Name: Rahul Gudivada Reg no:19BCE2469

Task 1: Stemming find the one that porter stems but lancaster doesn't.

In [1]:

```
import nltk
from nltk.stem import PorterStemmer
stemmerporter=PorterStemmer()
#stemmerporter.stem('happiness')
stemmerporter.stem('manageable')
```

Out[1]:

```
'manag'
```

In [2]:

```
from nltk.stem import LancasterStemmer
stemmerlancaster=LancasterStemmer()
stemmerlancaster.stem('manageable')
```

Out[2]:

```
'man'
```

In [3]:

```
from nltk.stem import RegexpStemmer
stemmerregex= RegexpStemmer('ing')
stemmerregex.stem('vibing')
```

Out[3]:

```
'vib'
```

Task 2: demonstrate snowball stemming in hindi/english/french

In [4]:

```
from nltk.stem import SnowballStemmer
print(" ".join (SnowballStemmer.languages))
```

```
arabic danish dutch english finnish french german hungarian italian norwegia
n porter portuguese romanian russian spanish swedish
```

In [5]:

```
stemmer=SnowballStemmer("german")
stemmer.stem('Gesundheit')
```

Out[5]:

```
'gesund'
```

In [6]:

```
stemmer=SnowballStemmer("french")
stemmer.stem('manger')
```

Out[6]:

```
'mang'
```

Task 3: stem paragraphs

In [7]:

```
import os
with open(r"C:\Users\rahul\OneDrive\Desktop\nlpppppp.txt",'r') as f:
    contents=f.read()
    print(contents)
```

```
hor: Love and Thunder is a 2022 American superhero film based on Marvel Comi
cs featuring the character Thor, produced by Marvel Studios and distributed
by Walt Disney Studios Motion Pictures. It is the sequel to Thor: Ragnarok
(2017) and the 29th film in the Marvel Cinematic Universe (MCU). The film is
directed by Taika Waititi, who co-wrote the script with Jennifer Kaytin Robi
nson, and stars Chris Hemsworth as Thor alongside Christian Bale, Tessa Thom
pson, Jaimie Alexander, Waititi, Russell Crowe, and Natalie Portman. In the
film, Thor attempts to find inner peace, but must return to action and recru
it Valkyrie (Thompson), Korg (Waititi), and Jane Foster (Portman)â€”who is n
ow the Mighty Thorâ€”to stop Gorr the God Butcher (Bale) from eliminating al
l gods.
```

In [8]:

```
example=[stemmerporter.stem(token) for token in contents.split(" ")]
print(" ".join(example))
```

```
hor: love and thunder is a 2022 american superhero film base on marvel comic
featur the charact thor, produc by marvel studio and distribut by walt disne
y studio motion pictures. it is the sequel to thor: ragnarok (2017) and the
29th film in the marvel cinemat univers (mcu). the film is direct by taika w
aititi, who co-wrot the script with jennif kaytin robinson, and star chri he
msworth as thor alongsid christian bale, tessa thompson, jaimi alexander, wa
ititi, russel crowe, and natali portman. in the film, thor attempt to find i
nner peace, but must return to action and recruit valkyri (thompson), korg
(waititi), and jane foster (portman)â€”who is now the mighti thorâ€”to stop
gorr the god butcher (bale) from elimin all gods.
```

Task 4: Lemmatizer

In [9]:

```
from nltk.stem import WordNetLemmatizer
lemmatizer=WordNetLemmatizer()
print(lemmatizer.lemmatize("cacti"))
print(lemmatizer.lemmatize("formulae"))
```

```
cactus
formula
```

In [10]:

```
print(lemmatizer.lemmatize("better",pos='a'))
```

```
good
```

Task 5: Jieba

In [11]:

```
!pip install jieba
```

```
Requirement already satisfied: jieba in c:\users\rahul\appdata\local\program
s\python\python310\lib\site-packages (0.42.1)

WARNING: You are using pip version 22.0.4; however, version 22.2 is availabl
e.
You should consider upgrading via the 'C:\Users\rahul\AppData\Local\Programs
\Python\Python310\python.exe -m pip install --upgrade pip' command.
```

In [12]:

```
import jieba
seg= jieba.cut("我喜欢踢足球。梅西是我最喜欢的球员")
print(" ".join(seg))
```

```
Building prefix dict from the default dictionary ...
Loading model from cache C:\Users\rahul\AppData\Local\Temp\jieba.cache
Loading model cost 0.621 seconds.
Prefix dict has been built successfully.

我 喜欢 踢足球 。 梅西 是 我 最 喜欢 的 球员
```

Task 6: Tokenization

In [13]:

```
texts = "Thor: Love and Thunder is a 2022 American superhero film based on Marvel Comics fe
for text in texts:
    sentences= nltk.sent_tokenize(texts)
    for sentence in sentences:
        words= nltk.word_tokenize(sentence)
        tagged= nltk.pos_tag(words)
        print (tagged)
```

```
'CC'), ('Natalie', 'NNP'), ('Portman', 'NNP'), ('.', '.')]
[('In', 'IN'), ('the', 'DT'), ('film', 'NN'), (',', ','), ('Thor', 'NNP'),
('attempts', 'VBZ'), ('to', 'TO'), ('find', 'VB'), ('inner', 'JJ'), ('peac
e', 'NN'), (',', ','), ('but', 'CC'), ('must', 'MD'), ('return', 'VB'),
('to', 'TO'), ('action', 'NN'), ('and', 'CC'), ('recruit', 'NN'), ('Valkyr
ie', 'NNP'), ('(', '('), ('Thompson', 'NNP'), (')', ')'), (',', ','), ('Ko
rg', 'NNP'), ('(', '('), ('Waititi', 'NNP'), (')', ')'), (',', ','), ('an
d', 'CC'), ('Jane', 'NNP'), ('Foster', 'NNP'), ('(', '('), ('Portman', 'NN
P'), (')', ')'), ('—who', 'NN'), ('is', 'VBZ'), ('now', 'RB'), ('the', 'D
T'), ('Mighty', 'NNP'), ('Thor—to', 'NNP'), ('stop', 'VB'), ('Gorr', 'NN
P'), ('the', 'DT'), ('God', 'NNP'), ('Butcher', 'NNP'), ('(', '('), ('Bal
e', 'NNP'), (')', ')'), ('from', 'IN'), ('eliminating', 'VBG'), ('all', 'D
T'), ('gods', 'NNS'), ('.', '.')]
[('Thor', 'NN'), (':', ':'), ('Love', 'NNP'), ('and', 'CC'), ('Thunder',
'NNP'), ('is', 'VBZ'), ('a', 'DT'), ('2022', 'JJ'), ('American', 'JJ'),
('superhero', 'NN'), ('film', 'NN'), ('based', 'VBN'), ('on', 'IN'), ('Mar
vel', 'NNP'), ('Comics', 'NNPS'), ('featuring', 'VBG'), ('the', 'DT'), ('c
haracter', 'NN'), ('Thor', 'NNP'), (',', ','), ('produced', 'VBN'), ('by',
'IN'), ('Marvel', 'NNP'), ('Studios', 'NNP'), ('and', 'CC'), ('distribute
```

In [ ]: