function

A function is a block of code which only runs when it is called.

You can pass data, known as parameters, into a function.

A function can return data as a result.

In Python a function is defined using the def keyword

Arguments

Information can be passed into functions as arguments.

Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma.

Parameters

By default, a function must be called with the correct number of arguments. Meaning that if your function expects 2 arguments, you have to call the function with 2 arguments, not more, and not less.

Arbitrary Arguments, *args

If you do not know how many arguments that will be passed into your function, add a * before the parameter name in the function definition.

Arbitrary Arguments are often shortened to *args

Arbitrary Keyword Arguments, **kwargs

If you do not know how many keyword arguments that will be passed into your function, add two asterisk: ** before the parameter name in the function definition.

This way the function will receive a dictionary of arguments, and can access the items accordingly

Higher order function:

When we are using function as parameter to the function called as higher order function

lambda

A lambda function is a small anonymous function. Anonymous function means function without name.

A lambda function can take any number of arguments, but can only have one expression.

Filter:

The filter function expects two arguments: function object and an iterable.

Function object returns a boolean value and is called for each element of the iterable.

 filter returns only those elements for which the function object returns True.

map

map functions expect a function object and any number of iterables, such as list, dictionary, etc.

It executes the function object for each element in the sequence and returns a list of the elements modified by the function object.

-