Chapter 4: Function

* Def: define function, like an if statement
  + “def” isn’t starting code, its store a code. The word following that is what your naming the function
  + Each indentation below is what you would like to run when you invoke “word”
    - i.e def word():
      * print(‘Hello’)
* Two types of funtctions;
  + Built in: functions provided by/installed with python
  + Defined by us: creating new reserved words that extend python language that we can call later as a function
* Call/invoke are the same
* Return value: what you get after executing an expression
  + You can assign a built in function an argument inside of (), and assign a value to it, so when you call that value it will produce an output
* A Function is some stored code that we use. A function takes some input and produces an output

Building our own Functions

* When defining, we are not executing, simply storing for later
* Once you def a function, you can call/invoke as many times as youd like. But if you don’t call it , it won’t execute
* An argument is a value we pass into he function as its input when we call the function
* We use arguments so we can direct the function to do different kinds of work when we call it at different times
* We put arguments in parentheses after the name of the function
* Return is the residual value, what shows up when the functions all done
* A fruitful function is one that produces a result (or return value)
* The return statement ends the function execution and ‘sends back’ the result of the function