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

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
True
```

```
grammar = nltk.CFG.fromstring("""
  S -> NP VP
  NP -> ProperNoun ProperNoun | Det Noun PP | Det Noun | ProperNoun
  VP -> Verb Verb Det Noun Prep Noun | Verb VP | Verb NP PP | Verb NP | Verb Verb NP PP
  PP -> Prep NP

  Det -> 'the'
  Noun -> 'God' | 'Cricket'
  ProperNoun -> 'Sachin' | 'Tendulkar'
  Verb -> 'is' | 'called'
  Prep -> 'of'
""")

parser = nltk.ChartParser(grammar)

tokens = ['Sachin', 'Tendulkar', 'is', 'called', 'the', 'God', 'of', 'Cricket']

trees = list(parser.parse(tokens))

if trees:
    for tree in trees:
        print(tree)
    trees[0].pretty_print()
else:
    print("No parse trees found for the given sentence with the current grammar.")
```

```
(S
  (NP (ProperNoun Sachin) (ProperNoun Tendulkar))
  (VP
    (Verb is)
    (Verb called)
    (Det the)
    (Noun God)
    (Prep of)
    (Noun Cricket)))
                                 S
             ____________________|_____
            NP                         VP
     _______|______         ___________|_______________
ProperNoun     ProperNoun Verb  Verb  Det Noun Prep   Noun
    |              |       |     |     |   |    |      |
  Sachin       Tendulkar   is  called the God   of  Cricket
```