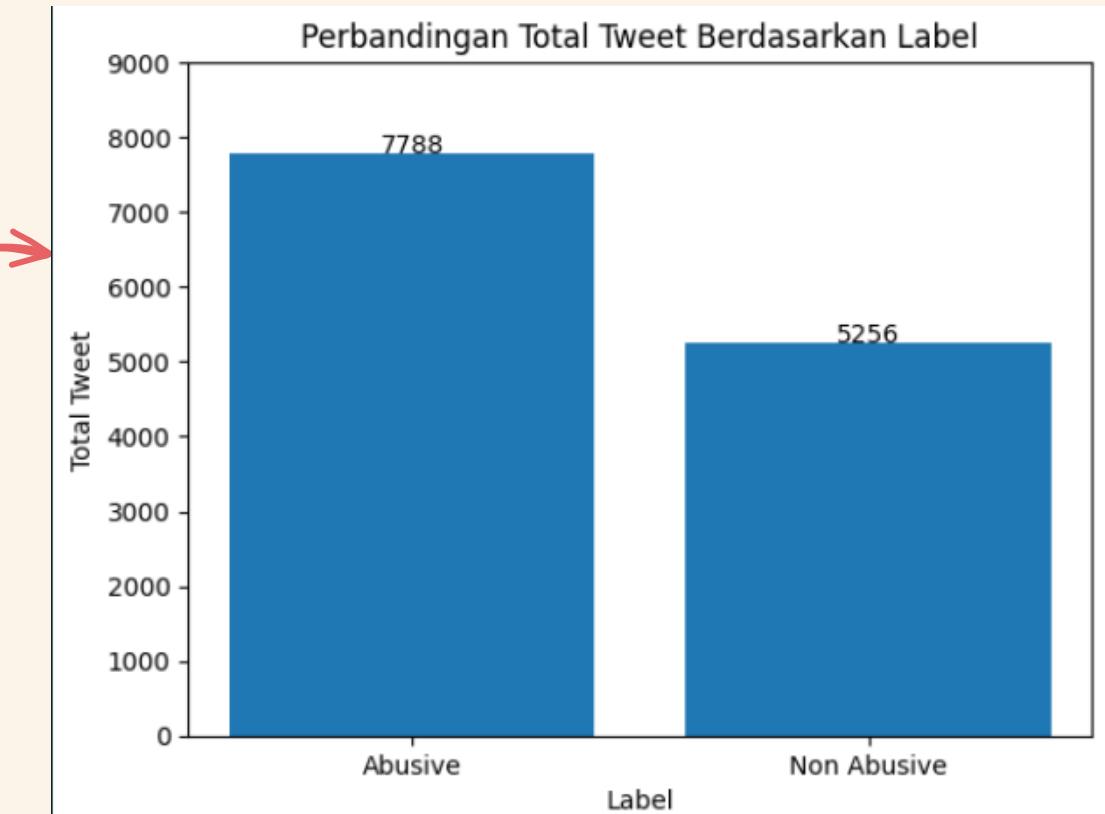
```
abusive = (df['label'] == 'Abusive').sum()
non_abusive = (df['label'] == 'Non Abusive').sum()

fig, ax = plt.subplots()

x = ['Abusive', 'Non Abusive']
y = [abusive, non_abusive]

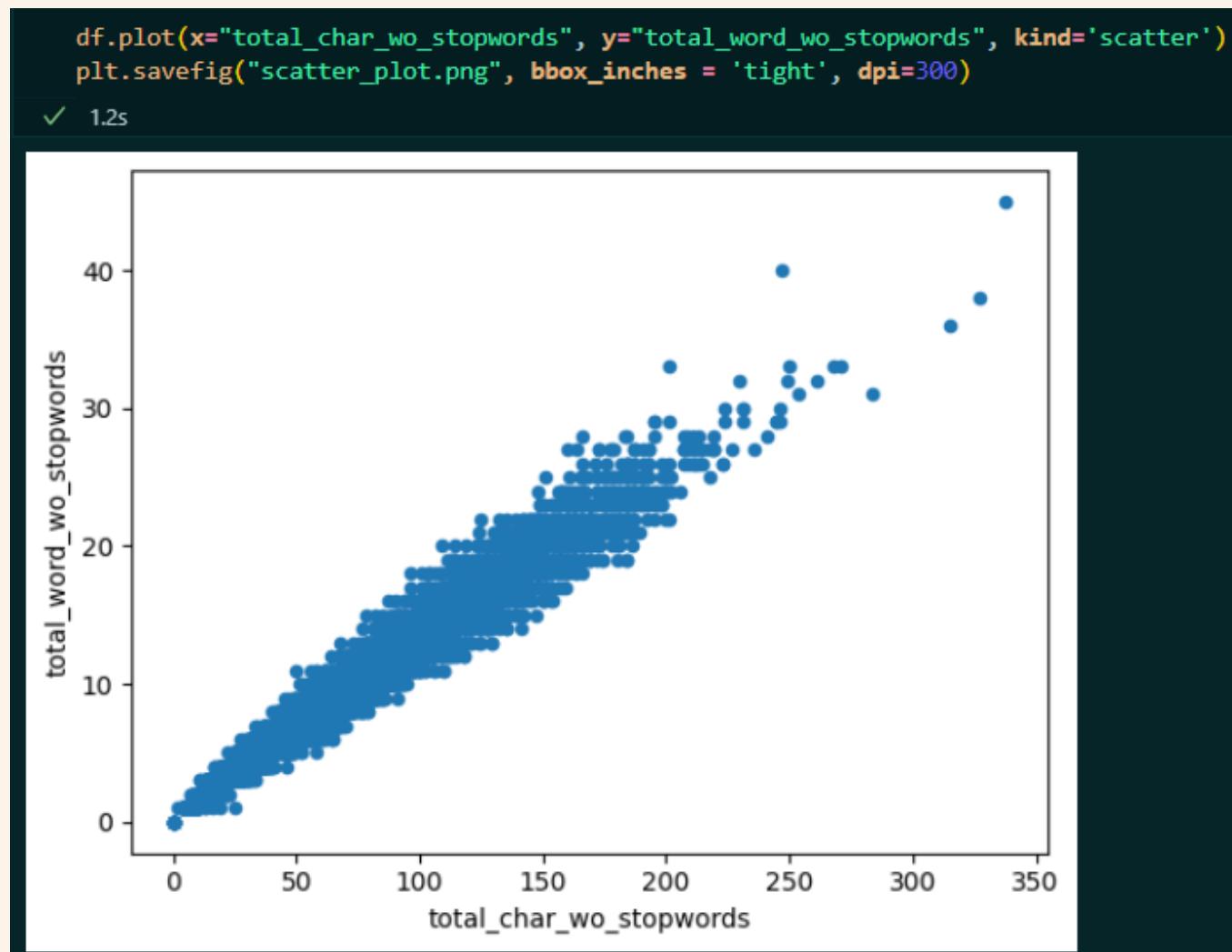
ax.bar(x,y)
plt.xlabel('Label')
plt.ylabel('Total Tweet')
plt.ylim(0, 9000)
plt.xticks(rotation=0)
plt.title("Perbandingan Total Tweet Berdasarkan Label")
for i in range(len(y)):
    ax.annotate(str(y[i]), (x[i], y[i]), ha='center')
plt.savefig("Labels.png", bbox_inches = 'tight', dpi=300)
plt.show()

✓ 0.8s
```



# BIVARIATE ANALYSIS

- SCATTER PLOT



Scatter Plot menunjukkan hasil korelasi positif antara jumlah karakter dan jumlah kata.. Kedua variabel tersebut saling mempengaruhi, pola yang dihasilkan adalah linier dari pojok kiri bawah ke pojok kanan atas

# BIVARIATE ANALYSIS

- HEATMAP PLOT

Berdasarkan Heatmap Plot diketahui bahwa antara total\_char\_wo\_stopwords dan total\_word\_wo\_stopwords berkorelasi positif, karena nilainya mendekati 1

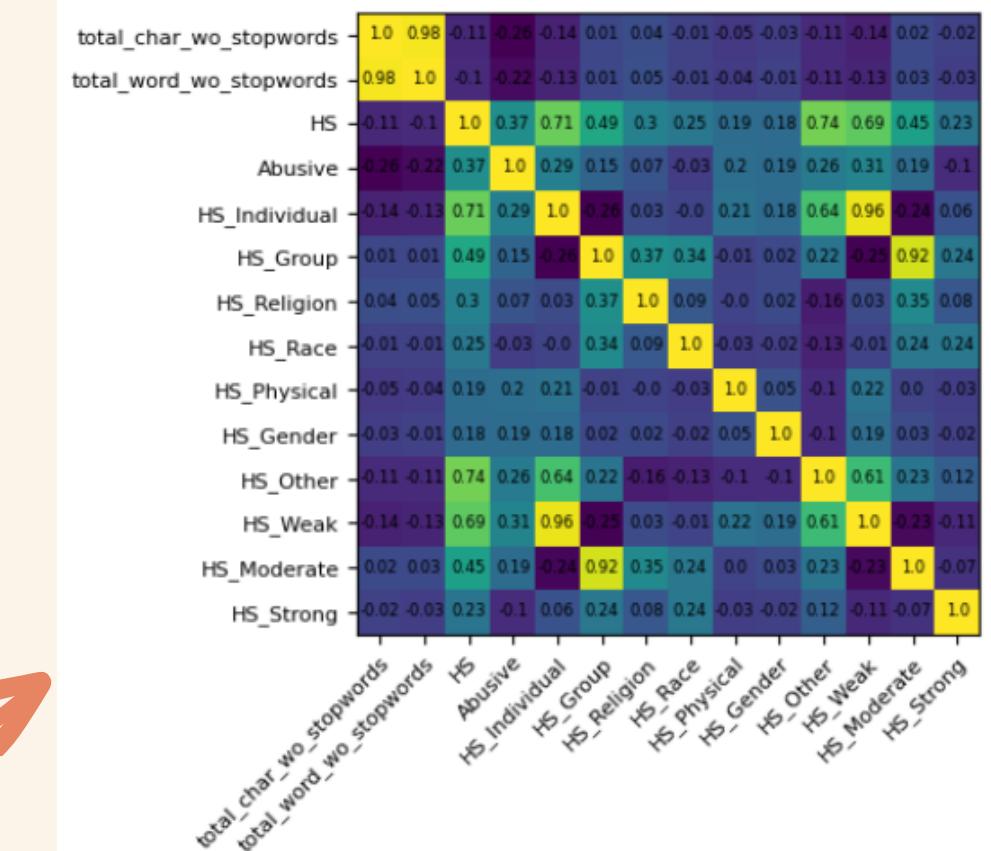
```
corr = df.corr()
fig, ax = plt.subplots()

im = ax.imshow(corr.values)

ax.set_xticks(np.arange(len(corr.columns)))
ax.set_yticks(np.arange(len(corr.columns)))
ax.set_xticklabels(corr.columns, fontsize=8)
ax.set_yticklabels(corr.columns, fontsize=8)

plt.setp(ax.get_xticklabels(), rotation=45, ha="right",
         rotation_mode="anchor")

for i in range(len(corr.columns)):
    for j in range(len(corr.columns)):
        text = ax.text(j, i, np.around(corr.iloc[i, j], decimals=2), ha="center", va="center", color="black", fontsize=6)
plt.tight_layout() # adjust subplots to fit the figure size
plt.savefig("scatter_plot.png", bbox_inches = 'tight', dpi=300)
plt.show()
✓ 2.5s
```



A heatmap visualization of a correlation matrix. The x-axis and y-axis both list the following categories: total\_char\_wo\_stopwords, total\_word\_wo\_stopwords, HS, Abusive, HS\_Individual, HS\_Group, HS\_Religion, HS\_Race, HS\_Physical, HS\_Gender, HS\_Other, HS\_Weak, HS\_Moderate, and HS\_Strong. The color scale ranges from dark purple (representing -1.0) to light yellow (representing 1.0). The diagonal elements are all white, indicating a correlation of 1.0. The highest positive correlations are visible along the top-left diagonal, particularly between total\_char\_wo\_stopwords and total\_word\_wo\_stopwords (approx. 0.98), and between HS and HS\_Individual (approx. 0.71).

total_char_wo_stopwords	1.0	0.98	-0.11	-0.26	-0.14	0.01	0.04	-0.01	-0.05	-0.03	-0.11	-0.14	0.02	-0.02
total_word_wo_stopwords	0.98	1.0	-0.1	-0.22	-0.13	0.01	0.05	-0.01	-0.04	-0.01	-0.11	-0.13	0.03	-0.03
HS	-0.11	-0.1	1.0	0.37	0.71	0.49	0.3	0.25	0.19	0.18	0.74	0.69	0.45	0.23
Abusive	-0.26	-0.22	0.37	1.0	0.29	0.15	0.07	-0.03	0.2	0.19	0.26	0.31	0.19	-0.1
HS_Individual	-0.14	-0.13	0.71	0.29	1.0	-0.26	0.03	-0.0	0.21	0.18	0.64	0.96	0.24	0.06
HS_Group	0.01	0.01	0.49	0.15	-0.26	1.0	0.37	0.34	-0.01	0.02	0.22	-0.25	0.92	0.24
HS_Religion	0.04	0.05	0.3	0.07	0.03	0.37	1.0	0.09	-0.0	0.02	-0.16	0.03	0.35	0.08
HS_Race	-0.01	-0.01	0.25	-0.03	-0.0	0.34	0.09	1.0	-0.03	-0.02	-0.13	-0.01	0.24	0.24
HS_Physical	-0.05	-0.04	0.19	0.2	0.21	-0.01	-0.0	-0.03	1.0	0.05	-0.1	0.22	0.0	-0.03
HS_Gender	-0.03	-0.01	0.18	0.19	0.18	0.02	0.02	-0.02	0.05	1.0	-0.1	0.19	0.03	-0.02
HS_Other	-0.11	-0.11	0.74	0.26	0.64	0.22	-0.16	0.13	-0.1	-0.1	1.0	0.61	0.23	0.12
HS_Weak	-0.14	-0.13	0.69	0.31	0.96	-0.25	0.03	-0.01	0.22	0.19	0.61	1.0	0.23	-0.11
HS_Moderate	0.02	0.03	0.45	0.19	-0.24	0.92	0.35	0.24	0.0	0.03	0.23	0.23	1.0	-0.07
HS_Strong	-0.02	-0.03	0.23	-0.1	0.06	0.24	0.08	0.24	-0.03	-0.02	0.12	-0.11	-0.07	1.0



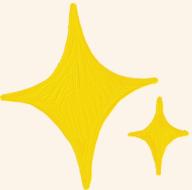
04

## RESTFUL API

- Swagger
- Flask

Conclusion

## RESTFUL API

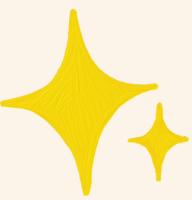


Pada penelitian ini data yang digunakan pada awalnya memang sudah lengkap, namun sebagai seorang Data Scientist, Data Analyst atau Data Engineer memerlukan proses yang disebut pembersihan data (Data Cleansing)

Saya sebagai data scientist akan mencoba membuat RESTful API yang dapat melakukan proses pembersihan data teks secara otomatis, baik berupa file, berdasarkan teks, kolom, dan lainnya yang akan dijelaskan pada implementasi API lebih lanjut.

RESTful API ini dibangun menggunakan Python dan Flask, kemudian didokumentasikan oleh Swagger UI, RESTful API ini terhubung ke RDBMS (SQLite).

# RESTFUL API



01

Python

Python merupakan bahasa pemrograman yang dipakai. Library yang digunakan untuk membersihkan data adalah Pandas, Regex, NLTK

02

SQLite

SQLite digunakan sebagai database untuk mengakses Tabel dan sebagai penyimpanan.

03

Flask

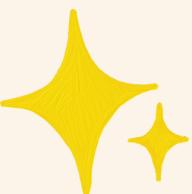
Flask sebagai framework RESTful API.

04

Swagger UI

Swagger UI untuk membuat dokumentasi RESTful API

# SWAGGER UI

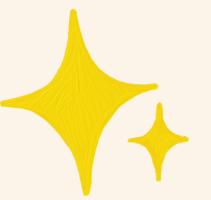


Untuk mendokumentasikan API saya menggunakan Swagger UI. Terdapat 5 (lima) endpoint pada Swagger ini:

The screenshot shows the Swagger UI interface for a Data Processing and Modeling API. The top navigation bar includes the Swagger logo, a link to '/docs.json', and an 'Explore' button. The main title is 'API Documentation for Data Processing and Modeling' (version 1.0.0). Below the title, it says '[ Base URL: 127.0.0.1:5000 ] /docs.json'. A subtitle reads 'Dokumentasi API untuk Data Processing dan Modeling'. The interface is organized into sections: 'MASUKAN INDEX YANG INGIN DIBERSIHKAN SECARA OTOMATIS', 'MASUKAN TEKS YANG INGIN DIBERSIHKAN SECARA OTOMATIS', and 'UPLOAD FILE (.csv), LALU PILIH APA YANG INGIN ANDA LAKUKAN'. Each section contains a list of POST requests with their respective URLs. At the bottom right, it says '[Powered by Flagger 0.9.5]'. The background of the slide features a spiral notebook design with green leaves.

- MASUKAN INDEX YANG INGIN DIBERSIHKAN SECARA OTOMATIS**
  - POST /bersihkan\_dataframe\_berdasarkan\_index
- MASUKAN TEKS YANG INGIN DIBERSIHKAN SECARA OTOMATIS**
  - POST /bersihkan\_dataframe\_berdasarkan\_text
- UPLOAD FILE (.csv), LALU PILIH APA YANG INGIN ANDA LAKUKAN**
  - POST /data\_sebelum\_dibersihkan
  - POST /upload\_file\_bersihkan\_download
  - POST /upload\_file\_bersihkan\_download\_JSON

# SWAGGER UI



## 1. Bersihkan Dataframe Berdasarkan Index

MASUKAN INDEX YANG INGIN DIBERSIHKAN SECARA OTOMATIS

POST /bersihkan\_dataframe\_berdasarkan\_index

Parameters

Name Description

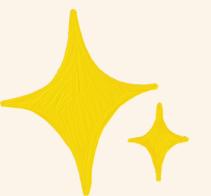
**index** \* required  
integer  
(formData)  
Masukkan index dari kolom tweet yang ingin kamu bersihkan, Pilih Indeks Ke 0 - 13168  
2401

**note**  
string  
(formData)  
Deskripsi.  
Index 2401

Execute Clear

Responses Response content type application/json

# SWAGGER UI



## 1. Bersihkan Dataframe Berdasarkan Index

Request URL  
`http://127.0.0.1:5000/bersihkan_dataframe_berdasarkan_index`

Server response

Code Details

200

Response body

```
{ "before": [ "Dgn rasa ikan gurih dgn hrg mrah, bsa didptkn disini lho! Knjngi : Pmpek plmbng, jln wr supratman senggol budaya, Tohpati" ], "clean": [ "ikan gurih mrah didptkn knjngi pmpek plmbng supratman senggol budaya tohpati" ] }
```

Download

Response headers

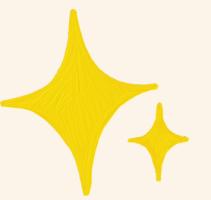
```
connection: close
content-length: 251
content-type: application/json
date: Sat, 01 Apr 2023 16:24:06 GMT
server: Werkzeug/2.2.3 Python/3.10.9
```

Responses

Code Description

- 200 Succesful response
- 400 Bad request
- 500 Internal Server Error

# SWAGGER UI



## 2. Bersihkan dataframe berdasarkan teks

MASUKAN TEKS YANG INGIN DIBERSIHKAN SECARA OTOMATIS

POST /bersihkan\_dataframe\_berdasarkan\_text

Parameters

Name	Description
text * required	Teks yang ingin dibersihkan.
string (formData)	#kom3 Menkumham: pernah kita debat UU F
note	Deskripsi
string (formData)	Text cleansing

Cancel

Execute Clear

Responses

Response content type application/json

Curl

```
curl -X POST "http://127.0.0.1:5000/bersihkan_dataframe_berdasarkan_text" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "text=%23kom3%20Menkumham%3A%20pernah%20kita%20debat%20UU%20Pilkada%20dulu%2C%20diketok%2C%20pendekatan%20waktu%20itu%20presiden%20mengeluarkan%20Perppu%2C%20tapi%20skrg%20ini%20presiden%20tdk%20menandatanganinya&note=Text%20cleansing"
```

# SWAGGER UI



## 2. Bersihkan dataframe berdasarkan teks

Request URL  
`http://127.0.0.1:5000/bersihkan_dataframe_berdasarkan_text`

Server response

Code Details

200 Response body  
menkumham pernah kita debat pilkada dulu diketok pendekatan waktu presiden mengeluarkan perppu tapi skrg presiden menandatanganinya [Download](#)

Response headers  
`connection: close  
content-length: 131  
content-type: text/html; charset=utf-8  
date: Sat, 01 Apr 2023 16:31:14 GMT  
server: Werkzeug/2.2.3 Python/3.10.9`

Responses

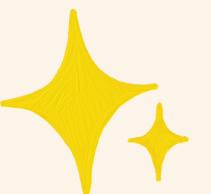
Code Description

200 Succesful response

400 Bad request

500 Internal Server Error

# SWAGGER UI



## 3. Upload File, dan lihat data sebelum dibersihkan

UPLOAD FILE (.csv), LALU PILIH APA YANG INGIN ANDA LAKUKAN

POST /data\_sebelum\_dibersihkan

Parameters

Name	Description
file * required	File yang mau dibersihkan
file (formData)	<input type="button" value="Choose File"/> data.csv
directory_path string (formData)	Masukkan direktori tanpa spasi (" ") untuk mendownload file
directory_path - Masukkan direktori tanpa sp:	

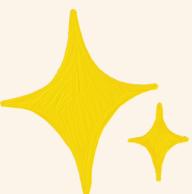
Responses

Response content type

Curl

```
curl -X POST "http://127.0.0.1:5000/data_sebelum_dibersihkan" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "file=@data.csv;type=text/csv"
```

# SWAGGER UI



3. Upload File, dan lihat data sebelum dibersihkan

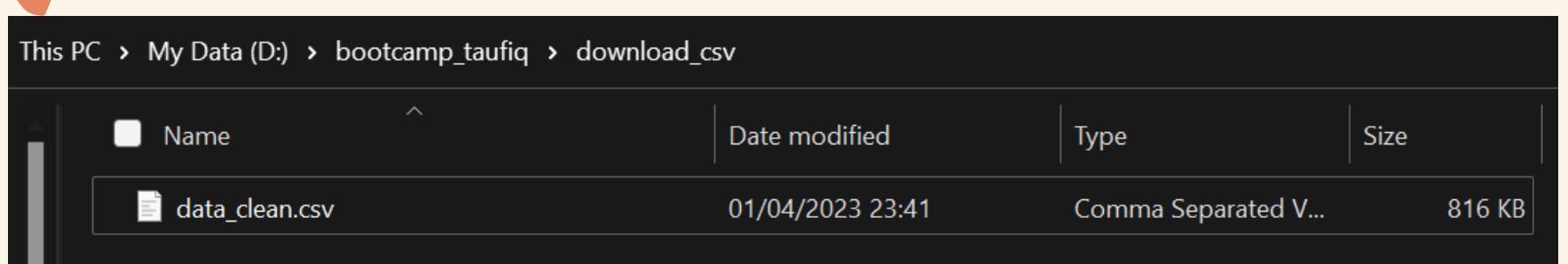
# SWAGGER UI

## 4. Upload File, Bersihkan & Download

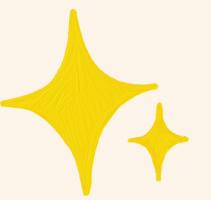
POST /upload\_file\_bersihkan\_download

Parameters

Name	Description
file * required	File yang mau dbersihkan
file (formData)	<input type="button" value="Choose File"/> data.csv
directory_path string (formData)	Masukkan direktori tanpa spasi (" ") untuk mendownload file <input type="text" value="D:\bootcamp_taufiq\download_csv"/>



# SWAGGER UI



## 5. Upload File, Bersihkan & Download

POST /upload\_file\_bersihkan\_download\_JSON

Parameters

Name	Description
file * required	File yang mau dibersihkan
file (formData)	<input type="button" value="Choose File"/> data.csv
directory_path string (formData)	Masukkan direktori tanpa spasi (" ") untuk mendownload file
	directory_path - Masukkan direktori tanpa sp:

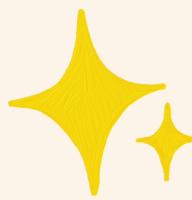
Responses

Response content type

Curl

```
curl -X POST "http://127.0.0.1:5000/upload_file_bersihkan_download_JSON" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "file=@data.csv;type=text/csv" -F "directory_path=D:\bootcamp_taufiq\download_csv"
```

# SWAGGER UI



## 5. Upload File, Bersihkan & Download

Request URL

[http://127.0.0.1:5000/upload\\_file\\_bersihkan\\_download\\_JSON](http://127.0.0.1:5000/upload_file_bersihkan_download_JSON)

Server response

Code Details

200

Response body

```
{"Tweet": {"0": "disaat cowok berusaha melacak perhatian lantas remehkan perhatian kasih khusus basic cowok bego", "1": "telat ngasih edan sarap bergaul cigax jifla calis licew", "2": "kada jabol", "7": "kelar watich aldnoah zero kampret emang endingnya karakter utama cowonya kena friendzone bray", "8": "admin belanja port terbaik makan kepala milo kepala horlicks cendol topping intahan jokowi nyata", "14": "guru enakan jablay guru esde knya menikmati pecun guru", "15": "lawan bicara intelek otak kencing onta mengakui hadis nabi sahabat kafir menolak ahlunnur", "16": "mampuan pemerintahannya menghadapi situasi ekonomi", "23": "pelajar bilah hilir deklarasi anti hoax pilkada damai", "24": "bandara udara internasional kertajati dibangun gubernur ahmad herosong", "29": "nenek nenek heran cebong biasay bohong", "30": "islam nusantara produk dipasarkan gencar antek ntara anti arab aseng hasyim muzadi", "31": "habis sahur sampe sibayik udah nematahari", "36": "ra bajingan melukai lady menerima ganjaran seratus kali lipat sanji", "37": "ajakan menolak berita hoax sukseskan pilkada wilayah kota keidiri", "38": "presiden jokowi luar ter shame mpok silvy agus", "43": "keanekaragaman budaya suku agama sesungguhnya kekayaan djarot hadapan peserta rakercabsu", "44": "islam dilokalisasi kristen nusantaranya liat gereja bali impor", "47": "sembahyang mengadopsi tradisi lokal kerjaan hindu lahmabhyanga lahir tradisi ritusnyembahhyang tunggal denganal menyembah", "48": "pendikan bodoh larang bawa agama biar iplng ngeden dungu", "52": "pinokio umur taonan", "53": "cerita silat ping indonesia liang shen penulis china", "54": "benci umat islam", "55": "provokasi mayat politisasi agama penyebab kekal sendiri", "59": "suka konser konser boros", "60": "makan racun ulama hukum rokok mubah", "61": "susilo bambang yudhoyono presiden khofifah emil masayarakat baguss", "62": "udah kali kpoper ngomong agama", "68": "lengserkan jokowi bangsat", "69": "selamat kartini", "70": "bajingan homo anak kamar mandi laki biarkan", "71": "cocot", "72": "budaya indonesia", "73": "ngajarin solat ajah aseng gimana", "78": "bunda ngentot", "79": "china babi china maling yhiwhjp", "80": "karno anti asing beliau jawa coba adain konsensus jawa anti asing asing kebo ohannya u keliru tutup akun", "81": "alqur alqur sndri bbrp suku smpai dtik mkna faktualnya umat islam yakini makna tersirat fiksi", "87": "kasar kampang", "88": "bentar rilis album onta", "89": "cebong bang", "90": "najis bange utama lengserkan jokowi", "96": "kena hestek udah kepanasan kepo banget", "97": "maju duit goyang kaki pape duit maju bodo", "98": "hkbp berdiri pecahnya kristen khatolik zaman penjajahan bertukang tipu penjilat penguasa ketahuan gerakin dibayar pakai nasi bungkus propaganda nasi bungkus gagal", "103": "wuh cebong sewot", "104": "gerakan menekankan kerja keras total tingkatkan bertepuk tangan", "108": "anti aseng", "109": "selamat semoga berkiprah tingkat nasional", "110": "ulama", "111": "ketahui diketapang asing berkerja perusahaan tambang buksid viral pilpres konspirasi dalamnya", "116": "genap april digelar hajatan pilpres acara tahunan negara indonesia memilih presiden semoga senantiasa berjalan aman lancar amin", "117": "pantat", "118": "ngak sidang paya cebong", "121": "mantan mantan mgerasa sedih ampe kepikiran gabisa tidur lohh ngerasa anak brengsek laki", "122": "cina perusak bangsa usir stuju", "123": "scra konstitusional ri umat islam", "127": "cowo jago drasex ngentot adek iuvian", "128": "kroasia kapir tauu ngapain blajar kapir lupa kapling sorga", "129": "mentri suharto homo", "130": "culun cemen berubah perl hajan", "135": "mati kebanyakan minum", "136": "ahok mrka trus hdpya", "137": "teng ekxes akibat tersandera hutang china", "138": "pride yunani hyperania keangkuhan harga kesombonganall sebur tolo partai tololai tolol", "142": "ting ting goyang raffi ahmad nagita slavina kampungan norak", "143": "video malaysia mengutip pernyataan", "144": "buru buru tolak wacana pilkada dani ratna paait onta lobang anus jngan takut bisa obat penenang", "151": "tunggu komando imam bejat cabul taik beer bawa ulama r banj sempak", "152": "kabinet berantakan cah belah persa lang dimana pdip incumbent dikalahkan mutlak mega melawan prabowo pilpres selesai poros pilpres salah lawan", "157": "kayak benci kaleng gatot gaduh gentleman mundur bersaing pemilihan banu banu kristen syriac salah perbedaan banu yahudi tetep pake eloah eloim yahwe syriac pake alah allah", "163": "muka merah cendol setan cuaca skrg panas", "164": "pemberitaan saracen siahaan", "166": "ngeliat perempuan gini status udah nemplok laki laki haduh ta slavina blak blakan ting ting kampungan viva", "167": "lupa kangen umat katolik minoritas diperhatika gubernur putus", "172": "wakakaka mengalaminya tetiba muncul grub bani hadi mbahku kenal", "173": "abam pres badar nuqaba pling boekk hebat bidang agama tkut tgur member sndiri lawak hambar pling skrg ekonomi mocat marit kacau", "177": "aelaa silit keheler ngana dikasianin sawang sosmed sadar bbrana seitanse kemarin ceritain dwuhb", "178": "cacat", "179": "rakyat makan asnal hetor
```

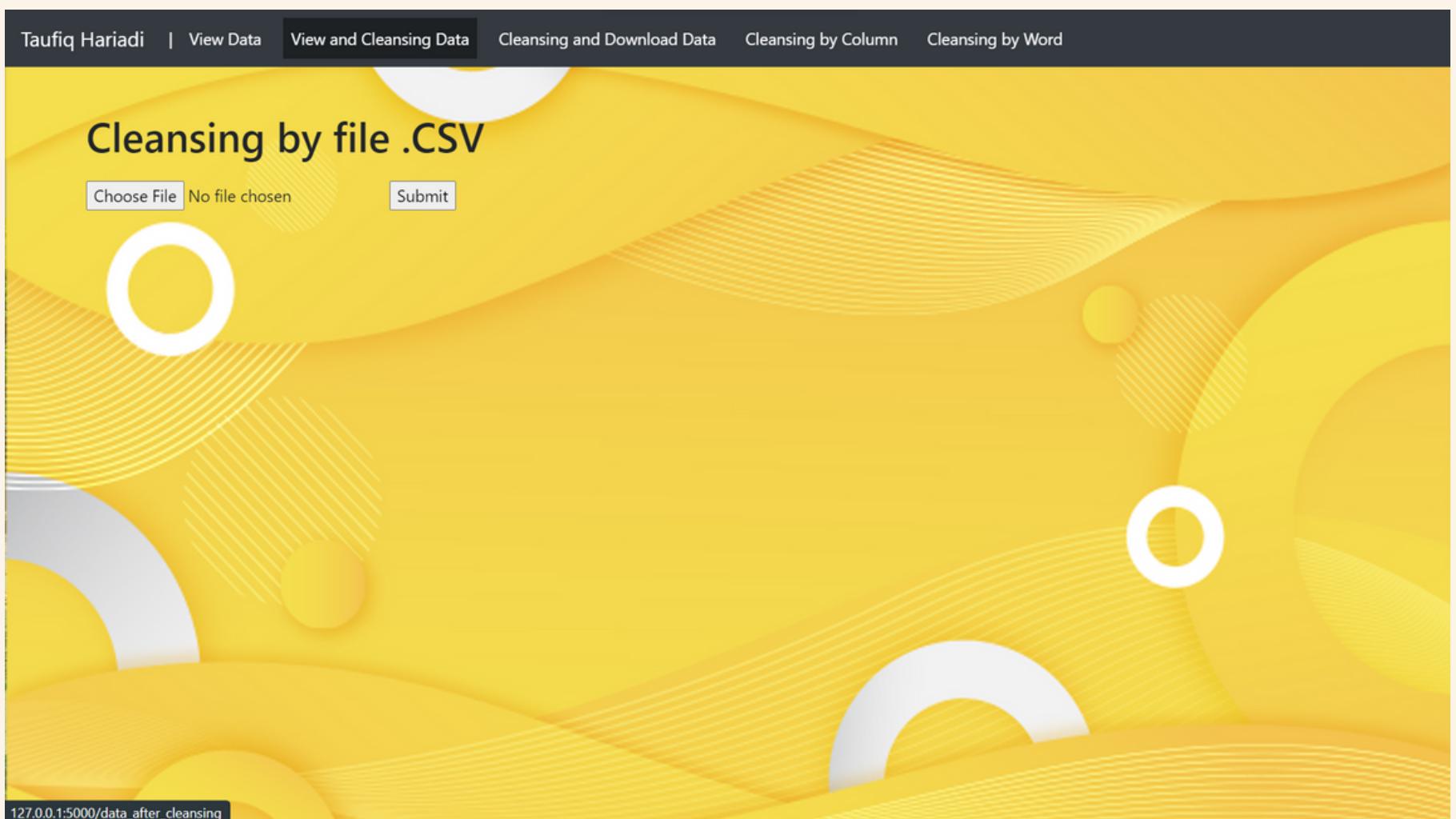
# FLASK

1. Lihat Data Sebelum dibersihkan

The screenshot shows a web application interface titled "View Data by file .CSV". At the top, there is a navigation bar with the user "Taufiq Hariadi" and links for "View Data", "View and Cleansing Data", "Cleansing and Download Data", "Cleansing by Column", and "Cleansing by Word". Below the navigation bar, the main content area has a yellow background with abstract white circular patterns. It contains a file input field labeled "Choose File" with the placeholder "No file chosen" and a "Submit" button.

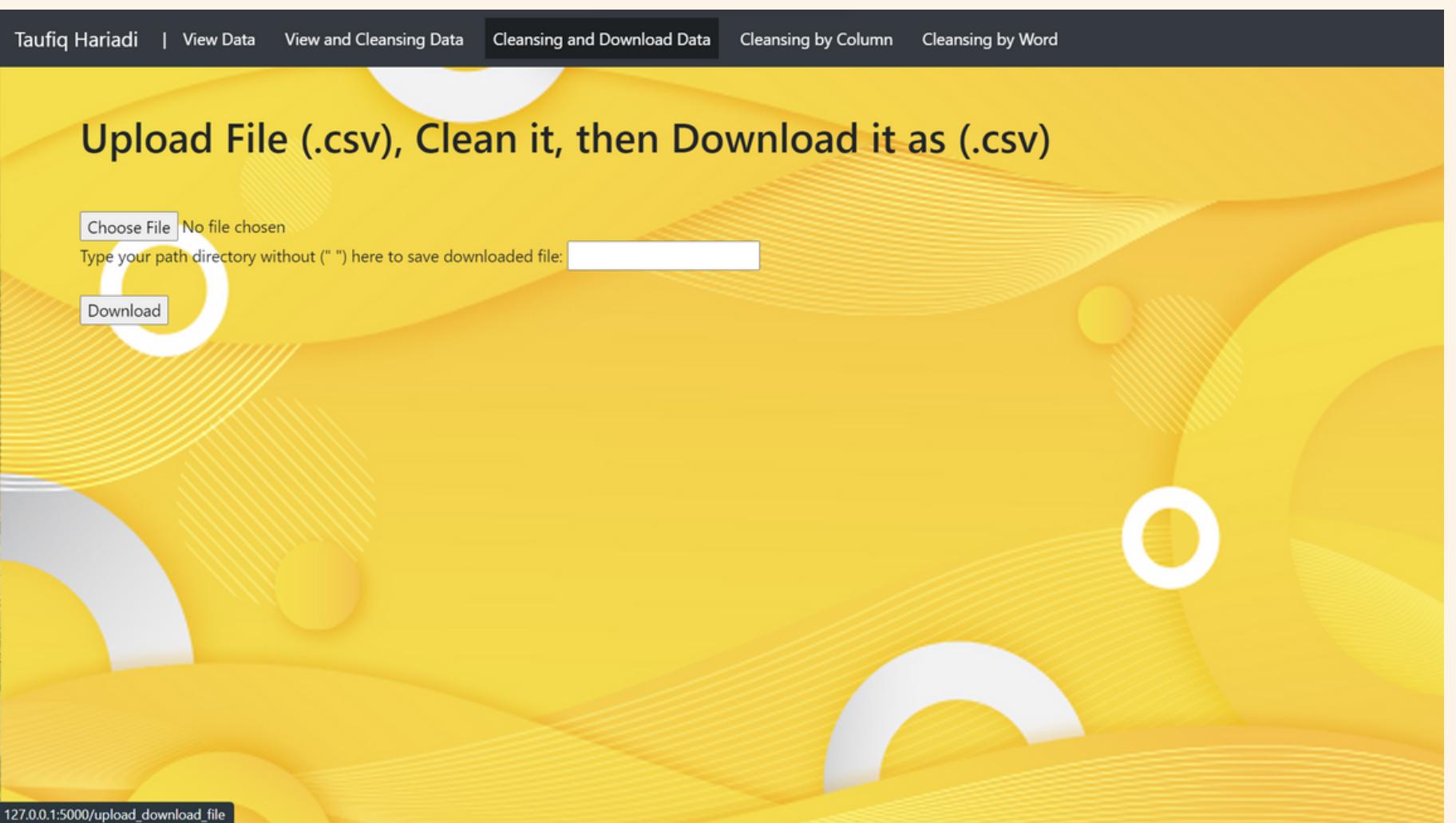
# FLASK

2. Lihat data yang sudah dibersihkan



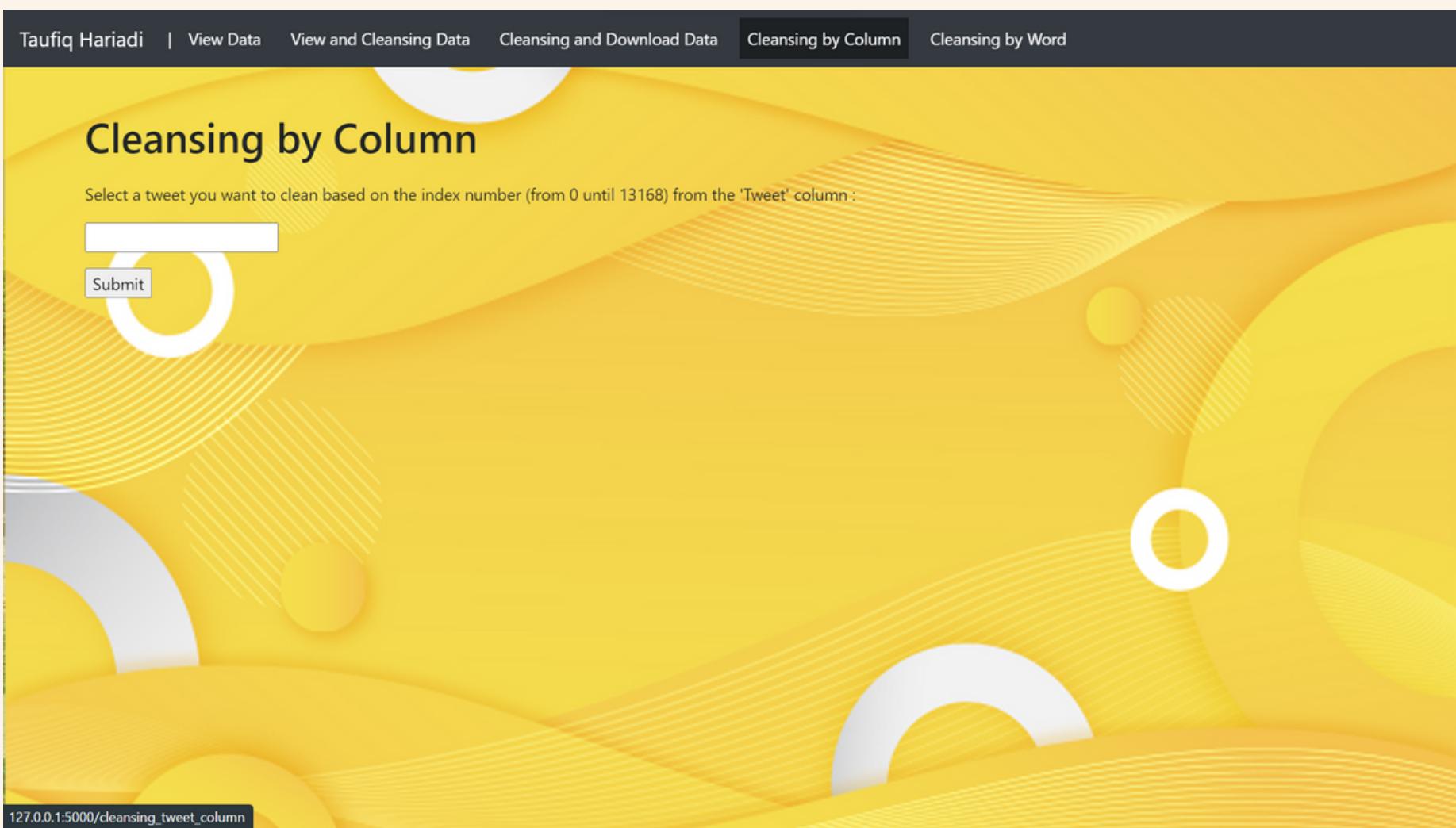
# FLASK

3. Lihat data yang sudah dibersihkan dan Download



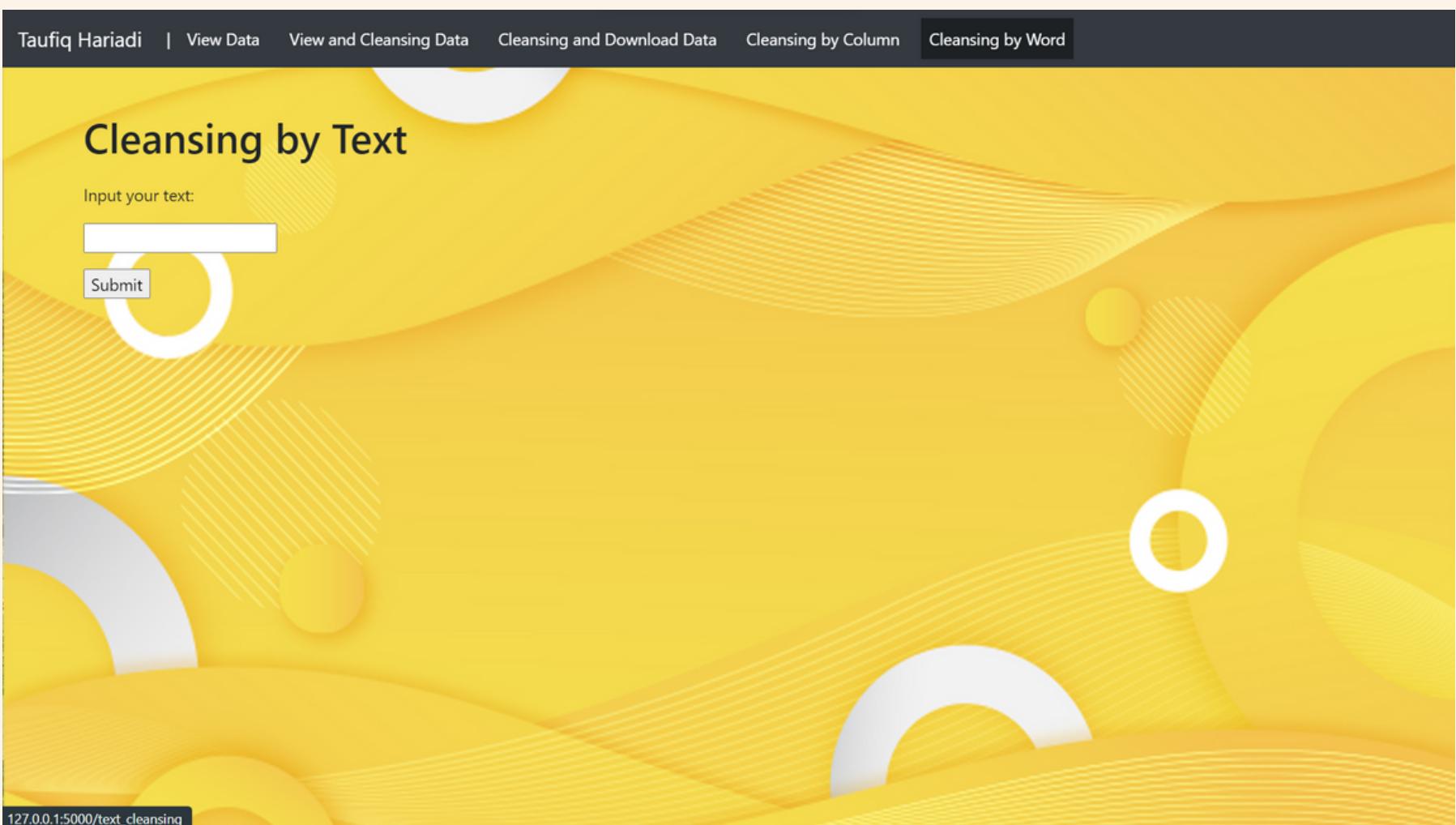
# FLASK

## 4. Bersihkan Data Berdasarkan Kolom



# FLASK

## 5. Bersihkan Data Berdasarkan Teks



05

## CONCLUSION

Conclusion

# CONCLUSION - DESCRIPTIVE STATISTICS

## Measures of Central Tendency

**61.68**

Mean of Total Character

**8.31**

Mean of Total Word

**53**

Median of Total Character

**7**

Median of Total Word

**11**

Mode of Total Character

**3**

Mode of Total Word

## Measures of Spread

**338**

Range of Total Character

**45**

Range of Total Word

**28, 53, 97**

Q1, Q2, Q3

**58**

Interquartile

**1875.5, 30.3**

Variance of Total Character & Word

**43.3, 5.5**

Standard Deviation of Total Character & Word

## Measures to Describe Shape of Distribution

**1.01, 0.95**

Skew of Total Character & Word

**0.95, 0.78**

Kurtosis of total Character & Word

- Skewness bernilai positif
- Kurtosis menghasilkan nilai outlier lebih sedikit



# Thank You

By M. Taufiq Hariadi