

Functions

A block of code that is only run when called

We can send parameters to the function and the function can return a value, or more than one value, by using the return function

We define a function using the def keyword

We must send the function the exact number of parameters it's expecting

We can make the number of parameters arbitrary by putting "*" in front of parameter

- this turns the parameters into a tuple
- examples <https://realpython.com/python-kwargs-and-args/>
- args is just a name, you could use anything else
- '*' is unpacking operator
 - can do other cool things like fill variables flexibly in assignment

```
my_list=[1,2,3,4,5,6]
a,*b,c=my_list
```

We can send parameters with key = value syntax so the order doesn't matter if it gets mixed up

We can make the number of keyword parameters arbitrary by putting "**" in front of parameter

- keyword parameters sent in as dictionary with keys and values.
- don't have to name it kwargs, could be any name
- '**' is unpacking operator

We can have regular arguments, *args, and **kwargs in that same order

- example <https://www.geeksforgeeks.org/args-kwargs-python/>
- example <https://towardsdatascience.com/10-examples-to-master-args-and-kwargs-in-python-6f1e8cc30749>

```
def arg_printer(a, b, *args, option=True, **kwargs):
    print(a, b)
    print(args)
    print(option)
    print(kwargs)
```

•

```
arg_printer(1, 4, 6, 5, param1=5, param2=6)
1 4
(6, 5)
True
{'param1': 5, 'param2': 6}
```

It's possible to give the function a default value if nothing is sent.

- define it where we define args for function

If your function is empty, use the pass keyword to avoid throwing an error

Recursion is possible. This is when a function calls itself.

- caution! Don't get caught in an infinite loop!

Variables inside of functions are local unless we define them with the global keyword

We can also use lambda functions

- lambda function just means "here comes an anonymous function"
 - example: lets say we want to sort a list of words by the number of unique letters
 - `list_of_words.sort(key=lambda x: len(set(list(x))))`