`` Please fill in an explanation of each function and an example of how to use it below.

List:

      append()
      extend()
      index()
      index(value, integer)
      insert(position)
      remove()
      pop()
      count()
      reverse()
      sort()
      [1]+[1]
      [2]*2
      [1,2][1:]
      [x for x in [2,3]]
      [x for x in [1,2] if x ==1]
      [y*2 for x in [[1,2],[3,4]] for y in x]

      A = [1]

Tuple:

      count()
      index()
      build a dictionary from tuples
      unpack tuples

Dicts:

      a_dict = {'I hate':'you', 'You should':'leave'}
      keys()
      items()
      hasvalues()
      _key()
      'never' in a_dict
      del a_dict['me']
      a_dict.clear()

Ok enough by me do the rest on your own!
use dir() to get built in functions***
Sets:11

_____add()
       clear()
       copy()
       difference()
       discard()
       intersection()
       issubset()
       pop()
       remove()
       union()
       update()

Strings:

_____capitalize()
       casefold()
       center()
       count()
       encode()
       find()
       partition()
       replace()
       split()
       title()
       zfill()

from collections import Counter

_____Fill in yourself

from itertools import * (Bonus: this one is optional, but recommended)
       Fill in yourself

```
flower_orders=['W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B
','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/
R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B','W/R/B',
'W/R/B','W/R/B','W/R/B','W/R','W/R','W/R','W/R','W/R','W/R','W/R','W/R','W/R'
,'W/R','W/R','W/R','W/R','W/R','W/R','W/R','R/V/Y','R/V/Y','R/V/Y','R/V/Y','R
/V/Y','R/V/Y','R/V/Y','R/V/Y','R/V/Y','R/V/Y','W/R/V','W/R/V','W/R/V','W/R/V'
,'W/R/V','W/R/V','W/R/V','W/R/V','W/R/V','W/R/V','W/N/R/V','W/N/R/V','W/N/R/V
','W/N/R/V','W/N/R/V','W/N/R/V','W/N/R/V','W/N/R/V','W/R/B/Y','W/R/B/Y','W/R/
B/Y','W/R/B/Y','W/R/B/Y','W/R/B/Y','B/Y','B/Y','B/Y','B/Y','B/Y','R/B/Y','R/B
/Y','R/B/Y','R/B/Y','R/B/Y','W/N/R/B/V/Y','W/N/R/B/V/Y','W/N/R/B/V/Y','W/N/R/
B/V/Y','W/N/R/B/V/Y','W/G','W/G','W/G','W/G','R/Y','R/Y','R/Y','R/Y','N/R/V/Y
','N/R/V/Y','N/R/V/Y','N/R/V/Y','W/R/B/V','W/R/B/V','W/R/B/V','W/R/B/V','W/N/
```

```
R/V/Y','W/N/R/V/Y','W/N/R/V/Y','W/N/R/V/Y','N/R/Y','N/R/Y','N/R/Y','W/V/O','W
/V/O','W/V/O','W/N/R/Y','W/N/R/Y','W/N/R/Y','R/B/V/Y','R/B/V/Y','R/B/V/Y','W/
R/V/Y','W/R/V/Y','W/R/V/Y','W/R/B/V/Y','W/R/B/V/Y','W/R/B/V/Y','W/N/R/B/Y','W
/N/R/B/Y','W/N/R/B/Y','R/G','R/G','B/V/Y','B/V/Y','N/B/Y','N/B/Y','W/B/Y','W/
B/Y','W/N/B','W/N/B','W/N/R','W/N/R','W/N/B/Y','W/N/B/Y','W/B/V/Y','W/B/V/Y',
'W/N/R/B/V/Y/G/M','W/N/R/B/V/Y/G/M','B/R','N/R','V/Y','V','N/R/V','N/V/Y','R/
B/O','W/B/V','W/V/Y','W/N/R/B','W/N/R/O','W/N/R/G','W/N/V/Y','W/N/Y/M','N/R/B
/Y','N/B/V/Y','R/V/Y/O','W/B/V/M','W/B/V/O','N/R/B/Y/M','N/R/V/O/M','W/N/R/Y/
G','N/R/B/V/Y','W/R/B/V/Y/P','W/N/R/B/Y/G','W/N/R/B/V/O/M','W/N/R/B/V/Y/M','W
/N/B/V/Y/G/M','W/N/B/V/V/Y/P']
```

1. Build a counter object and use the counter and confirm they have the same values.
2. Count how many objects have color W in them.
3. Make histogram of colors
Hint from JohnP - Itertools has a permutation function that might help with these next two.
4. Rank the pairs of colors in each order regardless of how many colors are in an order.
5. Rank the triplets of colors in each order regardless of how many colors are in an order.
6. Make dictionary color for keys and values are what other colors it is ordered with.
7. Make a graph showing the probability of having an edge between two colors based on how often they co-occur. (a numpy square matrix)
8. Make 10 business questions related to the questions we asked above.

```
dead_men_tell_tales =
['Four score and seven years ago our fathers brought forth on this',
'continent a new nation, conceived in liberty and dedicated to the',
'proposition that all men are created equal. Now we are engaged in',
'a great civil war, testing whether that nation or any nation so',
'conceived and so dedicated can long endure. We are met on a great',
'battlefield of that war. We have come to dedicate a portion of',
'that field as a final resting-place for those who here gave their',
'lives that that nation might live. It is altogether fitting and',
'proper that we should do this. But in a larger sense, we cannot',
'dedicate, we cannot consecrate, we cannot hallow this ground.',
'The brave men, living and dead who struggled here have consecrated',
'it far above our poor power to add or detract. The world will',
'little note nor long remember what we say here, but it can never',
'forget what they did here. It is for us the living rather to be',
'dedicated here to the unfinished work which they who fought here',
'have thus far so nobly advanced. It is rather for us to be here',
```

```
'dedicated to the great task remaining before us--that from these',
'honored dead we take increased devotion to that cause for which',
'they gave the last full measure of devotion--that we here highly',
'resolve that these dead shall not have died in vain, that this',
'nation under God shall have a new birth of freedom, and that',
'government of the people, by the people, for the people shall',
'not perish from the earth.']
```

1. Join everything
2. Remove spaces
3. Occurrence probabilities for letters
4. Tell me transition probabilities for every letter pairs
5. Make a 26x26 graph of 4. in numpy
#optional
6. plot graph of transition probabilities from letter to letter

Unrelated:
7. Flatten a nested list
Cool intro python resources:

   https://thomas-cokelaer.info/tutorials/python/index.html