Python for Data Science (LAB Session - 10)

Basics of NLP

Q-1)Explore NLP with example of The 20NewsGroup

In [1]:

```
from sklearn.datasets import fetch_20newsgroups
from nltk.tokenize import sent_tokenize, word_tokenize
#importing librery
```

In [2]:

```
data = fetch_20newsgroups() # assign value of dataset
a=data["data"]
df = a[0]
print(df) # print data of frist index
```

```
From: lerxst@wam.umd.edu (where's my thing)
Subject: WHAT car is this!?
Nntp-Posting-Host: rac3.wam.umd.edu
Organization: University of Maryland, College Park
Lines: 15

 I was wondering if anyone out there could enlighten me on this car I saw
the other day. It was a 2-door sports car, looked to be from the late 60s/
early 70s. It was called a Bricklin. The doors were really small. In additio
n,
the front bumper was separate from the rest of the body. This is
all I know. If anyone can tellme a model name, engine specs, years
of production, where this car is made, history, or whatever info you
have on this funky looking car, please e-mail.

Thanks,
- IL
   ---- brought to you by your neighborhood Lerxst ----
```

Q-A) Sentence Tokenization

In [3]:
```
print(sent_tokenize(df))
```

```
["From: lerxst@wam.umd.edu (where's my thing)\nSubject: WHAT car is this!?",
'Nntp-Posting-Host: rac3.wam.umd.edu\nOrganization: University of Maryland, C
ollege Park\nLines: 15\n\n I was wondering if anyone out there could enlighte
n me on this car I saw\nthe other day.', 'It was a 2-door sports car, looked
to be from the late 60s/\nearly 70s.', 'It was called a Bricklin.', 'The door
s were really small.', 'In addition,\nthe front bumper was separate from the
rest of the body.', 'This is \nall I know.', 'If anyone can tellme a model na
me, engine specs, years\nof production, where this car is made, history, or w
hatever info you\nhave on this funky looking car, please e-mail.', 'Thanks,\n
- IL\n   ---- brought to you by your neighborhood Lerxst ----']
```

Q-B) Word Tokenization

In [10]:
```
print(word_tokenize(df))
word_data = word_tokenize(df)
```

```
['From', ':', 'lerxst', '@', 'wam.umd.edu', '(', 'where', "'s", 'my', 'thin
g', ')', 'Subject', ':', 'WHAT', 'car', 'is', 'this', '!', '?', 'Nntp-Posting
-Host', ':', 'rac3.wam.umd.edu', 'Organization', ':', 'University', 'of', 'Ma
ryland', ',', 'College', 'Park', 'Lines', ':', '15', 'I', 'was', 'wondering',
'if', 'anyone', 'out', 'there', 'could', 'enlighten', 'me', 'on', 'this', 'ca
r', 'I', 'saw', 'the', 'other', 'day', '.', 'It', 'was', 'a', '2-door', 'spor
ts', 'car', ',', 'looked', 'to', 'be', 'from', 'the', 'late', '60s/', 'earl
y', '70s', '.', 'It', 'was', 'called', 'a', 'Bricklin', '.', 'The', 'doors',
'were', 'really', 'small', '.', 'In', 'addition', ',', 'the', 'front', 'bumpe
r', 'was', 'separate', 'from', 'the', 'rest', 'of', 'the', 'body', '.', 'Thi
s', 'is', 'all', 'I', 'know', '.', 'If', 'anyone', 'can', 'tellme', 'a', 'mod
el', 'name', ',', 'engine', 'specs', ',', 'years', 'of', 'production', ',',
'where', 'this', 'car', 'is', 'made', ',', 'history', ',', 'or', 'whatever',
'info', 'you', 'have', 'on', 'this', 'funky', 'looking', 'car', ',', 'pleas
e', 'e-mail', '.', 'Thanks', ',', '-', 'IL', '--', '--', 'brought', 'to', 'yo
u', 'by', 'your', 'neighborhood', 'Lerxst', '--', '--']
```

Q-C)Text Lemmatization

In [22]:

```
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
lemmatize_data = ""
list1 = []
list2 = []
for i in word_data:
    list2.append(i)
    l=lemmatizer.lemmatize(i)
    list1.append(l)

print("Lammatizer all list out word...\n")
for i in zip(list2,list1):
    print(i)
```

```
Lammatizer all list out word...

('From', 'From')
(':', ':')
('lerxst', 'lerxst')
('@', '@')
('wam.umd.edu', 'wam.umd.edu')
('(', '(')
('where', 'where')
("'s", "'s")
('my', 'my')
('thing', 'thing')
(')', ')')
('Subject', 'Subject')
(':', ':')
('WHAT', 'WHAT')
('car', 'car')
('is', 'is')
('this', 'this')
('!', '!')
('?', '?')
('Nntp-Posting-Host', 'Nntp-Posting-Host')
(':', ':')
('rac3.wam.umd.edu', 'rac3.wam.umd.edu')
('Organization', 'Organization')
(':', ':')
('University', 'University')
('of', 'of')
('Maryland', 'Maryland')
(',', ',')
('College', 'College')
('Park', 'Park')
('Lines', 'Lines')
(':', ':')
('15', '15')
('I', 'I')
('was', 'wa')
('wondering', 'wondering')
('if', 'if')
('anyone', 'anyone')
('out', 'out')
('there', 'there')
('could', 'could')
('enlighten', 'enlighten')
('me', 'me')
('on', 'on')
('this', 'this')
('car', 'car')
('I', 'I')
('saw', 'saw')
('the', 'the')
('other', 'other')
('day', 'day')
('.', '.')
('It', 'It')
('was', 'wa')
('a', 'a')
```

```
('2-door', '2-door')
('sports', 'sport')
('car', 'car')
(',', ',')
('looked', 'looked')
('to', 'to')
('be', 'be')
('from', 'from')
('the', 'the')
('late', 'late')
('60s/', '60s/')
('early', 'early')
('70s', '70')
('.', '.')
('It', 'It')
('was', 'wa')
('called', 'called')
('a', 'a')
('Bricklin', 'Bricklin')
('.', '.')
('The', 'The')
('doors', 'door')
('were', 'were')
('really', 'really')
('small', 'small')
('.', '.')
('In', 'In')
('addition', 'addition')
(',', ',')
('the', 'the')
('front', 'front')
('bumper', 'bumper')
('was', 'wa')
('separate', 'separate')
('from', 'from')
('the', 'the')
('rest', 'rest')
('of', 'of')
('the', 'the')
('body', 'body')
('.', '.')
('This', 'This')
('is', 'is')
('all', 'all')
('I', 'I')
('know', 'know')
('.', '.')
('If', 'If')
('anyone', 'anyone')
('can', 'can')
('tellme', 'tellme')
('a', 'a')
('model', 'model')
('name', 'name')
(',', ',')
('engine', 'engine')
('specs', 'spec')
```

```
(',', ',')
('years', 'year')
('of', 'of')
('production', 'production')
(',', ',')
('where', 'where')
('this', 'this')
('car', 'car')
('is', 'is')
('made', 'made')
(',', ',')
('history', 'history')
(',', ',')
('or', 'or')
('whatever', 'whatever')
('info', 'info')
('you', 'you')
('have', 'have')
('on', 'on')
('this', 'this')
('funky', 'funky')
('looking', 'looking')
('car', 'car')
(',', ',')
('please', 'please')
('e-mail', 'e-mail')
('.', '.')
('Thanks', 'Thanks')
(',', ',')
('-', '-')
('IL', 'IL')
('--', '--')
('--', '--')
('brought', 'brought')
('to', 'to')
('you', 'you')
('by', 'by')
('your', 'your')
('neighborhood', 'neighborhood')
('Lerxst', 'Lerxst')
('--', '--')
('--', '--')
```

In [15]:

```
lemmatizer = WordNetLemmatizer()
print("rocks :", lemmatizer.lemmatize("rocks"))
print("corpora :", lemmatizer.lemmatize("corpora"))
print("better :", lemmatizer.lemmatize("better", pos ="a"))
```

```
rocks : rock
corpora : corpus
better : good
```

In [17]:

```
sentence = ["This","sentence","was","transformed", "using", "WordNet", "Lemmatizer"]

lemmatizer = WordNetLemmatizer()

print (" ".join([lemmatizer.lemmatize(word) for word in sentence]))
```

```
This sentence wa transformed using WordNet Lemmatizer
```

Q-D)Stemming

In [27]:

```
from nltk.stem import PorterStemmer
stem = PorterStemmer()
res_str = ""
list1 = []
list2 = []
for i in word_data:
    list2.append(i)
    l=stem.stem(i)
    list1.append(l)

print("Stemming Text all list out word...\n")
for i in zip(list2,list1):
    print(i)
```

```
Stemming Text all list out word...

('From', 'from')
(':', ':')
('lerxst', 'lerxst')
('@', '@')
('wam.umd.edu', 'wam.umd.edu')
('(', '(')
('where', 'where')
("'s", "'s")
('my', 'my')
('thing', 'thing')
(')', ')')
('Subject', 'subject')
(':', ':')
('WHAT', 'what')
('car', 'car')
('is', 'is')
('this', 'thi')
('!', '!')
('?', '?')
('Nntp-Posting-Host', 'nntp-posting-host')
(':', ':')
('rac3.wam.umd.edu', 'rac3.wam.umd.edu')
('Organization', 'organ')
(':', ':')
('University', 'univers')
('of', 'of')
('Maryland', 'maryland')
(',', ',')
('College', 'colleg')
('Park', 'park')
('Lines', 'line')
(':', ':')
('15', '15')
('I', 'I')
('was', 'wa')
('wondering', 'wonder')
('if', 'if')
('anyone', 'anyon')
('out', 'out')
('there', 'there')
('could', 'could')
('enlighten', 'enlighten')
('me', 'me')
('on', 'on')
('this', 'thi')
('car', 'car')
('I', 'I')
('saw', 'saw')
('the', 'the')
('other', 'other')
('day', 'day')
('.', '.')
('It', 'It')
('was', 'wa')
('a', 'a')
```

```
('2-door', '2-door')
('sports', 'sport')
('car', 'car')
(',', ',')
('looked', 'look')
('to', 'to')
('be', 'be')
('from', 'from')
('the', 'the')
('late', 'late')
('60s/', '60s/')
('early', 'earli')
('70s', '70')
('.', '.')
('It', 'It')
('was', 'wa')
('called', 'call')
('a', 'a')
('Bricklin', 'bricklin')
('.', '.')
('The', 'the')
('doors', 'door')
('were', 'were')
('really', 'realli')
('small', 'small')
('.', '.')
('In', 'In')
('addition', 'addit')
(',', ',')
('the', 'the')
('front', 'front')
('bumper', 'bumper')
('was', 'wa')
('separate', 'separ')
('from', 'from')
('the', 'the')
('rest', 'rest')
('of', 'of')
('the', 'the')
('body', 'bodi')
('.', '.')
('This', 'thi')
('is', 'is')
('all', 'all')
('I', 'I')
('know', 'know')
('.', '.')
('If', 'If')
('anyone', 'anyon')
('can', 'can')
('tellme', 'tellm')
('a', 'a')
('model', 'model')
('name', 'name')
(',', ',')
('engine', 'engin')
('specs', 'spec')
```

```
(',', ',')
('years', 'year')
('of', 'of')
('production', 'product')
(',', ',')
('where', 'where')
('this', 'thi')
('car', 'car')
('is', 'is')
('made', 'made')
(',', ',')
('history', 'histori')
(',', ',')
('or', 'or')
('whatever', 'whatev')
('info', 'info')
('you', 'you')
('have', 'have')
('on', 'on')
('this', 'thi')
('funky', 'funki')
('looking', 'look')
('car', 'car')
(',', ',')
('please', 'pleas')
('e-mail', 'e-mail')
('.', '.')
('Thanks', 'thank')
(',', ',')
('-', '-')
('IL', 'IL')
('--', '--')
('--', '--')
('brought', 'brought')
('to', 'to')
('you', 'you')
('by', 'by')
('your', 'your')
('neighborhood', 'neighborhood')
('Lerxst', 'lerxst')
('--', '--')
('--', '--')
```

Q-E)Stop Words

In [31]:

```
from nltk.corpus import stopwords
word=stopwords.words('english')
word_data=word_tokenize(df)
remove_stop_str = ""
list1 = []
for i in word_data:
    if( i not in word):
        lst.append(i)
for i in lst:
    remove_stop_str += "".join(i)
print(remove_stop_str)
```

```
From:lerxst@wam.umd.edu(where'smything)Subject:WHATcaristhis!?Nntp-Posting-Ho
st:rac3.wam.umd.eduOrganization:UniversityofMaryland,CollegeParkLines:15Iwawo
nderingifanyoneouttherecouldenlightenmeonthiscarIsawtheotherday.Itwaa2-doorsp
ortcar,lookedtobefromthelate60s/early70.ItwacalledaBricklin.Thedoorwerereally
small.Inaddition,thefrontbumperwaseparatefromtherestofthebody.ThisisallIknow.
Ifanyonecantellmeamodelname,enginespec,yearofproduction,wherethiscarismade,hi
story,orwhateverinfoyouhaveonthisfunkylookingcar,pleasee-mail.Thanks,-IL----b
roughttoyoubyyourneighborhoodLerxst----From:lerxst@wam.umd.edu('sthing)Subjec
t:WHATcar!?Nntp-Posting-Host:rac3.wam.umd.eduOrganization:UniversityMaryland,
CollegeParkLines:15IwonderinganyonecouldenlightencarIsawday.It2-doorsportsca
r,lookedlate60s/early70s.ItcalledBricklin.Thedoorsreallysmall.Inaddition,fron
tbumperseparaterestbody.ThisIknow.Ifanyonetellmemodelname,enginespecs,yearspr
oduction,carmade,history,whateverinfofunkylookingcar,pleasee-mail.Thanks,-IL-
---broughtneighborhoodLerxst----From:lerxst@wam.umd.edu('sthing)Subject:WHATc
ar!?Nntp-Posting-Host:rac3.wam.umd.eduOrganization:UniversityMaryland,College
ParkLines:15IwonderinganyonecouldenlightencarIsawday.It2-doorsportscar,looked
late60s/early70s.ItcalledBricklin.Thedoorsreallysmall.Inaddition,frontbumpers
eparaterestbody.ThisIknow.Ifanyonetellmemodelname,enginespecs,yearsproductio
n,carmade,history,whateverinfofunkylookingcar,pleasee-mail.Thanks,-IL----brou
ghtneighborhoodLerxst----
```

In [57]:

```
data = "All work and no play makes jack dull boy. All work and no play makes j
ack a dull boy."
stopWords = set(stopwords.words('english'))
words = word_tokenize(data)
wordsFiltered = []

for w in words:
    if w not in stopWords:
        wordsFiltered.append(w)
print(wordsFiltered)
```

```
['All', 'work', 'play', 'makes', 'jack', 'dull', 'boy', '.', 'All', 'work',
'play', 'makes', 'jack', 'dull', 'boy', '.']
```

Q-F)RegEx

In [42]:

```
import re

print("Before Remove The punctuation in string\n-->")
print(df)
print("------------------------------------------------")
res = re.sub(r'[^\w\s]', '', df)
print("\nAfter Remove The punctuation in string\n-->")
print(res)
```

```
Before Remove The punctuation in string
-->
From: lerxst@wam.umd.edu (where's my thing)
Subject: WHAT car is this!?
Nntp-Posting-Host: rac3.wam.umd.edu
Organization: University of Maryland, College Park
Lines: 15

 I was wondering if anyone out there could enlighten me on this car I saw
the other day. It was a 2-door sports car, looked to be from the late 60s/
early 70s. It was called a Bricklin. The doors were really small. In additio
n,
the front bumper was separate from the rest of the body. This is
all I know. If anyone can tellme a model name, engine specs, years
of production, where this car is made, history, or whatever info you
have on this funky looking car, please e-mail.

Thanks,
- IL
   ---- brought to you by your neighborhood Lerxst ----





------------------------------------------------

After Remove The punctuation in string
-->
From lerxstwamumdedu wheres my thing
Subject WHAT car is this
NntpPostingHost rac3wamumdedu
Organization University of Maryland College Park
Lines 15

 I was wondering if anyone out there could enlighten me on this car I saw
the other day It was a 2door sports car looked to be from the late 60s
early 70s It was called a Bricklin The doors were really small In addition
the front bumper was separate from the rest of the body This is
all I know If anyone can tellme a model name engine specs years
of production where this car is made history or whatever info you
have on this funky looking car please email

Thanks
 IL
    brought to you by your neighborhood Lerxst
```

Q-G)Bag-of-Words

In [46]:

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer

s_tok = sent_tokenize(df)
cv = CountVectorizer()
cv_data = cv.fit([i for i in s_tok])
print("Bag OF Words:")
print()
print(cv_data.vocabulary_)
cv_tr = cv.transform([i for i in se])
pd.DataFrame(cv_tr.toarray(),columns=cv.get_feature_names())
```

```
Bag OF Words:

{'from': 24, 'lerxst': 38, 'wam': 79, 'umd': 77, 'edu': 21, 'where': 84, 'm
y': 47, 'thing': 74, 'subject': 69, 'what': 82, 'car': 14, 'is': 34, 'this':
75, 'nntp': 50, 'posting': 59, 'host': 29, 'rac3': 61, 'organization': 54, 'u
niversity': 78, 'of': 51, 'maryland': 44, 'college': 15, 'park': 57, 'lines':
39, '15': 0, 'was': 80, 'wondering': 85, 'if': 30, 'anyone': 5, 'out': 56, 't
here': 73, 'could': 16, 'enlighten': 23, 'me': 45, 'on': 52, 'saw': 64, 'th
e': 72, 'other': 55, 'day': 17, 'it': 35, 'door': 18, 'sports': 68, 'looked':
40, 'to': 76, 'be': 6, 'late': 37, '60s': 1, 'early': 20, '70s': 2, 'called':
12, 'bricklin': 8, 'doors': 19, 'were': 81, 'really': 62, 'small': 66, 'in':
32, 'addition': 3, 'front': 25, 'bumper': 10, 'separate': 65, 'rest': 63, 'bo
dy': 7, 'all': 4, 'know': 36, 'can': 13, 'tellme': 70, 'model': 46, 'name': 4
8, 'engine': 22, 'specs': 67, 'years': 86, 'production': 60, 'made': 42, 'his
tory': 28, 'or': 53, 'whatever': 83, 'info': 33, 'you': 87, 'have': 27, 'funk
y': 26, 'looking': 41, 'please': 58, 'mail': 43, 'thanks': 71, 'il': 31, 'bro
ught': 9, 'by': 11, 'your': 88, 'neighborhood': 49}
```

Out[46]:

	15	60s	70s	addition	all	anyone	be	body	bricklin	brought	...	wam	was	were	what
0	0	0	0	0	0	0	0	0	0	0	...	1	0	0	1
1	1	0	0	0	0	1	0	0	0	0	...	1	1	0	0
2	0	1	1	0	0	0	1	0	0	0	...	0	1	0	0
3	0	0	0	0	0	0	0	0	1	0	...	0	1	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	1	0
5	0	0	0	1	0	0	0	1	0	0	...	0	1	0	0
6	0	0	0	0	1	0	0	0	0	0	...	0	0	0	0
7	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0
8	0	0	0	0	0	0	0	0	0	1	...	0	0	0	0

9 rows × 89 columns

Q-H)POS Tagging

In [48]:

```
from nltk import pos_tag

word_data=word_tokenize(res)
tokens_tag = pos_tag(word_data)
for i in tokens_tag:
    print(i)
```

```
('From', 'IN')
('lerxstwamumdedu', 'JJ')
('wheres', 'NNS')
('my', 'PRP$')
('thing', 'NN')
('Subject', 'NNP')
('WHAT', 'NNP')
('car', 'NN')
('is', 'VBZ')
('this', 'DT')
('NntpPostingHost', 'NNP')
('rac3wamumdedu', 'NN')
('Organization', 'NNP')
('University', 'NNP')
('of', 'IN')
('Maryland', 'NNP')
('College', 'NNP')
('Park', 'NNP')
('Lines', 'NNP')
('15', 'CD')
('I', 'PRP')
('was', 'VBD')
('wondering', 'VBG')
('if', 'IN')
('anyone', 'NN')
('out', 'IN')
('there', 'RB')
('could', 'MD')
('enlighten', 'VB')
('me', 'PRP')
('on', 'IN')
('this', 'DT')
('car', 'NN')
('I', 'PRP')
('saw', 'VBD')
('the', 'DT')
('other', 'JJ')
('day', 'NN')
('It', 'PRP')
('was', 'VBD')
('a', 'DT')
('2door', 'JJ')
('sports', 'NNS')
('car', 'NN')
('looked', 'VBD')
('to', 'TO')
('be', 'VB')
('from', 'IN')
('the', 'DT')
('late', 'JJ')
('60s', 'NNS')
('early', 'RB')
('70s', 'CD')
('It', 'PRP')
('was', 'VBD')
('called', 'VBN')
('a', 'DT')
```

```
('Bricklin', 'NNP')
('The', 'DT')
('doors', 'NNS')
('were', 'VBD')
('really', 'RB')
('small', 'JJ')
('In', 'IN')
('addition', 'NN')
('the', 'DT')
('front', 'NN')
('bumper', 'NN')
('was', 'VBD')
('separate', 'JJ')
('from', 'IN')
('the', 'DT')
('rest', 'NN')
('of', 'IN')
('the', 'DT')
('body', 'NN')
('This', 'DT')
('is', 'VBZ')
('all', 'DT')
('I', 'PRP')
('know', 'VBP')
('If', 'IN')
('anyone', 'NN')
('can', 'MD')
('tellme', 'VB')
('a', 'DT')
('model', 'NN')
('name', 'NN')
('engine', 'NN')
('specs', 'CD')
('years', 'NNS')
('of', 'IN')
('production', 'NN')
('where', 'WRB')
('this', 'DT')
('car', 'NN')
('is', 'VBZ')
('made', 'VBN')
('history', 'NN')
('or', 'CC')
('whatever', 'WDT')
('info', 'VBP')
('you', 'PRP')
('have', 'VBP')
('on', 'IN')
('this', 'DT')
('funky', 'NN')
('looking', 'VBG')
('car', 'NN')
('please', 'NN')
('email', 'VBP')
('Thanks', 'NNP')
('IL', 'NNP')
('brought', 'VBD')
```

```
('to', 'TO')
('you', 'PRP')
('by', 'IN')
('your', 'PRP$')
('neighborhood', 'NN')
('Lerxst', 'NN')
```

Q-I)N-grams

In [52]:

```
cv = CountVectorizer(ngram_range=(3, 3))
cv_data = cv.fit([i for i in se])
print("N-grams\n")
print(cv_data.vocabulary_)
cv_tr = cv.transform([i for i in s_tok])
pd.DataFrame(cv_tr.toarray(),columns=cv.get_feature_names())
```

```
N-grams

{'from lerxst wam': 23, 'lerxst wam umd': 41, 'wam umd edu': 88, 'umd edu whe
re': 86, 'edu where my': 20, 'where my thing': 96, 'my thing subject': 49, 't
hing subject what': 78, 'subject what car': 69, 'what car is': 94, 'car is th
is': 11, 'nntp posting host': 51, 'posting host rac3': 61, 'host rac3 wam': 3
0, 'rac3 wam umd': 63, 'umd edu organization': 85, 'edu organization universi
ty': 19, 'organization university of': 58, 'university of maryland': 87, 'of
maryland college': 52, 'maryland college park': 46, 'college park lines': 15,
'park lines 15': 60, 'lines 15 was': 42, '15 was wondering': 0, 'was wonderin
g if': 92, 'wondering if anyone': 98, 'if anyone out': 32, 'anyone out ther
e': 4, 'out there could': 59, 'there could enlighten': 77, 'could enlighten m
e': 16, 'enlighten me on': 22, 'me on this': 47, 'on this car': 55, 'this car
saw': 80, 'car saw the': 14, 'saw the other': 65, 'the other day': 75, 'it wa
s door': 39, 'was door sports': 90, 'door sports car': 17, 'sports car looke
d': 68, 'car looked to': 12, 'looked to be': 43, 'to be from': 83, 'be from t
he': 5, 'from the late': 24, 'the late 60s': 74, 'late 60s early': 40, '60s e
arly 70s': 1, 'it was called': 38, 'was called bricklin': 89, 'the doors wer
e': 72, 'doors were really': 18, 'were really small': 93, 'in addition the':
34, 'addition the front': 2, 'the front bumper': 73, 'front bumper was': 26,
'bumper was separate': 7, 'was separate from': 91, 'separate from the': 66,
'from the rest': 25, 'the rest of': 76, 'rest of the': 64, 'of the body': 54,
'this is all': 82, 'is all know': 36, 'if anyone can': 31, 'anyone can tellm
e': 3, 'can tellme model': 9, 'tellme model name': 70, 'model name engine': 4
8, 'name engine specs': 50, 'engine specs years': 21, 'specs years of': 67,
'years of production': 99, 'of production where': 53, 'production where thi
s': 62, 'where this car': 97, 'this car is': 79, 'car is made': 10, 'is made
history': 37, 'made history or': 45, 'history or whatever': 29, 'or whatever
info': 57, 'whatever info you': 95, 'info you have': 35, 'you have on': 101,
'have on this': 28, 'on this funky': 56, 'this funky looking': 81, 'funky loo
king car': 27, 'looking car please': 44, 'car please mail': 13, 'thanks il br
ought': 71, 'il brought to': 33, 'brought to you': 6, 'to you by': 84, 'you b
y your': 100, 'by your neighborhood': 8, 'your neighborhood lerxst': 102}
```

Out[52]:

	15 was wondering	60s early 70s	addition the front	anyone can tellme	anyone out there	be from the	brought to you	bumper was separate	by your neighborhood	can tellme model
0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	1	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	1	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	1
8	0	0	0	0	0	0	1	0	1	0

9 rows × 103 columns

Q- I)TF-IDF

In [54]:

```python
from sklearn.feature_extraction.text import TfidfVectorizer
tf = TfidfVectorizer()
tf_data = tf.fit_transform([i for i in s_tok])
pd.DataFrame(tf_data.toarray(),columns=tf.get_feature_names())
```

Out[54]:

	15	60s	70s	addition	all	anyone	be	body	bricklin	brou
0	0.000000	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	0.0000
1	0.194244	0.00000	0.00000	0.000000	0.000000	0.164062	0.00000	0.000000	0.000000	0.0000
2	0.000000	0.29842	0.29842	0.000000	0.000000	0.000000	0.29842	0.000000	0.000000	0.0000
3	0.000000	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.564838	0.0000
4	0.000000	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	0.0000
5	0.000000	0.00000	0.00000	0.285264	0.000000	0.000000	0.00000	0.285264	0.000000	0.0000
6	0.000000	0.00000	0.00000	0.000000	0.581208	0.000000	0.00000	0.000000	0.000000	0.0000
7	0.000000	0.00000	0.00000	0.000000	0.000000	0.165600	0.00000	0.000000	0.000000	0.0000
8	0.000000	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.000000	0.3504

9 rows × 89 columns