Unit 4 Assignment

**ANLY:520-51 (Fall 2016)**

Dean D’souza

H.U. ID: 168424

# **Solutions:**

1. **Tokenize and tag the following sentence: They wind back the clock, while we chase after the wind. What different pronunciations and parts of speech are involved?**

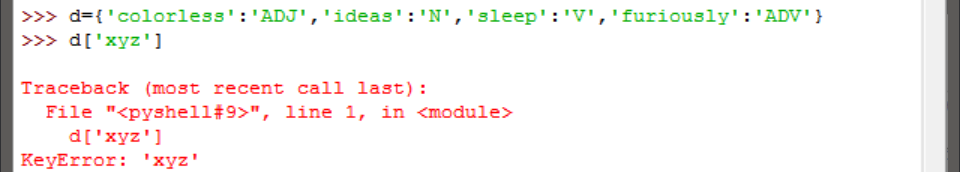
We perform the required commands as follows:

****

The above screenshot shows us the parts of speech for each token. This includes a mix of determiners, nouns, etc.

1. **Using the Python interpreter in interactive mode, experiment with the dictionary examples in this chapter. Create a dictionary d, and add some entries. What happens if you try to access a non-existent entry, e.g. d['xyz']?**

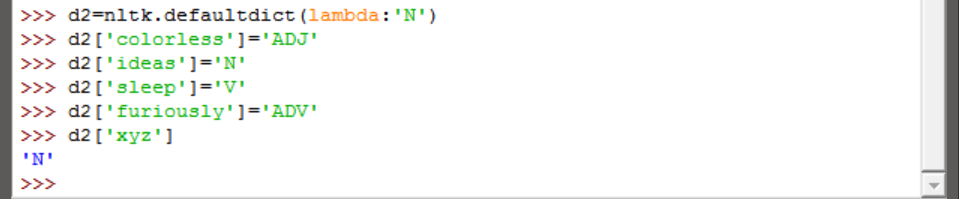
We perform the required commands as follows using different conventions for creating the dictionaries:



The above method is a standard and easy way to create the dictionary. However, in this case we can see that accessing a non-existent entry gives us a KeyError.



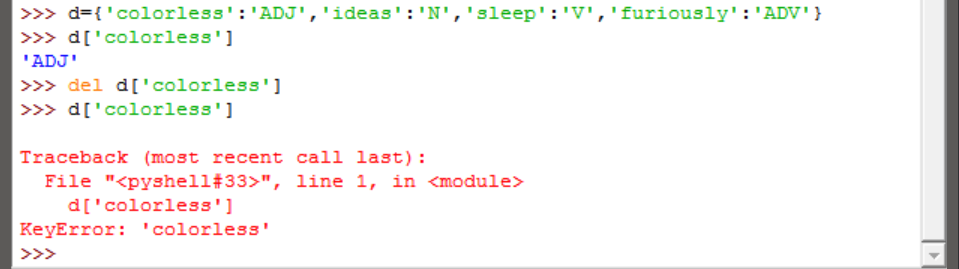
The above method uses the defaultdict() function provided by nltk (also present in python 2.5 and later). We can see that accessing a non-existent key gives us an empty [].



The above usage of the defaultdict() method assigns a default value for new keys. Hence, accessing a non-existent key just gives us the default value of ‘N’.

1. **Try deleting an element from a dictionary d, using the syntax del d['abc']. Check that the item was deleted.**

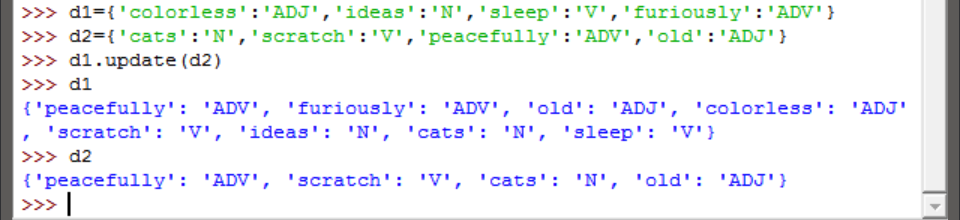
We perform the required task using the simple method of creating dictionaries (seen in the previous question) and observe the result:



As we can see, the key value pair was deleted such that an attempt to access it gave a KeyError.

1. **Create two dictionaries, d1 and d2, and add some entries to each. Now issue the command d1.update(d2). What did this do? What might it be useful for?**

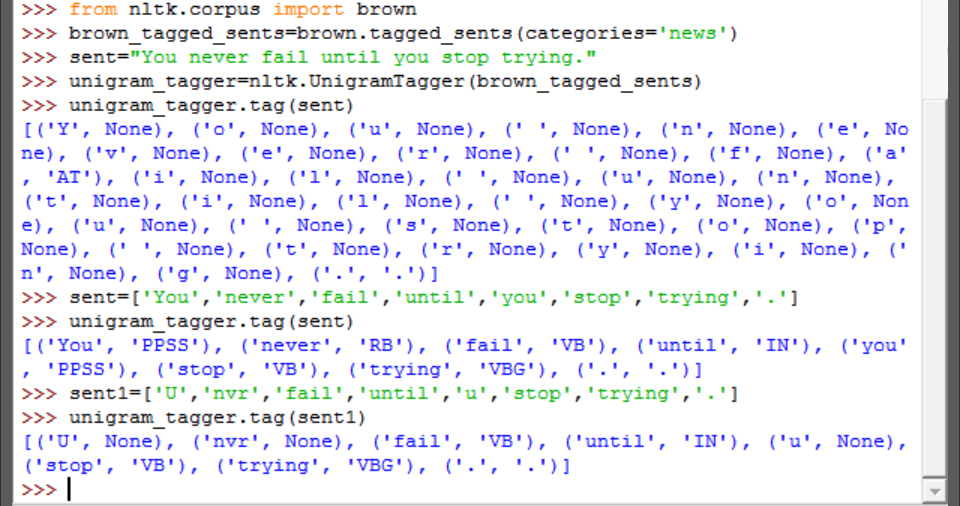
We again use the simple method of creating the dictionaries and observe the results as follows:



As expected, the dictionary d1 updates itself to contain the values present in d2. This would be especially useful if more key value pairs need to be added to the dictionary over time. Additionally, it would make the task of constructing the dictionary easier as one could work on a few words at a time and then update as necessary.

1. **Train a unigram tagger and run it on some new text. Observe that some words are not assigned a tag. Why not?**

We perform the required commands as follows (We use the brown corpus from nltk to construct the unigram tagger, but we do not separate it into training and test data sets as the text to be tested is provided by us):



As we can see, when we pass a simple string, the unigram tagger separates each character as a token. Hence we remake the sentence as a list of tokens before processing. In this case, the tagger was able to tag all the words, so instead we modified the sentence again, replacing words with commonly used chat short forms. We then got a few tokens which were not tagged. This is because the unigram tagger has never encountered such words or tokens before and needs to be trained further to account for such words. Additionally, None seems to be the default tag for such new words.