Chapter 2:

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

p245