Module 8: OOP

Class:

- Class is a blueprint for the object.
- It encapsulates data(variables) & methods (functions) together.

Object

- Object is called the instance of class.

Encapsulation & Instantiation:

- Process of creating class is called encapsulation and process of creating object is called instantiation.

Initializer:

- Initializer is a special method that is used to initialize the instance variables of a class ➔ __init__()
- We create instance variables and initialize them in initializer.
- The first parameter is self which contains the memory address of the instance.
- Class can either have: Default Initializer OR Parameterized Initializer

---

Inheritance:

- Inheritance is the mechanism of deriving one class from another such that the new derived class can inherit all the members of the existing base class.
- The syntax of inheritance is:
  - class SubClass(SuperClass):
- Advantage of inheritance is reusability and hence more code less time.

Types of Inheritance:

- Python supports following type on inheritance :
  - Single
  - Multilevel
  - Hierarchical
  - Hybrid
  - Multiple

Polymorphism:

- If a variable, object or method exhibits different behavior in different contexts then it is called polymorphism.

Method Overriding:

- In Method overriding we override the method of super class in the subclass.
- We do this coz we want to change the functionality for the same method in the subclass.

Operator Overloading:

- + operator internally calls __add__() method.
- By overriding this method we can make + operator to work with user defined objects also.

Binary Operators:

OPERATOR	MAGIC METHOD
+	__add__(self, other)
–	__sub__(self, other)
*	__mul__(self, other)
/	__truediv__(self, other)
//	__floordiv__(self, other)
%	__mod__(self, other)
**	__pow__(self, other)

Assignment Operators :

OPERATOR	MAGIC METHOD
-=	__isub__(self, other)
+=	__iadd__(self, other)
*=	__imul__(self, other)
/=	__idiv__(self, other)
//=	__ifloordiv__(self, other)
%=	__imod__(self, other)
**=	__ipow__(self, other)

Comparison Operators :

OPERATOR	MAGIC METHOD
<	__lt__(self, other)
>	__gt__(self, other)
<=	__le__(self, other)
>=	__ge__(self, other)
==	__eq__(self, other)
!=	__ne__(self, other)

Unary Operators :

OPERATOR	MAGIC METHOD
–	__neg__(self, other)
+	__pos__(self, other)
~	__invert__(self, other)