Implementation of tokenization, stemming/ Lemmatization and stop words removal

a) Display all tokens in the form of word cloud

b) Apply stop words removal

c) Display the result of stemming and lemmatization

Para1: You are musicophile,” One who loves music”. In your interactions with your MCA classmates you found another musicophile ,”Ameyatma”(classmate) and become friends. Now you want to flaunt your programming skills and love for music to your friend. Write a class called “showoff” which has a instance variable “song”(String datatype) and a method called “unscramble”. The method unscrambles the words in the string “song” and prints it. The variable “song” is a stanza from a song but is strange. It has words and numbers in between them, you need to order the words depending on the numbers and print the correct stanza. Consider the following example and run your program for the given two testcases. Create an object of the class and call the method.

Para2: Social media mining is the process of obtaining big data from user-generated content on social media sites and mobile apps in order to extract actionable patterns, form conclusions about users, and act upon the information, often for the purpose of advertising to users or conducting research. The term is an analogy to the resource extraction process of mining for rare minerals. Resource extraction mining requires mining companies to shift through vast quantities of raw ore to find the precious minerals; likewise, social media mining requires human data analysts and automated software programs to shift through massive amounts of raw social media data in order to discern patterns and trends relating to social media usage, online behaviours, sharing of content, connections between individuals, online buying behaviour, and more. These patterns and trends are of interest to companies, governments and not-for-profit organizations, as these organizations can use these patterns and trends to design their strategies or introduce new programs, new products, processes or services. Social media mining uses a range of basic concepts from computer science, data mining, machine learning and statistics. Social media miners develop algorithms suitable for investigating massive files of social media data. Social media mining is based on theories and methodologies from social network analysis, network science, sociology, ethnography, optimization and mathematics. It encompasses the tools to formally represent, measure and model meaningful patterns from large-scale social media data. [1] In the 2010s, major corporations, governments and not-for-profit organizations engaged in social media mining to obtain data about customers, clients and citizens.

```
!pip install nltk

Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.4.0)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2023.12.25)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-
packages (from nltk) (4.66.2)
```

```
import nltk
nltk.download('punkt')
nltk.download('wordnet')
nltk.download('stopwords')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
True
```

Para 1

```
para1 = "You are musicophile,” One who loves music”. In your
interactions with your MCA classmates you found another
musicophile ,”Ameyatma”(classmate) and become friends. Now you want to
flaunt your programming skills and love for music to your friend.
Write a class called “showoff” which has a instance variable
“song”(String datatype) and a method called “unscramble”. The method
unscrambles the words in the string “song” and prints it. The variable
“song” is a stanza from a song but is strange. It has words and
numbers in between them, you need to order the words depending on the
numbers and print the correct stanza. Consider the following example
and run your program for the given two testcases. Create an object of
the class and call the method."
```

TOKENIZATION

```
tokens1 = nltk.word_tokenize(para1)
tokens1
```

```
['You',
 'are',
 'musicophile',
 ',',
 '”',
 'One',
 'who',
 'loves',
 'music',
 '”',
 '.',
 'In',
 'your',
 'interactions',
 'with',
 'your',
 'MCA',
 'classmates',
 'you',
 'found',
 'another',
 'musicophile',
 ',',
 '”',
 'Ameyatma',
 '”',
 '(',
 'classmate',
 ')',
 'and',
 'become',
 'friends',
 '.',
 'Now',
 'you',
 'want',
 'to',
 'flaunt',
 'your',
 'programming',
 'skills',
 'and',
 'love',
 'for',
 'music',
 'to',
 'your',
 'friend',
 '.',
 'Write',
 'a',
 'class',
 'called',
 '“',
 'showoff',
 '”',
 'which',
 'has',
 'a',
 'instance',
 'variable',
 '“',
 'song',
 '”',
 '(',
 'String',
 'datatype',
 ')',
 'and',
 'a',
 'method',
 'called',
 '“',
 'unscramble',
 '”',
 '.',
 'The',
 'method',
 'unscrambles',
 'the',
 'words',
 'in',
 'the',
 'string',
 '“',
 'song',
 '”',
 'and',
 'prints',
 'it',
 '.',
 'The',
 'variable',
 '“',
 'song',
 '”',
 'is',
 'a',
 'stanza',
 'from',
 'a',
 'song',
 'but',
 'is',
 'strange',
 '.',
 'It',
 'has',
 'words',
 'and',
 'numbers',
 'in',
 'between',
 'them',
 ',',
 'you',
 'need',
 'to',
 'order',
 'the',
 'words',
 'depending',
 'on',
 'the',
 'numbers',
 'and',
 'print',
 'the',
 'correct',
 'stanza',
 '.',
 'Consider',
 'the',
 'following',
 'example',
 'and',
 'run',
 'your',
 'program',
 'for',
 'the',
 'given',
 'two',
 'testcases',
 '.',
 'Create',
 'an',
 'object',
 'of',
 'the',
 'class',
 'and',
 'call',
 'the',
 'method',
 '.']
```

Stop Words

```
import nltk
from nltk.corpus import stopwords
```

```python
#print(stopwords.words('english'))

stop_words = set(stopwords.words('english'))

filtered_sentence = []

for w in tokens1:
    if w not in stop_words:
        filtered_sentence.append(w)

print("Actual Token Words\n")
print("Length: ", len(tokens1))
print(tokens1)
print("\nToken Words After Removing Stopwords\n")
print("Length: ", len(filtered_sentence))
print(filtered_sentence)
```

```
Actual Token Words

Length:  156
['You', 'are', 'musicophile', ',', '”', 'One', 'who', 'loves',
'music', '”', '.', 'In', 'your', 'interactions', 'with', 'your',
'MCA', 'classmates', 'you', 'found', 'another', 'musicophile', ',',
'”', 'Ameyatma', '”', '(', 'classmate', ')', 'and', 'become',
'friends', '.', 'Now', 'you', 'want', 'to', 'flaunt', 'your',
'programming', 'skills', 'and', 'love', 'for', 'music', 'to', 'your',
'friend', '.', 'Write', 'a', 'class', 'called', '“', 'showoff', '”',
'which', 'has', 'a', 'instance', 'variable', '“', 'song', '”', '(',
'String', 'datatype', ')', 'and', 'a', 'method', 'called', '“',
'unscramble', '”', '.', 'The', 'method', 'unscrambles', 'the',
'words', 'in', 'the', 'string', '“', 'song', '”', 'and', 'prints',
'it', '.', 'The', 'variable', '“', 'song', '”', 'is', 'a', 'stanza',
'from', 'a', 'song', 'but', 'is', 'strange', '.', 'It', 'has',
'words', 'and', 'numbers', 'in', 'between', 'them', ',', 'you',
'need', 'to', 'order', 'the', 'words', 'depending', 'on', 'the',
'numbers', 'and', 'print', 'the', 'correct', 'stanza', '.',
'Consider', 'the', 'following', 'example', 'and', 'run', 'your',
'program', 'for', 'the', 'given', 'two', 'testcases', '.', 'Create',
'an', 'object', 'of', 'the', 'class', 'and', 'call', 'the', 'method',
'.']

Token Words After Removing Stopwords

Length:  103
['You', 'musicophile', ',', '”', 'One', 'loves', 'music', '”', '.',
'In', 'interactions', 'MCA', 'classmates', 'found', 'another',
'musicophile', ',', '”', 'Ameyatma', '”', '(', 'classmate', ')',
'become', 'friends', '.', 'Now', 'want', 'flaunt', 'programming',
'skills', 'love', 'music', 'friend', '.', 'Write', 'class', 'called',
'“', 'showoff', '”', 'instance', 'variable', '“', 'song', '”', '(',
'String', 'datatype', ')', 'method', 'called', '“', 'unscramble', '”',
'.', 'The', 'method', 'unscrambles', 'words', 'string', '“', 'song',
'”', 'prints', '.', 'The', 'variable', '“', 'song', '”', 'stanza',
'song', 'strange', '.', 'It', 'words', 'numbers', ',', 'need',
'order', 'words', 'depending', 'numbers', 'print', 'correct',
'stanza', '.', 'Consider', 'following', 'example', 'run', 'program',
'given', 'two', 'testcases', '.', 'Create', 'object', 'class', 'call',
'method', '.']
```

Stemming Using Porter Method

```
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize

ps = PorterStemmer()

poster_sentence = []

for w in filtered_sentence:
    poster_sentence.append(ps.stem(w))
    print(w, " : ", ps.stem(w))

print("\n\n",poster_sentence)
print("\nLength or rather number of words ",len(poster_sentence))

You  :  you
musicophile  :  musicophil
‚  :  ‚
„  :  „
One  :  one
loves  :  love
music  :  music
”  :  ”
.  :  .
In  :  in
interactions  :  interact
MCA  :  mca
classmates  :  classmat
found  :  found
another  :  anoth
musicophile  :  musicophil
‚  :  ‚
„  :  „
Ameyatma  :  ameyatma
”  :  ”
(  :  (
classmate  :  classmat
)  :  )
become  :  becom
```

```
friends  :  friend
.  :  .
Now  :  now
want  :  want
flaunt  :  flaunt
programming  :  program
skills  :  skill
love  :  love
music  :  music
friend  :  friend
.  :  .
Write  :  write
class  :  class
called  :  call
“  :  “
showoff  :  showoff
”  :  ”
instance  :  instanc
variable  :  variabl
“  :  “
song  :  song
”  :  ”
(  :  (
String  :  string
datatype  :  datatyp
)  :  )
method  :  method
called  :  call
“  :  “
unscramble  :  unscrambl
”  :  ”
.  :  .
The  :  the
method  :  method
unscrambles  :  unscrambl
words  :  word
string  :  string
“  :  “
song  :  song
”  :  ”
prints  :  print
.  :  .
The  :  the
variable  :  variabl
“  :  “
song  :  song
”  :  ”
stanza  :  stanza
song  :  song
```

```
strange  :  strang
.  :  .
It  :  it
words  :  word
numbers  :  number
,  :  ,
need  :  need
order  :  order
words  :  word
depending  :  depend
numbers  :  number
print  :  print
correct  :  correct
stanza  :  stanza
.  :  .
Consider  :  consid
following  :  follow
example  :  exampl
run  :  run
program  :  program
given  :  given
two  :  two
testcases  :  testcas
.  :  .
Create  :  creat
object  :  object
class  :  class
call  :  call
method  :  method
.  :  .


 ['you', 'musicophil', ',', '”', 'one', 'love', 'music', '”', '.',
'in', 'interact', 'mca', 'classmat', 'found', 'anoth', 'musicophil',
',', '”', 'ameyatma', '”', '(', 'classmat', ')', 'becom', 'friend',
'.', 'now', 'want', 'flaunt', 'program', 'skill', 'love', 'music',
'friend', '.', 'write', 'class', 'call', '“', 'showoff', '”',
'instanc', 'variabl', '“', 'song', '”', '(', 'string', 'datatyp', ')',
'method', 'call', '“', 'unscrambl', '”', '.', 'the', 'method',
'unscrambl', 'word', 'string', '“', 'song', '”', 'print', '.', 'the',
'variabl', '“', 'song', '”', 'stanza', 'song', 'strang', '.', 'it',
'word', 'number', ',', 'need', 'order', 'word', 'depend', 'number',
'print', 'correct', 'stanza', '.', 'consid', 'follow', 'exampl',
'run', 'program', 'given', 'two', 'testcas', '.', 'creat', 'object',
'class', 'call', 'method', '.']

Length or rather number of words  103
```

Lemmatization

```python
from nltk.stem import WordNetLemmatizer

lemmatizer = WordNetLemmatizer()
lemmatizer_sentence = []

for w in filtered_sentence:
    lemmatizer_sentence.append(lemmatizer.lemmatize(w))
    print(w, " : ", lemmatizer.lemmatize(w))

#lemmatizer_tokens = (lemmatizer.lemmatize(token) for token in
filtered_sentence)
```

```
You  :  You
musicophile  :  musicophile
,  :  ,
”  :  ”
One  :  One
loves  :  love
music  :  music
”  :  ”
.  :  .
In  :  In
interactions  :  interaction
MCA  :  MCA
classmates  :  classmate
found  :  found
another  :  another
musicophile  :  musicophile
,  :  ,
”  :  ”
Ameyatma  :  Ameyatma
”  :  ”
(  :  (
classmate  :  classmate
)  :  )
become  :  become
friends  :  friend
.  :  .
Now  :  Now
want  :  want
flaunt  :  flaunt
programming  :  programming
skills  :  skill
love  :  love
music  :  music
friend  :  friend
.  :  .
Write  :  Write
class  :  class
called  :  called
```

```
“  :  “
showoff  :  showoff
”  :  ”
instance  :  instance
variable  :  variable
“  :  “
song  :  song
”  :  ”
(  :  (
String  :  String
datatype  :  datatype
)  :  )
method  :  method
called  :  called
“  :  “
unscramble  :  unscramble
”  :  ”
.  :  .
The  :  The
method  :  method
unscrambles  :  unscrambles
words  :  word
string  :  string
“  :  “
song  :  song
”  :  ”
prints  :  print
.  :  .
The  :  The
variable  :  variable
“  :  “
song  :  song
”  :  ”
stanza  :  stanza
song  :  song
strange  :  strange
.  :  .
It  :  It
words  :  word
numbers  :  number
,  :  ,
need  :  need
order  :  order
words  :  word
depending  :  depending
numbers  :  number
print  :  print
correct  :  correct
stanza  :  stanza
```

```
.  :  .
Consider  :  Consider
following  :  following
example  :  example
run  :  run
program  :  program
given  :  given
two  :  two
testcases  :  testcases
.  :  .
Create  :  Create
object  :  object
class  :  class
call  :  call
method  :  method
.  :  .
```

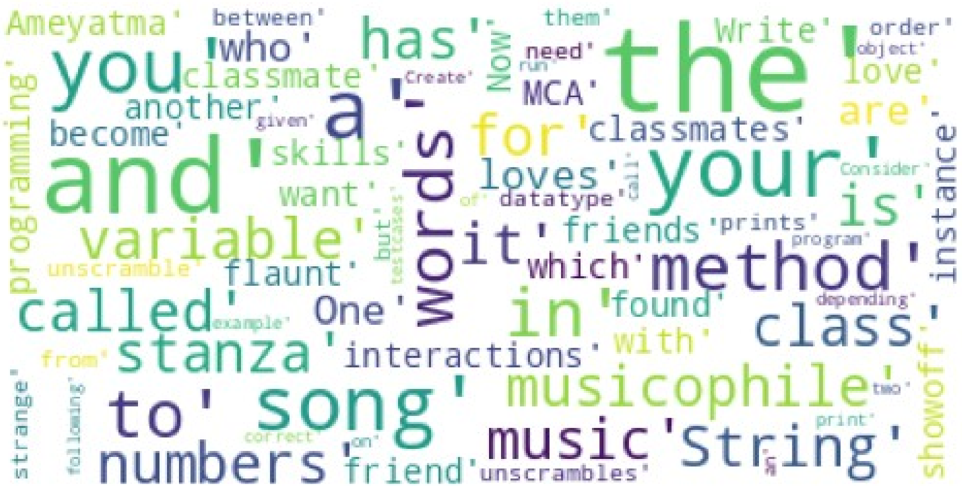
Word Cloud For Original Para

```
#Word cloud
from os import path
from PIL import Image
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
import matplotlib.pyplot as plt

wordcloud1 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(tokens1))

print(wordcloud1)
fig = plt.figure(1)
plt.imshow(wordcloud1)
plt.axis('off')
plt.show()
fig.savefig("word1.png", dpi=900)
```

```
<wordcloud.wordcloud.WordCloud object at 0x7b9ed3c7c6a0>
```



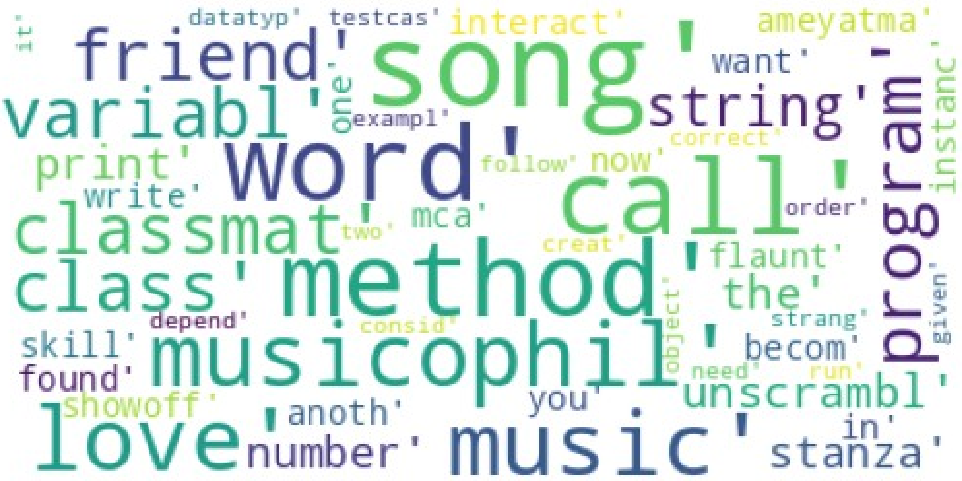
Word Cloud After Stemming

```
#Word cloud
from os import path
from PIL import Image
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
import matplotlib.pyplot as plt

wordcloud2 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(poster_sentence))

print(wordcloud2)
fig = plt.figure(1)
plt.imshow(wordcloud2)
plt.axis('off')
plt.show()
fig.savefig("word2.png", dpi=900)
```

```
<wordcloud.wordcloud.WordCloud object at 0x7b9e8c57b610>
```

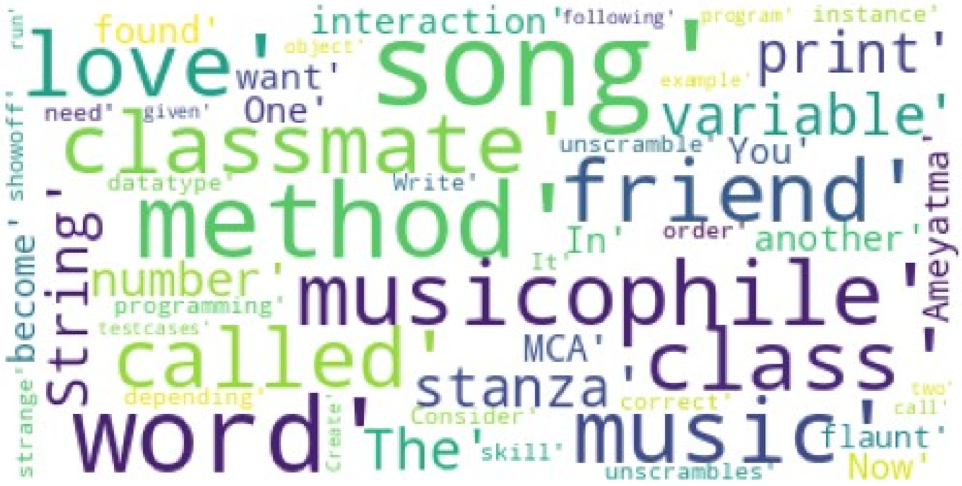


Word Cloud After Lemmatization

```
wordcloud3 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(lemmatizer_sentence))

print(wordcloud3)
fig = plt.figure(1)
plt.imshow(wordcloud3)
plt.axis('off')
plt.show()
fig.savefig("word3.png", dpi=900)
```

```
<wordcloud.wordcloud.WordCloud object at 0x7b9e8c4e00d0>
```



PARA 2

```
para2 = "Social media mining is the process of obtaining big data from
user-generated content on social media sites and mobile apps in order
to extract actionable patterns, form conclusions about users, and act
upon the information, often for the purpose of advertising to users or
conducting research. The term is an analogy to the resource extraction
process of mining for rare minerals. Resource extraction mining
requires mining companies to shift through vast quantities of raw ore
to find the precious minerals; likewise, social media mining requires
human data analysts and automated software programs to shift through
massive amounts of raw social media data in order to discern patterns
and trends relating to social media usage, online behaviours, sharing
of content, connections between individuals, online buying behaviour,
and more. These patterns and trends are of interest to companies,
governments and not-for-profit organizations, as these organizations
can use these patterns and trends to design their strategies or
introduce new programs, new products, processes or services. Social
media mining uses a range of basic concepts from computer science,
data mining, machine learning and statistics. Social media miners
develop algorithms suitable for investigating massive files of social
media data. Social media mining is based on theories and methodologies
from social network analysis, network science, sociology, ethnography,
optimization and mathematics. It encompasses the tools to formally
represent, measure and model meaningful patterns from large-scale
social media data. [1] In the 2010s, major corporations, governments
and not-for-profit organizations engaged in social media mining to
obtain data about customers, clients and citizens."
```

Tokenization

```
tokens2 = nltk.word_tokenize(para2)
tokens2
```

```
['Social',
 'media',
 'mining',
 'is',
 'the',
 'process',
 'of',
 'obtaining',
 'big',
 'data',
 'from',
 'user-generated',
 'content',
 'on',
 'social',
 'media',
 'sites',
 'and',
 'mobile',
 'apps',
 'in',
 'order',
 'to',
 'extract',
 'actionable',
 'patterns',
 ',',
 'form',
 'conclusions',
 'about',
 'users',
 ',',
 'and',
 'act',
 'upon',
 'the',
 'information',
 ',',
 'often',
 'for',
 'the',
 'purpose',
 'of',
 'advertising',
 'to',
 'users',
 'or',
 'conducting',
 'research',
 '.',
 'The',
 'term',
 'is',
 'an',
 'analogy',
 'to',
 'the',
 'resource',
 'extraction',
 'process',
 'of',
 'mining',
 'for',
 'rare',
 'minerals',
 '.',
 'Resource',
 'extraction',
 'mining',
 'requires',
 'mining',
 'companies',
 'to',
 'shift',
 'through',
 'vast',
 'quantities',
 'of',
 'raw',
 'ore',
 'to',
 'find',
 'the',
 'precious',
 'minerals',
 ';',
 'likewise',
 ',',
 'social',
 'media',
 'mining',
 'requires',
 'human',
 'data',
 'analysts',
 'and',
 'automated',
 'software',
 'programs',
 'to',
 'shift',
 'through',
 'massive',
 'amounts',
 'of',
 'raw',
 'social',
 'media',
 'data',
 'in',
 'order',
 'to',
 'discern',
 'patterns',
 'and',
 'trends',
 'relating',
 'to',
 'social',
 'media',
 'usage',
 ',',
 'online',
 'behaviours',
 ',',
 'sharing',
 'of',
 'content',
 ',',
 'connections',
 'between',
 'individuals',
 ',',
 'online',
 'buying',
 'behaviour',
 ',',
 'and',
 'more',
 '.',
 'These',
 'patterns',
 'and',
 'trends',
 'are',
 'of',
 'interest',
 'to',
 'companies',
 ',',
 'governments',
 'and',
 'not-for-profit',
 'organizations',
 ',',
 'as',
 'these',
 'organizations',
 'can',
 'use',
 'these',
 'patterns',
 'and',
 'trends',
 'to',
 'design',
 'their',
 'strategies',
 'or',
 'introduce',
 'new',
 'programs',
 ',',
 'new',
 'products',
 ',',
 'processes',
 'or',
 'services',
 '.',
 'Social',
 'media',
 'mining',
 'uses',
 'a',
 'range',
 'of',
 'basic',
 'concepts',
 'from',
 'computer',
 'science',
 ',',
 'data',
 'mining',
 ',',
 'machine',
 'learning',
 'and',
 'statistics',
 '.',
 'Social',
 'media',
 'miners',
 'develop',
 'algorithms',
 'suitable',
 'for',
 'investigating',
 'massive',
 'files',
 'of',
 'social',
 'media',
 'data',
 '.',
 'Social',
 'media',
 'mining',
 'is',
 'based',
 'on',
 'theories',
 'and',
 'methodologies',
 'from',
 'social',
 'network',
 'analysis',
 ',',
 'network',
 'science',
 ',',
 'sociology',
 ',',
 'ethnography',
 ',',
 'optimization',
 'and',
 'mathematics',
 '.',
 'It',
 'encompasses',
 'the',
 'tools',
 'to',
 'formally',
 'represent',
 ',',
 'measure',
 'and',
 'model',
 'meaningful',
 'patterns',
 'from',
 'large-scale',
 'social',
 'media',
 'data',
 '.',
 '[',
 '1',
 ']',
 'In',
 'the',
 '2010s',
 ',',
 'major',
 'corporations',
 ',',
 'governments',
 'and',
 'not-for-profit',
 'organizations',
 'engaged',
 'in',
 'social',
 'media',
 'mining',
 'to',
 'obtain',
 'data',
 'about',
 'customers',
 ',',
 'clients',
 'and',
 'citizens',
 '.']
```

Original Word Cloud

```
#Word cloud
wordcloud4 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(tokens2))

print(wordcloud4)
fig = plt.figure(1)
plt.imshow(wordcloud4)
plt.axis('off')
plt.show()
fig.savefig("word4.png", dpi=900)
```

```
<wordcloud.wordcloud.WordCloud object at 0x7b9eaa190640>
```

mining'
and'
to
data
the'
of'
social'
media'
patterns'
these'
from'
in'
for'
trends'
is'
or'
process'
minerals'
programs'
requires'
resource'
science'
content'
companies'
organizations'
extraction'
massive'
order'
governments'
profit'
users'
about'
information'
online'
through'
network'
obtaining'
mobile'
extract'
apps'
generated'
purpose'
sites'
conclusions'
actionable'
raw'
big'
shift'
act'
on'
upon'
often'
new'
user'
form'
social'
advertising'
conducting'
analogy'
analysts'
individuals'
sharing'
design'
find'
precious'
vast'

Removal Of Stop Words

```
import nltk
from nltk.corpus import stopwords

#print(stopwords.words('english'))

stop_words2 = set(stopwords.words('english'))

filtered_sentence2 = []

for w in tokens2:
    if w not in stop_words2:
        filtered_sentence2.append(w)

print("Actual Token Words\n")
print("Length:" ,len(tokens2) )
print(tokens2)
print("\nToken Words After Removing Stopwords\n")
print("Length: ", len(filtered_sentence2))
print(filtered_sentence2)
```

```
Actual Token Words

Length: 289
['Social', 'media', 'mining', 'is', 'the', 'process', 'of',
'obtaining', 'big', 'data', 'from', 'user-generated', 'content', 'on',
'social', 'media', 'sites', 'and', 'mobile', 'apps', 'in', 'order',
'to', 'extract', 'actionable', 'patterns', ',', 'form', 'conclusions',
'about', 'users', ',', 'and', 'act', 'upon', 'the', 'information',
',', 'often', 'for', 'the', 'purpose', 'of', 'advertising', 'to',
'users', 'or', 'conducting', 'research', '.', 'The', 'term', 'is',
'an', 'analogy', 'to', 'the', 'resource', 'extraction', 'process',
'of', 'mining', 'for', 'rare', 'minerals', '.', 'Resource',
'extraction', 'mining', 'requires', 'mining', 'companies', 'to',
'shift', 'through', 'vast', 'quantities', 'of', 'raw', 'ore', 'to',
'find', 'the', 'precious', 'minerals', ';', 'likewise', ',', 'social',
'media', 'mining', 'requires', 'human', 'data', 'analysts', 'and',
'automated', 'software', 'programs', 'to', 'shift', 'through',
'massive', 'amounts', 'of', 'raw', 'social', 'media', 'data', 'in',
'order', 'to', 'discern', 'patterns', 'and', 'trends', 'relating',
'to', 'social', 'media', 'usage', ',', 'online', 'behaviours', ',',
'sharing', 'of', 'content', ',', 'connections', 'between',
'individuals', ',', 'online', 'buying', 'behaviour', ',', 'and',
'more', '.', 'These', 'patterns', 'and', 'trends', 'are', 'of',
'interest', 'to', 'companies', ',', 'governments', 'and', 'not-for-
profit', 'organizations', ',', 'as', 'these', 'organizations', 'can',
'use', 'these', 'patterns', 'and', 'trends', 'to', 'design', 'their',
'strategies', 'or', 'introduce', 'new', 'programs', ',', 'new',
'products', ',', 'processes', 'or', 'services', '.', 'Social',
'media', 'mining', 'uses', 'a', 'range', 'of', 'basic', 'concepts',
'from', 'computer', 'science', ',', 'data', 'mining', ',', 'machine',
'learning', 'and', 'statistics', '.', 'Social', 'media', 'miners',
'develop', 'algorithms', 'suitable', 'for', 'investigating',
'massive', 'files', 'of', 'social', 'media', 'data', '.', 'Social',
'media', 'mining', 'is', 'based', 'on', 'theories', 'and',
'methodologies', 'from', 'social', 'network', 'analysis', ',',
'network', 'science', ',', 'sociology', ',', 'ethnography', ',',
'optimization', 'and', 'mathematics', '.', 'It', 'encompasses', 'the',
'tools', 'to', 'formally', 'represent', ',', 'measure', 'and',
'model', 'meaningful', 'patterns', 'from', 'large-scale', 'social',
'media', 'data', '.', '[', '1', ']', 'In', 'the', '2010s', ',',
'major', 'corporations', ',', 'governments', 'and', 'not-for-profit',
'organizations', 'engaged', 'in', 'social', 'media', 'mining', 'to',
'obtain', 'data', 'about', 'customers', ',', 'clients', 'and',
'citizens', '.']

Token Words After Removing Stopwords

Length:  215
['Social', 'media', 'mining', 'process', 'obtaining', 'big', 'data',
'user-generated', 'content', 'social', 'media', 'sites', 'mobile',
'apps', 'order', 'extract', 'actionable', 'patterns', ',', 'form',
'conclusions', 'users', ',', 'act', 'upon', 'information', ',',
'often', 'purpose', 'advertising', 'users', 'conducting', 'research',
'.', 'The', 'term', 'analogy', 'resource', 'extraction', 'process',
'mining', 'rare', 'minerals', '.', 'Resource', 'extraction', 'mining',
'requires', 'mining', 'companies', 'shift', 'vast', 'quantities',
'raw', 'ore', 'find', 'precious', 'minerals', ';', 'likewise', ',',
'social', 'media', 'mining', 'requires', 'human', 'data', 'analysts',
'automated', 'software', 'programs', 'shift', 'massive', 'amounts',
'raw', 'social', 'media', 'data', 'order', 'discern', 'patterns',
'trends', 'relating', 'social', 'media', 'usage', ',', 'online',
'behaviours', ',', 'sharing', 'content', ',', 'connections',
'individuals', ',', 'online', 'buying', 'behaviour', ',', '.',
'These', 'patterns', 'trends', 'interest', 'companies', ',',
'governments', 'not-for-profit', 'organizations', ',',
'organizations', 'use', 'patterns', 'trends', 'design', 'strategies',
'introduce', 'new', 'programs', ',', 'new', 'products', ',',
'processes', 'services', '.', 'Social', 'media', 'mining', 'uses',
'range', 'basic', 'concepts', 'computer', 'science', ',', 'data',
'mining', ',', 'machine', 'learning', 'statistics', '.', 'Social',
'media', 'miners', 'develop', 'algorithms', 'suitable',
'investigating', 'massive', 'files', 'social', 'media', 'data', '.',
'Social', 'media', 'mining', 'based', 'theories', 'methodologies',
'social', 'network', 'analysis', ',', 'network', 'science', ',',
'sociology', ',', 'ethnography', ',', 'optimization', 'mathematics',
'.', 'It', 'encompasses', 'tools', 'formally', 'represent', ',',
'measure', 'model', 'meaningful', 'patterns', 'large-scale', 'social',
'media', 'data', '.', '[', '1', ']', 'In', '2010s', ',', 'major',
'corporations', ',', 'governments', 'not-for-profit', 'organizations',
'engaged', 'social', 'media', 'mining', 'obtain', 'data', 'customers',
',', 'clients', 'citizens', '.']
```

Stemming

```
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize

ps2 = PorterStemmer()

poster_sentence2 = []

for w in filtered_sentence2:
    poster_sentence2.append(ps2.stem(w))
    print(w, " : ", ps2.stem(w))

print("\n\n",poster_sentence2)
print("\nLength or rather number of words ",len(poster_sentence2))
```

```
Social  :  social
media  :  media
mining  :  mine
process  :  process
obtaining  :  obtain
big  :  big
data  :  data
user-generated  :  user-gener
content  :  content
social  :  social
media  :  media
sites  :  site
mobile  :  mobil
```

```
apps  :  app
order  :  order
extract  :  extract
actionable  :  action
patterns  :  pattern
,  :  ,
form  :  form
conclusions  :  conclus
users  :  user
,  :  ,
act  :  act
upon  :  upon
information  :  inform
,  :  ,
often  :  often
purpose  :  purpos
advertising  :  advertis
users  :  user
conducting  :  conduct
research  :  research
.  :  .
The  :  the
term  :  term
analogy  :  analog
resource  :  resourc
extraction  :  extract
process  :  process
mining  :  mine
rare  :  rare
minerals  :  miner
.  :  .
Resource  :  resourc
extraction  :  extract
mining  :  mine
requires  :  requir
mining  :  mine
companies  :  compani
shift  :  shift
vast  :  vast
quantities  :  quantiti
raw  :  raw
ore  :  ore
find  :  find
precious  :  preciou
minerals  :  miner
;  :  ;
likewise  :  likewis
,  :  ,
social  :  social
```

```
media  :  media
mining  :  mine
requires  :  requir
human  :  human
data  :  data
analysts  :  analyst
automated  :  autom
software  :  softwar
programs  :  program
shift  :  shift
massive  :  massiv
amounts  :  amount
raw  :  raw
social  :  social
media  :  media
data  :  data
order  :  order
discern  :  discern
patterns  :  pattern
trends  :  trend
relating  :  relat
social  :  social
media  :  media
usage  :  usag
,  :  ,
online  :  onlin
behaviours  :  behaviour
,  :  ,
sharing  :  share
content  :  content
,  :  ,
connections  :  connect
individuals  :  individu
,  :  ,
online  :  onlin
buying  :  buy
behaviour  :  behaviour
,  :  ,
.  :  .
These  :  these
patterns  :  pattern
trends  :  trend
interest  :  interest
companies  :  compani
,  :  ,
governments  :  govern
not-for-profit  :  not-for-profit
organizations  :  organ
,  :  ,
organizations  :  organ
use  :  use
patterns  :  pattern
trends  :  trend
design  :  design
strategies  :  strategi
introduce  :  introduc
new  :  new
programs  :  program
,  :  ,
new  :  new
products  :  product
,  :  ,
processes  :  process
services  :  servic
.  :  .
Social  :  social
media  :  media
mining  :  mine
uses  :  use
range  :  rang
basic  :  basic
concepts  :  concept
computer  :  comput
science  :  scienc
,  :  ,
data  :  data
mining  :  mine
,  :  ,
machine  :  machin
learning  :  learn
statistics  :  statist
.  :  .
Social  :  social
media  :  media
miners  :  miner
develop  :  develop
algorithms  :  algorithm
suitable  :  suitabl
investigating  :  investig
massive  :  massiv
files  :  file
social  :  social
media  :  media
data  :  data
.  :  .
Social  :  social
media  :  media
mining  :  mine
```

```
based  :  base
theories  :  theori
methodologies  :  methodolog
social  :  social
network  :  network
analysis  :  analysi
,  :  ,
network  :  network
science  :  scienc
,  :  ,
sociology  :  sociolog
,  :  ,
ethnography  :  ethnographi
,  :  ,
optimization  :  optim
mathematics  :  mathemat
.  :  .
It  :  it
encompasses  :  encompass
tools  :  tool
formally  :  formal
represent  :  repres
,  :  ,
measure  :  measur
model  :  model
meaningful  :  meaning
patterns  :  pattern
large-scale  :  large-scal
social  :  social
media  :  media
data  :  data
.  :  .
[  :  [
1  :  1
]  :  ]
In  :  in
2010s  :  2010
,  :  ,
major  :  major
corporations  :  corpor
,  :  ,
governments  :  govern
not-for-profit  :  not-for-profit
organizations  :  organ
engaged  :  engag
social  :  social
media  :  media
mining  :  mine
obtain  :  obtain
```

```
data  :  data
customers  :  custom
,  :  ,
clients  :  client
citizens  :  citizen
.  :  .


 ['social', 'media', 'mine', 'process', 'obtain', 'big', 'data',
'user-gener', 'content', 'social', 'media', 'site', 'mobil', 'app',
'order', 'extract', 'action', 'pattern', ',', 'form', 'conclus',
'user', ',', 'act', 'upon', 'inform', ',', 'often', 'purpos',
'advertis', 'user', 'conduct', 'research', '.', 'the', 'term',
'analog', 'resourc', 'extract', 'process', 'mine', 'rare', 'miner',
'.', 'resourc', 'extract', 'mine', 'requir', 'mine', 'compani',
'shift', 'vast', 'quantiti', 'raw', 'ore', 'find', 'preciou', 'miner',
';', 'likewis', ',', 'social', 'media', 'mine', 'requir', 'human',
'data', 'analyst', 'autom', 'softwar', 'program', 'shift', 'massiv',
'amount', 'raw', 'social', 'media', 'data', 'order', 'discern',
'pattern', 'trend', 'relat', 'social', 'media', 'usag', ',', 'onlin',
'behaviour', ',', 'share', 'content', ',', 'connect', 'individu', ',',
'onlin', 'buy', 'behaviour', ',', '.', 'these', 'pattern', 'trend',
'interest', 'compani', ',', 'govern', 'not-for-profit', 'organ', ',',
'organ', 'use', 'pattern', 'trend', 'design', 'strategi', 'introduc',
'new', 'program', ',', 'new', 'product', ',', 'process', 'servic',
'.', 'social', 'media', 'mine', 'use', 'rang', 'basic', 'concept',
'comput', 'scienc', ',', 'data', 'mine', ',', 'machin', 'learn',
'statist', '.', 'social', 'media', 'miner', 'develop', 'algorithm',
'suitabl', 'investig', 'massiv', 'file', 'social', 'media', 'data',
'.', 'social', 'media', 'mine', 'base', 'theori', 'methodolog',
'social', 'network', 'analysi', ',', 'network', 'scienc', ',',
'sociolog', ',', 'ethnographi', ',', 'optim', 'mathemat', '.', 'it',
'encompass', 'tool', 'formal', 'repres', ',', 'measur', 'model',
'meaning', 'pattern', 'large-scal', 'social', 'media', 'data', '.',
'[', '1', ']', 'in', '2010', ',', 'major', 'corpor', ',', 'govern',
'not-for-profit', 'organ', 'engag', 'social', 'media', 'mine',
'obtain', 'data', 'custom', ',', 'client', 'citizen', '.']

Length or rather number of words  215
```

Lemmatization

```
from nltk.stem import WordNetLemmatizer

lemmatizer2 = WordNetLemmatizer()
lemmatizer_sentence2 = []

for w in filtered_sentence2:
```

```python
    lemmatizer_sentence2.append(lemmatizer2.lemmatize(w))
    print(w, " : ", lemmatizer2.lemmatize(w))
```

```
Social  :  Social
media  :  medium
mining  :  mining
process  :  process
obtaining  :  obtaining
big  :  big
data  :  data
user-generated  :  user-generated
content  :  content
social  :  social
media  :  medium
sites  :  site
mobile  :  mobile
apps  :  apps
order  :  order
extract  :  extract
actionable  :  actionable
patterns  :  pattern
,  :  ,
form  :  form
conclusions  :  conclusion
users  :  user
,  :  ,
act  :  act
upon  :  upon
information  :  information
,  :  ,
often  :  often
purpose  :  purpose
advertising  :  advertising
users  :  user
conducting  :  conducting
research  :  research
.  :  .
The  :  The
term  :  term
analogy  :  analogy
resource  :  resource
extraction  :  extraction
process  :  process
mining  :  mining
rare  :  rare
minerals  :  mineral
.  :  .
Resource  :  Resource
extraction  :  extraction
mining  :  mining
```

```
requires  :  requires
mining  :  mining
companies  :  company
shift  :  shift
vast  :  vast
quantities  :  quantity
raw  :  raw
ore  :  ore
find  :  find
precious  :  precious
minerals  :  mineral
;  :  ;
likewise  :  likewise
,  :  ,
social  :  social
media  :  medium
mining  :  mining
requires  :  requires
human  :  human
data  :  data
analysts  :  analyst
automated  :  automated
software  :  software
programs  :  program
shift  :  shift
massive  :  massive
amounts  :  amount
raw  :  raw
social  :  social
media  :  medium
data  :  data
order  :  order
discern  :  discern
patterns  :  pattern
trends  :  trend
relating  :  relating
social  :  social
media  :  medium
usage  :  usage
,  :  ,
online  :  online
behaviours  :  behaviour
,  :  ,
sharing  :  sharing
content  :  content
,  :  ,
connections  :  connection
individuals  :  individual
,  :  ,
online  :  online
```

```
buying  :  buying
behaviour  :  behaviour
,  :  ,
.  :  .
These  :  These
patterns  :  pattern
trends  :  trend
interest  :  interest
companies  :  company
,  :  ,
governments  :  government
not-for-profit  :  not-for-profit
organizations  :  organization
,  :  ,
organizations  :  organization
use  :  use
patterns  :  pattern
trends  :  trend
design  :  design
strategies  :  strategy
introduce  :  introduce
new  :  new
programs  :  program
,  :  ,
new  :  new
products  :  product
,  :  ,
processes  :  process
services  :  service
.  :  .
Social  :  Social
media  :  medium
mining  :  mining
uses  :  us
range  :  range
basic  :  basic
concepts  :  concept
computer  :  computer
science  :  science
,  :  ,
data  :  data
mining  :  mining
,  :  ,
machine  :  machine
learning  :  learning
statistics  :  statistic
.  :  .
Social  :  Social
media  :  medium
miners  :  miner
```

```
develop  :  develop
algorithms  :  algorithm
suitable  :  suitable
investigating  :  investigating
massive  :  massive
files  :  file
social  :  social
media  :  medium
data  :  data
.  :  .
Social  :  Social
media  :  medium
mining  :  mining
based  :  based
theories  :  theory
methodologies  :  methodology
social  :  social
network  :  network
analysis  :  analysis
,  :  ,
network  :  network
science  :  science
,  :  ,
sociology  :  sociology
,  :  ,
ethnography  :  ethnography
,  :  ,
optimization  :  optimization
mathematics  :  mathematics
.  :  .
It  :  It
encompasses  :  encompasses
tools  :  tool
formally  :  formally
represent  :  represent
,  :  ,
measure  :  measure
model  :  model
meaningful  :  meaningful
patterns  :  pattern
large-scale  :  large-scale
social  :  social
media  :  medium
data  :  data
.  :  .
[  :  [
1  :  1
]  :  ]
In  :  In
2010s  :  2010s
```

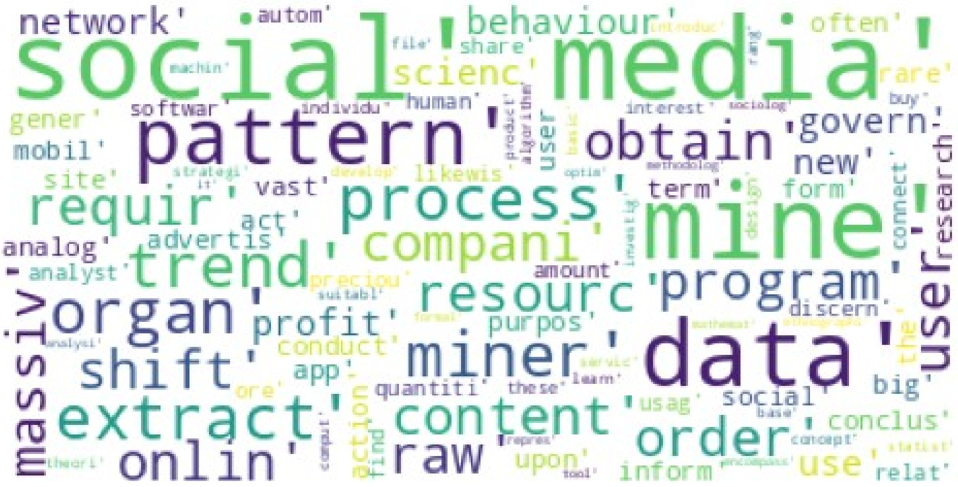
```
,  :  ,
major  :  major
corporations  :  corporation
,  :  ,
governments  :  government
not-for-profit  :  not-for-profit
organizations  :  organization
engaged  :  engaged
social  :  social
media  :  medium
mining  :  mining
obtain  :  obtain
data  :  data
customers  :  customer
,  :  ,
clients  :  client
citizens  :  citizen
.  :  .
```

Word Cloud after Stemming

```
wordcloud5 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(poster_sentence2))

print(wordcloud5)
fig = plt.figure(1)
plt.imshow(wordcloud5)
plt.axis('off')
plt.show()
fig.savefig("word5.png", dpi=900)
```

```
<wordcloud.wordcloud.WordCloud object at 0x7b9eaa192230>
```

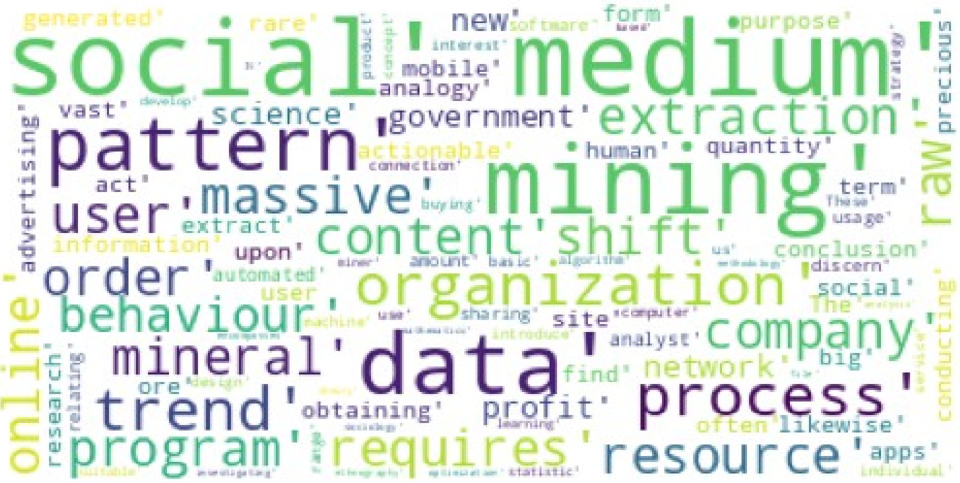


Word Cloud after Lemmatization

```
wordcloud6 = WordCloud(
                          background_color='white',
                          stopwords=stop_words,
                          max_words=100,
                          max_font_size=50,
                          random_state=42
                         ).generate(str(lemmatizer_sentence2))

print(wordcloud6)
fig = plt.figure(1)
plt.imshow(wordcloud6)
plt.axis('off')
plt.show()
fig.savefig("word6.png", dpi=900)

<wordcloud.wordcloud.WordCloud object at 0x7b9e8c4e1cf0>
```


generated'
rare'
new'
software'
form'
purpose'
social'
medium'
interest'
mobile'
analogy'
precious'
vast'
science'
government'
extraction'
pattern'
actionable'
human'
quantity'
connection'
advertising'
act'
massive'
mining'
term'
raw'
user'
buying'
usage'
information'
extract'
upon'
content'
shift'
conclusion'
order'
automated'
organization'
discern'
user'
social'
online'
behaviour'
sharing'
site'
company'
analyst'
conducting'
mineral'
data'
find'
network'
big'
research'
relating'
ore'
trend'
obtaining'
profit'
process'
often'
likewise'
program'
requires'
resource'
apps'
statistic'
individual'