1. Functions in python are a piece of code used to gather a set of instructions that you want to use repeatedly and to carry out a specific task. Functions make your code cleaner (if the complexity of the code is complicated or long) and to help you find your way around your code. Also, a function can or can not return one or more values.
2. Python’s return value of a function is a way to send objects back to you main code. After using the return value in a function, the function will end , which means the return statement is the last thing to write in your function, and all the things written after it will not be done.

For example,

def plus(number1, number2) -> int :

return number1 + number2

The function sums up the inputs, which are number1 and number2 , and their return value would be the sum of number1 and number2.

1. When we call a function, a chunk of memory holds the formal parameter values and function local variables.

When you are passing by value, you are making a copy in memory of the actual parameter's **value** that is passed in, a copy of the contents of the actual parameter. That’s why we pass by value when we want to use the parameters for some calculation, and not for changing it in the program itself.

Pass by reference is the complete opposite, like its name, we are making a copy of **the address** of the actual parameter, so when we want to change the parameter completely ( also in the program itself and not only in the function ) , we would use pass by reference.

If you aren't going to change a variable, you use pass by value. But if you are passing something in that uses a lot of memory,like passing an object or passing an array even if you aren't changing it, you use pass by reference .

1. Scope defines the accessibility of the python object, so if i want to access a particular variable in my code, i must define the scope as it can’t be accessed from any place in the whole program, and not only for variables, could be also for functions, objects and so on.

There are 2 types of scopes, local and global scopes.

Like we said, a scoper is a variable that is only available from the space that it was created , so for local scope, the variable that was created inside the function, belongs to the local scope of that function, and could only be used in that scope, inside the function only and not in the main program. So for example if i have a function inside a function, even though function a is inside function b, if declared a variable with only the scope of function a , you couldn’t use the variable in function b.

On the other hand, a global scope is a variable that is created in the main program and belongs to the global scope of the code, and includes any scope, both global and local and be available to all your code.

9. Recursion is used to solve complex issues by breaking them down into less complex ones using iteration of the function itself. Recursion is a process by which a function calls itself specifically or in a roundabout way. The comparing function is called a recursive function.

In python, we would rather use loops and not recursion, because recursion has the overhead of keeping up and updating the stack. Loops doesn't use the stack, and doesn’t require calling the function several times.