

Magic / Dunder methods in python

There are special methods that you can define in your classes and when invoked, they give you a powerful way to manipulate objects and their behavior.

Magic methods, also known as “dunders” from the double underscore surrounding their names are powerful tools that allow you to customize the behavior of the your classes. They are used to implement special methods such as the addition subtraction and comparison operators, as will as some more advanced techniques like descriptors and properties.

Let's take a look at some of the most commonly used magic methods in python.

`__init__` method

The init method is a special method that is automatically invoked when you create a new instance of a class. This method is responsible for setting up the object's initial state, and it is where you would typically define any

`__str__` and `__repr__` methods

The str and repr methods are both used to convert an object to a string representation. The str method is used when you want to get a string representation of an object that can be used to recreate the object.

`__len__` method

The len method is used get the length of an object. This is useful when you want to be able to find the size of a data structure, such as a list or dictionary.

`__call__` method

The call method id used to make an object callable meaning that you can pass ist as a parameter to a function and it will be executed when the

function is called. This is an incredibly powerful tools that allow you to create objects like functions.

Source Code

```
class programmer:
    name="Munawar"

    def __len__(self):
        i=0
        for c in self.name:
            i=i+1
        return i

p=programmer()
print(p.name)
print(len(p))
```

Thank You