Chapter 2:

1. Why are functions advantageous?
   1. Replicable code, functions are kind of like macros in SAS – saves having to re-write code
2. When does the code in a function execute: when the function is defined or when the function is called?
   1. When the function is called
3. What statement creates a function?
   1. Def
4. What is the difference between a function and a function call?
   1. A function is what the actual program/logic of the function is. The function call is the little bit of code that calls that logic
5. How many global scopes are there in a Python program? How many local scopes?
   1. Not sure. 1 global scope, 1 local scope?
6. What happens to variables in a local scope when the function call returns?
   1. When the function call returns, all local variables are deleted/reset
7. What is a return value? Can a return value be part of an expression?
   1. A function returns a value, it can be anything a number or an expression
8. If a function does not have a return statement, what is the return value of a call to that function?
   1. The return is the last statement of the function
9. How can you force a variable in a function to refer to the global variable?
   1. The global keyword before you declare the variable
10. What is the data type of None?
    1. Null?