

Usage of *args:

*args and **kwargs are mostly used in function definitions. *args and **kwargs allow you to pass an unspecified number of arguments to a function, so when writing the function definition, you do not need to know how many arguments will be passed to your function. *args is used to send a **non-keyworded** variable length argument list to the function. Here's an example to help you get a clear idea in notebook.

Usage of **kwargs:

kwargs allows you to pass **keyworded variable length of arguments to a function. You should use **kwargs if you want to handle **named arguments** in a function. Here is an example to get you going with it in notebook.

When to use them?

It really depends on what your requirements are. The most common use case is when making function decorators (discussed in another chapter).

Moreover it can be used in monkey patching as well. Monkey patching means modifying some code at runtime. Consider that you have a class with a function called `get_info` which calls an API and returns the response data.

If we want to test it we can replace the API call with some test data. For instance