

a. Map words to properties using python dictionaries

Code and Output:

pos={}
pos

{ }

pos['colorless'] = 'ADJ'

pos

{ 'colorless': 'ADJ' }

pos['ideas'] = 'N'

pos['sleep'] = 'v'

pos['furiously'] = 'ADV'

pos

{ 'colorless': 'ADJ', 'ideas': 'N', 'sleep': 'v', 'furiously': 'ADV' }

pos['ideas']

'N'

pos['colorless']

'ADJ'

list(pos)

['colorless', 'ideas', 'sleep', 'furiously']

sorted(pos)

['colorless', 'furiously', 'ideas', 'sleep']

[w for w in pos if w.endswith('s')]

['colorless', 'ideas']

for word in sorted(pos):

print(word + ":", pos[word])

```
colorless: ADJ
furiously: ADV
ideas: N
sleep: v
```

```
pos.keys()
```

```
dict_keys(['colorless', 'ideas', 'sleep', 'furiously'])
```

```
pos.items()
```

```
dict_items([('colorless', 'ADJ'), ('ideas', 'N'), ('sleep', 'v'), ('furiously', 'ADV')])
```

```
pos['sleep'] = ['N', 'V']
```

```
pos.items()
```

```
dict_items([('colorless', 'ADJ'), ('ideas', 'N'), ('sleep', ['N', 'V']), ('furiously', 'ADV')])
```

```
pos = {'colorless': 'ADJ', 'ideas': 'N', 'sleep': 'V', 'furiously': 'ADV'}
```

```
pos
```

```
{'colorless': 'ADJ', 'ideas': 'N', 'sleep': 'V', 'furiously': 'ADV'}
```

b. Study Default Tagger, Regular Expression tagger, Unigram Tagger

Code and Output:

```
#default_tagger
```

```
import nltk
```

```
from nltk.corpus import brown
```

```
brown_tagged_sents = brown.tagged_sents(categories='news')
```

```
brown_sents = brown.sents(categories='news')
```

```
tags = [tag for (word, tag) in brown.tagged_words(categories='news')]
```

```
nltk.FreqDist(tags).max()
```

```
'NN'
```

```
raw = 'I do not like green eggs and ham, I do not like them Sam I am!'
```

```
tokens = nltk.word_tokenize(raw)
```

```
default_tagger = nltk.DefaultTagger('NN')
```

```
default_tagger.tag(tokens)
```

```
[('I', 'NN'), ('do', 'NN'), ('not', 'NN'), ('like', 'NN'), ('green', 'NN'), ('eggs', 'NN'), ('and', 'NN'), ('ham', 'NN'), (',', 'NN'), ('I', 'NN'), ('do', 'NN'), ('not', 'NN'), ('like', 'NN'), ('them', 'NN'), ('Sam', 'NN'), ('I', 'NN'), ('am', 'NN'), ('!', 'NN')]
```

```
default_tagger.evaluate(brown_tagged_sents)
```

```
0.13089484257215028
```

```

#regex_tagger

import nltk

from nltk.corpus import brown

brown_tagged_sents = brown.tagged_sents(categories='news')

brown_sents = brown.sents(categories='news')

patterns = [

    (r'.*ing$', 'VBG'), # gerunds

    (r'.*ed$', 'VBD'), # simple past

    (r'.*es$', 'VBZ'), # 3rd singular present

    (r'.*ould$', 'MD'), # modals

    (r'.*\s$', 'NN$'), # possessive nouns

    (r'.*s$', 'NNS'), # plural nouns

    (r'^-?[0-9]+(.[0-9]+)?$', 'CD'), # cardinal numbers

    (r'.*', 'NN') # nouns (default)

]

regex_tagger = nltk.RegexpTagger(patterns)

regex_tagger.tag(brown_sents[3])

```

```

[('``', 'NN'), ('Only', 'NN'), ('a', 'NN'), ('relative',
'NN'), ('handful', 'NN'), ('of', 'NN'), ('such', 'NN'),
('reports', 'NNS'), ('was', 'NNS'), ('received', 'VBD'),
(''''', 'NN'), (',', 'NN'), ('the', 'NN'), ('jury',
'NN'), ('said', 'NN'), (',', 'NN'), ('``', 'NN'), ('con
sidering', 'VBG'), ('the', 'NN'), ('widespread', 'NN'),
('interest', 'NN'), ('in', 'NN'), ('the', 'NN'), ('elec
tion', 'NN'), (',', 'NN'), ('the', 'NN'), ('number', 'N
N'), ('of', 'NN'), ('voters', 'NNS'), ('and', 'NN'), ('
the', 'NN'), ('size', 'NN'), ('of', 'NN'), ('this', 'NN
S'), ('city', 'NN'), (''''', 'NN'), (',', 'NN')]

```

```

regex_tagger.evaluate(brown_tagged_sents)

```

```

0.20326391789486245

```

```
#unigram_tagger
import nltk

from nltk.corpus import brown

brown_tagged_sents = brown.tagged_sents(categories='news')
brown_sents = brown.sents(categories='news')
unigram_tagger = nltk.UnigramTagger(brown_tagged_sents)
unigram_tagger.tag(brown_sents[2007])

[('Various', 'JJ'), ('of', 'IN'), ('the', 'AT'), ('apar
tments', 'NNS'), ('are', 'BER'), ('of', 'IN'), ('the',
'AT'), ('terrace', 'NN'), ('type', 'NN'), (',', ','), ('
being', 'BEG'), ('on', 'IN'), ('the', 'AT'), ('ground'
, 'NN'), ('floor', 'NN'), ('so', 'QL'), ('that', 'CS'),
('entrance', 'NN'), ('is', 'BEZ'), ('direct', 'JJ'), ('
.', '.')]

unigram_tagger.evaluate(brown_tagged_sents)

0.9349006503968017
```