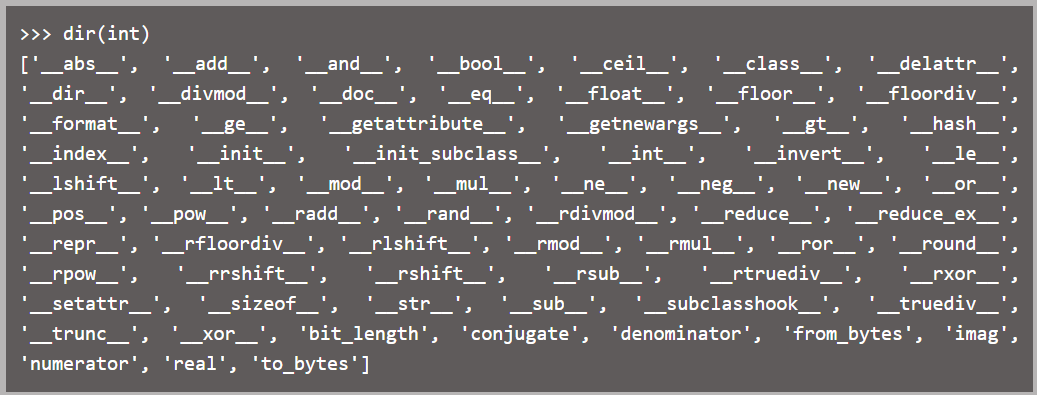
Report

# Dunder Methods

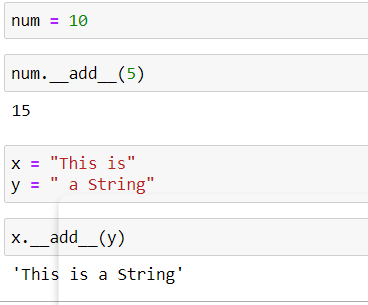
Dunder methods are special methods that start and end with double underscores. They are not meant to be invoked directly by the programmer, but instead are invoked internally from the class on a certain action.

For example, when two numbers are added using the + operator, the ***\_\_add\_\_()*** method is invoked internally.

Built-in python classes define a lot of dunder methods. Using the ***dir()*** function, the dunder methods inherited by a class are shown. So for example, when the ***dir()*** is used on the *int* class the following is shown.



Dunder methods are most frequently used to define overloaded behaviors of predefined operators in python. For example, the + operator is by default used along with numeric values, but it can also be used as a concatenation operator in string, list and tuple classes. This means that the + operator is overloaded.

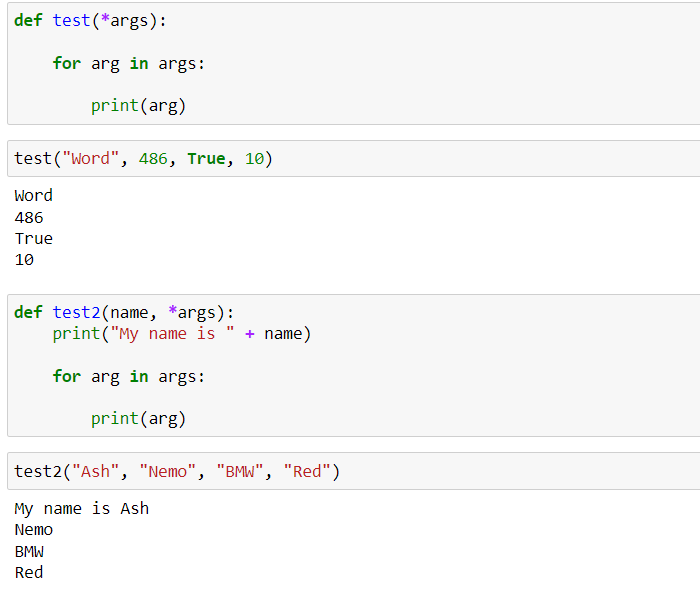


# \*args

\*args is used in python to pass a varying number of arguments to a function. It is used to pass non-key worded argument list of varying length.

\*args allows the addition of arguments to a function other than the formal ones previously defined, any additional arguments get added to the argument list *args*.

The use of \* make the variable attached to it an iterable object.



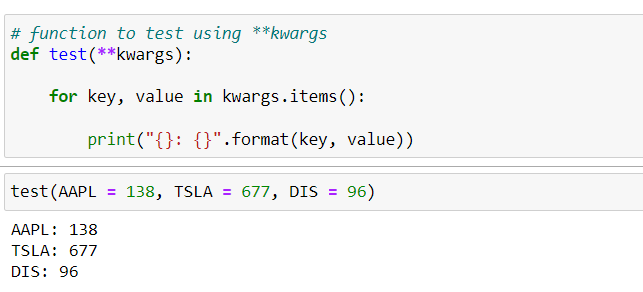
# \*\*kwargs

\*\*kwargs is used to pass a key worded argument list of varying length to a function.

kwargs is used with the double star because the double star allows the passing through of keyword arguments.

A keyword argument is when a name is provided for an argument while passing it to a function.

\*\*kwargs can be thought of as a dictionary that maps each keyword to the value passed alongside it.



# Encapsulation

Encapsulation is one of the fundamental concepts in object-oriented programming. It describes the idea of wrapping data and the methods that work on data within one unit. This puts restrictions on accessing variables and methods directly and can prevent the accidental modification of data. To prevent accidental change, an object’s variable can only be changed by an object’s method. Those types of variables are known as private variables.

A class is an example of encapsulation as it encapsulates all the data that is member functions, variables, etc.

# Abstraction

**Abstraction** is defined as a process of handling complexity by hiding unnecessary information from the user. This is one of the coreconcepts of object-oriented programming that enables the user to implement even more complex logic on top of the provided abstraction without understanding or even thinking about all the hidden back-end complexity.

Abstraction is achieved by encapsulation, where the complex process’ code and data are wrapped into a unit that is given a name that can be called.