Python is an ***object-oriented programming (OOP) language***.

***Everything*** is in Python ***treated as an object***. Every object belongs to its class.

Including variable, function, list, tuple, dictionary, set, etc.

An object is a real-life entity. An object is an instance of a class. An object is the collection of data/ attribute and functions/ methods.

A class is a collection of objects. A class is a blueprint/ template through which objects are created. A class is like an object contructor.

\_\_init\_\_() Method

All classes have a function called \_\_init\_\_(), which is always executed when the class is being initiated.

The \_\_init\_\_() function is called automatically every time the class is being used to create a new object.

self parameter

The self parameter is a reference to the current instance of the class, and is used to access variables that belong to the class.

It does not have to be named self , you can call it whatever you like, but it has to be the first parameter of any function in the class

Object.\_\_dict\_\_ => to display all the attributes an object. | returns a dictionary of all the attribute

hasattr(object, attribute) => to check object has an attribute or not | return boolean value

getattr(object, attribute) => return the value of the attribute of that object

getattr(object, attribute, third argument) => the function return third argument as default value when the attribute is not present for that object

delattr(object, attribute) => to delete an attribute of an object

class attribute => common attribute of all the objects such as organization name

instance/ object attribute => unique attribute of an object such as student name and roll number

if an object has instance attribute & class attribute as same so the attribute which going to print is instance attribute if there is no instance attribute then class attribute is going to print

internally how a method is called by python

ClassName.Function(ObjectName) is same as ObjectName.Function()

self = Object