A.PRE PROCESSING

1. Load CSV to DataFrame

```
import pandas as pd
import re
import nltk

# pd.set_option('display.max_rows', None)
df = pd.read_csv('data_tweets.csv', encoding='utf-8')
df.drop(columns=['is_replied', 'reply_to_users', 'parent_tweet_id', 'reply_to_users', 'username', 'text_html',
                 'timestamp_epochs', 'tweet_url', 'user_id','is_reply_to','has htags','links','has_media',
                 'img_urls','video_url'], inplace=True)
df = df.reindex(columns=['tweet_id', 'screen_name','timestamp','text','likes', 'retweets','replies'])
df.sort_values(by='timestamp',ascending=True, inplace=True)
```

2. Text Cleaning

```
# Case Folding
df.text = df.text.str.lower()

# Hapus Angka
df.text = df.text.replace({r"\d+":''}, regex=True)

# Hapus Link
df.text = df.text.replace({r"http\S+":''}, regex=True)
df.text = df.text.replace({r"pic.twitter.com[A-Za-z0-9!”#$%&’()*+,-./:;<=>?@[\]^_`{|}~]+":''}, regex=True)

# Hapus Titik
df.text = df.text.replace({r"[.]":' '}, regex=True)

# Hapus Mention
df.text = df.text.replace({'@[A-Za-z0-9_]+':''}, regex=True)

# Hapus Hashtags
df.text = df.text.replace({'#[A-Za-z0-9_]+':''}, regex=True)

# Hapus WhiteSpace
df.text = df.text.replace({"\r":' '}, regex=True)
df.text = df.text.replace({"\n":' '}, regex=True)

# Hapus Semua Karakter kecuali huruf dan spasi
df.text = df.text.replace({'[^a-z\s]':' '}, regex=True)

# Hapus Kata yang kurang dari 3 huruf
df.text = df.text.replace({r'\b\w{1,3}\b':''}, regex=True)

# Hapus multi spasi
df.text = df.text.replace({' +':' '}, regex=True)
```

```
df.text = df.text.str.strip()

# Hapus Data Duplikat
df.drop_duplicates(subset ="tweet_id", keep = False, inplace = True)

# Hapus Blank Row
df = df[df.text != '']
df = df.dropna(subset = ['text'])

df
```

3. Stopword Remove

```
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory

stopword = StopWordRemoverFactory().create_stop_word_remover()
df['text'] = df.apply(lambda row: stopword.remove(row['text']), axis=1)

df
```

4. Stemming

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

stemmer = StemmerFactory().create_stemmer()
df['text'] = df.apply(lambda row: stemmer.stem(row['text']), axis=1)

df
```

5. Export to CSV

```
export_csv = df.to_csv (r'data_preprocessing.csv', index = None, header=True)
```

B. PROCESSING

1. Import CSV

```
from ast import literal_eval
import pandas as pd
import io
df = pd.read_csv('data_preprocessing.csv', encoding='utf-8')
df['text'] = df['text'].apply(literal_eval)
df['timestamp'] = pd.to_datetime(df['timestamp'])
pd.set_option('display.max_rows', None)
```

2. Analisis Sentimen

```
def gantiBaku(data):
    rep = pd.read_csv('replace.csv', encoding='utf-8')
    temp = []
    for x in data:
        a = rep.loc[rep['text'] == x]
        if len(a['text']) != 0:
            temp.append(a['replace'].iloc[0])
        else:
            temp.append(x)
    return temp

def filterKata(data,input):
    with open(input, 'r') as f: poslist = [line.strip() for line in f]
    hasil = [x for x in data if x in poslist]
    return hasil

def cekSentimen(data):
    hasil = len(data['positif']) - len(data['negatif'])
    if (hasil<0):
        return 'Negatif'
    elif (hasil>0):
        return 'Positif'
    return 'Netral'

df['text'] = df.apply(lambda row: gantiBaku(row['text']), axis=1)
df['positif'] = df.apply(lambda row: filterKata(row['text'],'positif.txt'), axis=1)
df['negatif'] = df.apply(lambda row: filterKata(row['text'],'negatif.txt'), axis=1)
df['sentimen'] = df.apply(lambda row: cekSentimen(row), axis=1)
df
```

3. Analisis Sentimen

3.1 Sentimen Keseluruhan

```
plo = df['sentimen'].value_counts().plot.bar(title='Tweet Sentiment UMM Campus 2019')
```

3.2 Sentimen Berdasarkan Bulan

```
import calendar
import numpy as np
import matplotlib.pyplot as plt
d = {'timestamp':'Month','sentimen':'Sentimen'}
dataBulan = (df.groupby([df['timestamp'].dt.month, 'sentimen'])['sentimen']
        .count()
        .reset_index(name='Count')
        .rename(columns=d))
```

```
dataBulan['Month'] = dataBulan['Month'].apply(lambda x: calendar.month_abbr[x]
)
positif = dataBulan.loc[(dataBulan['Sentimen'] == 'Positif')]
negatif = dataBulan.loc[(dataBulan['Sentimen'] == 'Negatif')]
netral = dataBulan.loc[(dataBulan['Sentimen'] == 'Netral')]


# create plot
fig = plt.figure(figsize=(10,5))
ax = plt.subplot(111)
index = np.arange(12)
bar_width = 0.3
opacity = 0.8

rects1 = plt.bar(index, positif['Count'], bar_width,
alpha=opacity,
color='g',
label='Positif')

rects2 = plt.bar(index + bar_width, netral['Count'], bar_width,
alpha=opacity,
color='b',
label='Netral')

rects3 = plt.bar(index + bar_width + bar_width, negatif['Count'], bar_width,
alpha=opacity,
color='r',
label='Negatif')

plt.xlabel('Bulan')
plt.ylabel('Jumlah')
plt.title('Sentimen Berdasarkan Bulan')
plt.xticks(index + bar_width, positif['Month'])
plt.legend()

plt.tight_layout()
plt.show()
```