In [2]:
```
# Blank NLP pipeline
import spacy
```

In [3]:
```
nlp = spacy.blank("en")
doc = nlp("Captain america ate 100$ of samosa. Then he said I can do this all day.'
for token in doc:
    print(token)
```

```
Captain
america
ate
100
$
of
samosa
.
Then
he
said
I
can
do
this
all
day
.
```

In [4]:
```
nlp.pipe_names
```

Out[4]:
```
[]
```

Blank means - blank pipeline, but it printed tokens because pipeline names comes after tokenizer

Downloading spacy pipelines

In [6]:
```
nlp = spacy.load("en_core_web_sm")
nlp.pipe_names
```

Out[6]:
```
['tok2vec', 'tagger', 'parser', 'attribute_ruler', 'lemmatizer', 'ner']
```

In [8]:
```
nlp.pipeline
```

Out[8]:
```
[('tok2vec', <spacy.pipeline.tok2vec.Tok2Vec at 0x1aaaa2bd640>),
 ('tagger', <spacy.pipeline.tagger.Tagger at 0x1aaaa393e20>),
 ('parser', <spacy.pipeline.dep_parser.DependencyParser at 0x1aaa900c660>),
 ('attribute_ruler',
  <spacy.pipeline.attributeruler.AttributeRuler at 0x1aaaa3f18c0>),
 ('lemmatizer', <spacy.lang.en.lemmatizer.EnglishLemmatizer at 0x1aaaa42e740>),
 ('ner', <spacy.pipeline.ner.EntityRecognizer at 0x1aaa900c510>)]
```

In [11]:
```
doc = nlp("Captain america ate 100$ of samosa. Then he said I can do this all day.'

for token in doc:
    print (token, "|", token.pos_, "|", token.lemma_)
    ###print (token, "|", spacy.explain(token.pos_), "|", token.lemma_) - prop noun
```

```
Captain | PROPN | Captain
america | PROPN | america
ate | VERB | eat
100 | NUM | 100
$ | NUM | $
of | ADP | of
samosa | PROPN | samosa
. | PUNCT | .
Then | ADV | then
he | PRON | he
said | VERB | say
I | PRON | I
can | AUX | can
do | VERB | do
this | PRON | this
all | DET | all
day | NOUN | day
. | PUNCT | .
```

NER

In [18]:

```
doc = nlp("Tesla Inc is going to acquire twitter for $45 billion")
for ent in doc.ents:
    print(ent.text,ent.label_)
```

```
Tesla Inc ORG
$45 billion MONEY
```

To explain more about labels

In [19]:

```
doc = nlp("Tesla Inc is going to acquire twitter for $45 billion")
for ent in doc.ents:
    print(ent.text,ent.label, spacy.explain(ent.label_))
```

```
Tesla Inc 383 Companies, agencies, institutions, etc.
$45 billion 394 Monetary values, including unit
```

For more good display

In [20]:

```
from spacy import displacy
displacy.render(doc, style="ent")
```

Tesla Inc **ORG** is going to acquire twitter for $45 billion **MONEY**

Adding a component to a blank pipeline - such as NER

In [23]:

```
source_nlp = spacy.load("en_core_web_sm")
nlp = spacy.blank("en")
nlp.add_pipe("ner",source = source_nlp)
nlp.pipe_names
```

Out[23]:

```
['ner']
```

In [25]:

```
doc = nlp("Tesla Inc is going to acquire twitter for $45 billion")
for ent in doc.ents:
    print(ent.text, ent.label_)
```

```
Tesla Inc ORG
$45 billion MONEY
```