# OOPS

In Python, object-oriented Programming (OOPs) uses objects and classes in programming.

It aims to implement real-world entities like inheritance, polymorphisms, encapsulation, etc. in the programming.

The main concept of OOPs is to bind the data and the functions that work on that together as a single unit so that no other part of the code can access this data.

***Main concepts in OOPS***

* Class
* Objects
* Polymorphism
* Encapsulation
* Inheritance
* Data abstraction

## Class

A class is a collection of objects. A class contains the blueprints or the prototype from which the objects are being created. It is a logical entity that contains some attributes and methods.

Syntax:

class Classname:

statement 1

:

:

Statement n

### ****Self:****

1. Class methods must have an extra first parameter in the method definition. We do not give a value for this parameter when we call the method, Python provides it
2. If we have a method that takes no arguments, then we still have to have one argument.

### ****The \_\_init\_\_ constructor/method****

It is run as soon as an object of a class is instantiated. The method is useful to do any initialization you want to do with your object.

**What is an Object ?**

Everything is in Python treated as an object, including variable, function, list, tuple, dictionary, set, etc.

*Every object belongs to its class.*

For example - An integer variable belongs to integer class. An object is a real-life entity.