

# Unit 3 - Functions and Parameters - Study Guide

## Functions and Parameters

- Why do we need functions?
  - They let us reuse code without having to copy+paste.
  - They let us save space in our programs.
  - They make our programs more readable.
- The 5 things necessary to define a function:
  - The `def` keyword
    - Short for define, tells Python that a function is being created.
  - The name of the function
    - Follows the same rules as naming variables.
    - Should be related to what the function does.
  - The Parameter List
    - Parentheses surround the parameter list.
    - Parameters are named here
      - These variables are initialized when the function is called.
  - The colon
    - Tells Python that the next line(s) will be indented.
  - The body
    - The code that is run when the function is called.
    - Needs to be indented.
- Parameters
  - Variables created within the definition of a function.
  - The way we get information into the function .
  - There is no limit to how many parameters can be added to a function definition - you can even have no parameters! When there is more than one, they should be separated by commas.
  - The number of parameters in the definition must match the number in the call.
  - We can make them optional by giving them a **default value**
    - `parameter_name = default_value`
    - This default value will be given to the parameter when one isn't provided during the function call.
    - Optional parameters must come *after* required parameters.
  - When a function is called, the parameters in the function call are placed into the defined parameters in order.

## Functions and Return Values

- Some functions will *return* a value when they are called. This means that they will equal something when they do their thing!
- We can make our functions *return* a value using the `return` keyword, though we don't have to. *Returning* a value is optional.
- When the `return` keyword is run, the value after the keyword will immediately be returned, and the function will stop running.
- Multiple values can be returned from a function by placing those values one after another with commas after the `return` keyword.

## Namespaces & Scope

- A variable's **namespace** defines where that variable is accessible from. The **scope** of a variable determines which **namespace** it belongs to!
- Variables that are not defined inside of a function are **global variables** - they can be seen/accessed from anywhere in the program.
- Variables that **are** defined inside of a function are **local** to that function - they can only be seen/accessed from within that function.
  - This includes parameters! Parameters will be local to the function they are a part of.
- Variables in different namespaces can have the same name. If they do, they still won't affect one another.

## APIs

- API is short for **Application Programming Interface**.
- APIs allow us as programmers to make use of code written by other people without needing to fully understand how it works - we just need to know how to use it!
- There are APIs for all sorts of different applications, from graphing data to browsing the internet!
- APIs enable the interfacing of hardware and software, the interfacing of different pieces of software, as well as simply allowing different devices to communicate with one another!