A dunder method:

-is a method (that Python knows about), which has two underscores before it and two underscores after it.

-It's a contract between the person who implemented a certain class and the Python interpreter, which knows when to call that particular dunder method.

Some examples of dunder methids: -

_abs__
__add__
__aenter__
__aexit__
__aiter__
__and__
__anext__
__await__
__bool__
__bytes__
__call__
__class__
__cmp__
__complex__
__contains__
__delattr__
__delete__
__delitem__
__delslice__
__dir__
__div__
__divmod__
__enter__
__eq__
__exit__
__float__
__floordiv__
__format__

__fspath__
__ge__
__get__
__getattribute__
__getitem__
__getnewargs__
__getslice__
__gt__
__hash__
__iadd__
__iand__
__import__
__imul__
__index__
__init__
__init_subclass__
__instancecheck__
__int__
__invert__
__ior__
__isub__
__iter__
__ixor__
__le__
__len__
__lshift__
__lt__
__mod__
__mul__
__ne__
__neg__
__new__
__next__
__nonzero__
__or__
__pos__
__pow__
__prepare__
__radd__
__rand__

__rdiv__
__rdivmod__
__reduce__
__reduce_ex__
__repr__
__reversed__
__rfloordiv__
__rlshift__
__rmod__
__rmul__
__ror__
__round__
__rpow__
__rrshift__
__rshift__
__rsub__
__rtruediv__
__rxor__
__set__
__setattr__
__setitem__
__setslice__
__sizeof__
__str__
__sub__
__subclasscheck__
__subclasses__
__truediv__
__xor__