Text   AI (ML, DL, NLP)   Date __/__/__

①
```
import  nltk  as  nlp
from  nltk.tokenize  import  sent_tokenize, word_tokenize
```

```
aText = " . . . . . . "
print ( sent_tokenize (aText))
```

②
```
from  nltk.text  import  Text
t = Text (aText)        // Tokens ko wapis text ma
```

③
```
t.count ("is")
t.vocab ()            // ( {'y': 3
                             ? : 2 } )
```

④
```
from  nltk.corpus  import  stopwords
```

```
stopwords.words ('english')
```

```
stopwords.fileids ()      // all languages in
                              stop words.
```

```
stopwords.raw ('arabic')
```
- no change
- as it is sb aye ga.

⑤
```
from  nltk.stem  import  PorterStemmer
 "     "     "      "     SnowballStemmer
 "     "     "      "     LancasterStemmer
```

```
ps = PorterStemmer ()
ps.stem (w)
```

Step ⇒   normalize   For - Ngrams ⇒ Word tokenize
          lemmatize   ngrams
          Stopwords.   Freq Dist

_____ Date ___/___/___

⑥ from   nltk   import   WordNetLemmatizer

    wnl = WordNetLemmatizer()
    wnl.lemmatize(w).

⑦ import   String
    p = String.punctuation

    from   textblob   import   TextBlob   // spellings ko
                                                   theek krnay ka
⑧   lia hta ga.
    $t = "\ ...\ "$
    $t_1 = TextBlob(t)$
    $t_c = t_1.correct()$

⑨ $g = nltk.ngrams(tokens, 1)$ for bigram use 2
    $gfD = nltk.FreqDist(g)$ for trigrams use 3
                              ↓
                          word tokens.

    for i in gfD.items():
        print(i)

    Time Series

⑩ data['Date'] = pd.to_datetime(data['Date'])

    data['Year'] = data['Date'].dt.year
    "   ['Month'] "   "   " .dt.month