functions

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