```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Unzipping taggers/averaged_perceptron_tagger.zip.
True
```

```
from nltk.chunk import RegexpParser
from nltk.tokenize import word_tokenize
```

```
sentence = "Vidyavardhini's college of Engineering and Technology VASAI"
```

Tokenization

```
tokens = word_tokenize(sentence)
```

```
tokens
```

```
['Vidyavardhini',
 "'s",
 'college',
 'of',
 'Engineering',
 'and',
 'Technology',
 'VASAI']
```

POS tagging

```
pos_tags = nltk.pos_tag(tokens)
```

```
pos_tags
```

```
[('Vidyavardhini', 'NNP'),
 ("'s", 'POS'),
 ('college', 'NN'),
 ('of', 'IN'),
 ('Engineering', 'NNP'),
 ('and', 'CC'),
 ('Technology', 'NNP'),
 ('VASAI', 'NNP')]
```

Chunking patterns

```
chunk_patterns = r"""
    NP: {<DT>?<JJ>*<NN>}  # Chunk noun phrases
    VP: {<VB.*><NP|PP>}  # Chunk verb phrases
"""
```

```
chunk_patterns
```

```
'\n    NP: {<DT>?<JJ>*<NN>}  # Chunk noun phrases\n    VP: {<VB.*><NP|PP>}  # Chunk verb phrases\n'
```

Create a chunk parser

```
chunk_parser = RegexpParser(chunk_patterns)
```

```
chunk_parser
```

```
<chunk.RegexpParser with 2 stages>
```

Perform chunking

```
result = chunk_parser.parse(pos_tags)


print(result)
```

```
     (S
       Vidyavardhini/NNP
       's/POS
       (NP college/NN)
       of/IN
       Engineering/NNP
       and/CC
       Technology/NNP
       VASAI/NNP)
```