20 Most Used Magic Methods in Python OOP

blog.DailyDoseofDS.com

Magic Method	Syntax	Usage/Description
__new__	__new__(cls, *args, **kwags):	Invoked before __init__ to allocate memory to object
__init__	__init__(self, *args, **kwags):	Invoked after __new__ to initialise the object
__str__	__str__(self):	Invoked when str(obj) or print(obj) is used
__int__	__int__(self):	Invoked when int(obj) is used
__len__	__len__(self):	Invoked when len(obj) is used
__call__	__call__(self, *args, **kwags):	Invoked when class object is called as a function: obj()
__getitem__	__getitem__(self, key):	Invoked when object is indexed: obj[key]
__setitem__	__getitem__(self, key, value):	Invoked when object is indexed and value is set: obj[key]=value
__delitem__	__delitem__(self, key):	Invoked when object's index is deleted: del obj[key]
__contains__	__contains__(self, item):	Invoked when the in operator is used: item in obj
__bool__	__bool__(self):	Invoked when object is used in boolean context: if obj or bool(obj)
__iter__	__iter__(self):	Invoked when object is iterated: for x in obj
__eq__	__eq__(self, other):	Invoked when == operator is used to compare two objects: obj1 == obj2
__ne__	__ne__(self, other):	Invoked when != operator is used to compare two objects: obj1 != obj2
__gt__	__gt__(self, other):	Invoked when > operator is used to compare two objects: obj1 > obj2
__add__	__add__(self, other):	Invoked when two objects are added: obj1 + obj2
__mul__	__mul__(self, other):	Invoked when two objects are multiplied: obj1 * obj2
__abs__	__abs__(self):	Invoked to compute absolute value of object: abs(obj)
__neg__	__neg__(self):	Invoked when unary operator - is used on an object: -obj
__invert__	__invert__(self):	Invoked when ~(tilde) operator is used to invert an object: ~obj