**Sets**

A ***set*** is a list of ***unique*** ***elements***, usually taken from a list. For example, if the word “banana” is a list of letters, then the set of unique elements in “banana” is:

b, a, n

To create a set, use the set() function. Here is an example:

sentence = "the quick red fox jumps over the lazy brown dog"

lettersUsed = set(sentence) # make a set from the sentence

print (lettersUsed)

The set is a little confusing to navigate. Let's try sorting it:

sentence = "the quick red fox jumps over the lazy brown dog"

lettersUsed = set(sentence) # make a set from the sentence

lettersUsed.sort()

print (lettersUsed)

Uh-oh. You can't sort a set.

Let's turn it into a list then:

sentence = "the quick red fox jumps over the lazy brown dog"

lettersUsed = set(sentence) # make a set from the sentence

lettersUsed.sort()

print (lettersUsed)

Notice that the space character is in the set. Suppose we want to take it out:

sentence = "the quick red fox jumps over the lazy brown dog"

lettersUsed = set(sentence) - set(" ")

print (lettersUsed)

Finally, a set isn’t indexed. The elements of a set do not have a particular order. Sets don’t have a sort function, but lists do. We can convert the set into a list and then call the sort() function to sort it alphabetically:

sentence = "the quick red fox jumps over the lazy brown dog"

lettersUsed = set(sentence) - set(" ")

letters = list(lettersUsed) # convert to a list

list.sort(letters) # sort it

print (letters)

Sets are not used as much as the other types of array structures.

Trivial Fact: The sentence (“the quick red fox...” ) is a fixture among typing classes because it contains every letter of the alphabet.

Python has a *somewhat* awkward slice operator [:] to work with part of a list:

mySiblings = ["Jacob", "Max", "Sarah", "Melanie"]

print (mySiblings[1:3]) # Elements 1 and 2

print (mySiblings[0 : -1]) # prints to the end minus one

sisters = mySiblings[2:4] # elements 2 and 3

print (sisters)

**Summary**

**List**: An indexed set of similar items. Think groceries:

myList = [‘milk’, ‘sugar’, ‘flour’, ‘salt’]

print (myList[2]) #will print ‘flour’

**Set**: A non-indexed list of unique items. Think names in a classroom:

names = [‘Chris’, ‘Alex’, ‘Alex', ‘Michael’] # list of names in class

uniqueNames = set(names) # ‘Chris', 'Alex', 'Michael'

**Tuple**: An indexed, immutable set of values. Think coordinates:

point = 4,5 # once set, cannot be changed

**Dictionary**: A set of paired items in the form key:value. Think record books:

ages = {‘travis’:16, ‘alanna’:15, ‘ahmed’:17}