In [1]:
```
import spacy
```

In [2]:
```
nlp = spacy.load('en_core_web_sm')
```

In [4]:
```
mystring = '"We\'re moving to L.A.!"'
```

In [5]:
```
print(mystring)
```

```
"We're moving to L.A.!"
```

In [6]:
```
doc = nlp(mystring)

for token in doc:
    print(token.text)
```

```
"
We
're
moving
to
L.A.
!
"
```

In [7]:
```
for token in doc:
    print(token.text, end= ' | ')
```

```
" | We | 're | moving | to | L.A. | ! | " |
```

In [8]:
```
doc2 = nlp(u"We're here to help! Send snail-email, email support@mysite.
```

In [9]:
```
for t in doc2:
    print(t)
```

```
We
're
here
to
help
!
Send
snail
-
email
,
email
support@mysite.com
or
visit
us
at
https://www.oursite.com (https://www.oursite.com)
!
```

In [10]:
```
for t in doc2:
    print(t, end = ' | ')
```

```
We | 're | here | to | help | ! | Send | snail | - | email | , | email | support@mysite.com | or | visit | us | at | https://www.oursite.com (https://www.oursite.com) | ! |
```

In [11]:
```
doc3 = nlp("A 5 km NYC cab rise costs $10.30")

for t in doc3:
    print(t, end = ' | ')
```

```
A | 5 | km | NYC | cab | rise | costs | $ | 10.30 |
```

In [12]:
```
doc4 = nlp(u"Let's visit St. Louis in the U.S. next year.")

for t in doc4:
    print(t, end = ' | ')
```

```
Let | 's | visit | St. | Louis | in | the | U.S. | next | year | . |
```

In [13]:
```
# Counting Tokens
len(doc)
```

Out[13]: 8

In [14]:
```
len(doc.vocab)
```

Out[14]: 57852

In [16]:
```
len(doc4.vocab)
```

Out[16]: 57852

In [17]:
```
# Token can be retrieved by index position and slice
doc5 = nlp(u"It is better to give than to receive.")
for t in doc5:
    print(t, end = ' | ')
```

```
It | is | better | to | give | than | to | receive | . |
```

In [18]:
```
len(doc5)
```

Out[18]: 9

In [19]:
```
doc5[2]
```

Out[19]: better

In [20]:
```
doc5[2:5]
```

Out[20]: better to give

In [21]:
```
#Retrieve the last 4 tokens
doc5[-4:]
```

Out[21]: than to receive.

In [22]:
```
# Tokens cannot be reassigned
doc6 = nlp(u"My dinner was horrible.")
doc7 = nlp(u'Your dinner was delicious.')
```

In [23]:
```
doc6[3] = doc7[3]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-23-c84f13888331> in <module>
----> 1 doc6[3] = doc7[3]

TypeError: 'spacy.tokens.doc.Doc' object does not support item assignment
```

In [24]:
```
# Named Entities
doc8 = nlp(u'Apple to build a Hong Kong factory for $6 million')
```

In [25]:
```
for t in doc8:
    print(t, end = ' | ')
```

```
Apple | to | build | a | Hong | Kong | factory | for | $ | 6 | million |
```

In [26]:
```
for entity in doc8.ents:
    print(entity, end = ' | ')
```

```
Apple | Hong Kong | $6 million |
```

In [27]:
```
for entity in doc8.ents:
    print(entity.text, end = ' | ')
    print(entity.label_, end = ' | ')
    print(spacy.explain(entity.label_))
```

```
Apple | ORG | Companies, agencies, institutions, etc.
Hong Kong | GPE | Countries, cities, states
$6 million | MONEY | Monetary values, including unit
```

In [28]:
```
len(doc8.ents)
```

Out[28]: 3

In [29]:
```
doc9 = nlp(u'Autonomous cars shift the insurance liability towards manuf
```

In [30]:
```
for chunk in doc9.noun_chunks:
    print(chunk.text)
```

```
Autonomous cars
the insurance liability
manufacturers
```

In [31]:
```
doc10 = nlp(u'Red cars do not carry higher insurance rates.')
for chunk in doc10.noun_chunks:
    print(chunk.text)
```

```
Red cars
higher insurance rates
```

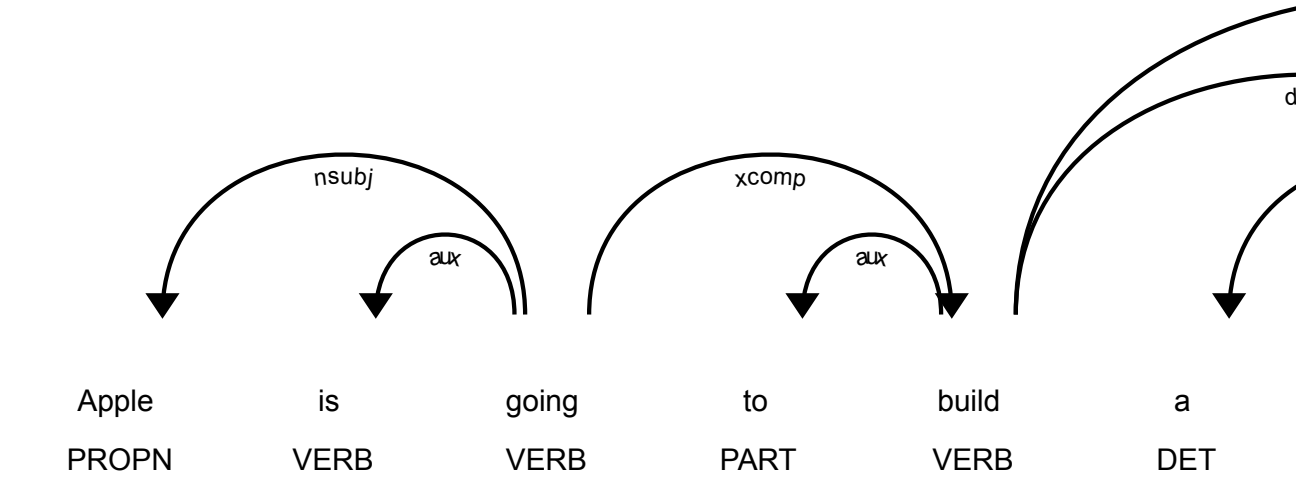
In [32]:
```
doc11 = nlp(u'He was a one-eye, one-horned, flying, purple people-eater.
for chunk in doc11.noun_chunks:
    print(chunk.text)
```

```
He
a one-eye, one-horned, flying, purple people-eater
```

In [40]:
```
# visualizing the dependency parse
from spacy import displacy

doc = nlp(u'Apple is going to build a U.K. factory for $6 million.')

displacy.render(doc, style ='dep', jupyter = True, options={'distance':
```



In [41]:
```
displacy.render(doc, style='ent', jupyter=True)
```

Apple **ORG** is going to build a U.K. **GPE** factory for $6 million **MONEY** .

In [42]:
```
doc = nlp(u'Over the lsst quarter Apple sold nearly 20 thousand ipods fo
```

In [43]:
```
displacy.render(doc, style='ent', jupyter=True)
```

Over the lsst quarter **CARDINAL** Apple **ORG** sold nearly 20 thousand **CARDINAL**

ipods for a profit of $6 million **MONEY** .

In [ ]:
```
# Creating visualizations outside of Jupyter

doc = nlp(u'This is a sentence.')
displacy.serve(doc, style = 'dep')
```

```
Serving on port 5000...
Using the 'dep' visualizer
```

In [ ]:
```

```