Advanced Python

Subject: Object-oriented Programming (OOP) in Python

Lecturer : Reza Akbari Movahed

Object-oriented Programming (OOP) in Python

- OOP is a programming paradigm that uses objects and classes in programming.
- It aims to implement real-world entities.
- The main concept of OOPs is to bind the data and the functions that work on that together as a single unit.

Object-oriented Programming (OOP) in Python

Object-oriented Programming (OOP) in Python

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.

Class Definition syntax

```
class ClassName:
    # Statement-1
    .
    .
    .
    # Statement-N
```

```
# Python3 program to
# demonstrate defining
# a class

class Dog:
    pass
```

Object-oriented Programming (OOP) in Python

Object

- The object is an entity that has a state and behavior associated with it.
- It may be any real-world object like a mouse, keyboard, chair, table, pen, etc.
- Integers, strings, floating-point numbers, even arrays, and dictionaries, are all objects.

An object consists of :

- **State:** It is represented by the attributes of an object. It also reflects the properties of an object.
- **Behavior:** It is represented by the methods of an object. It also reflects the response of an object to other objects.
- **Identity:** It gives a unique name to an object and enables one object to interact with other objects.

Object-oriented Programming (OOP) in Python

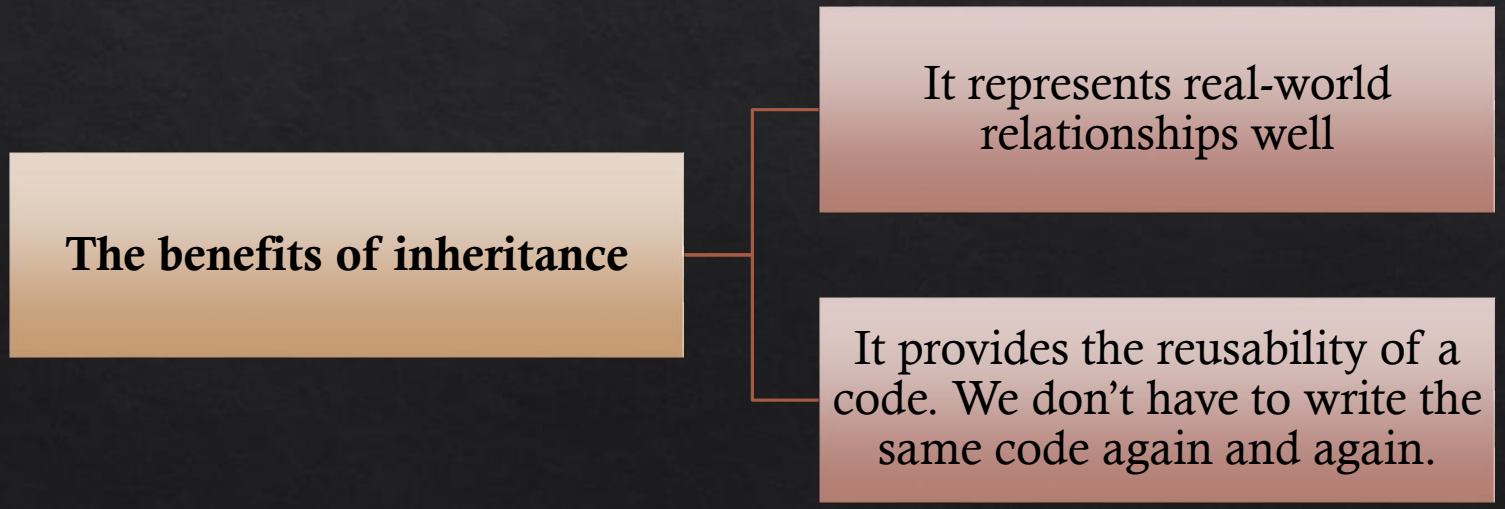
Basic keywords of OOP in Python

self	When we call a method of an object as myobject.method(arg1, arg2), this is automatically converted by Python into MyClass.method(myobject, arg1, arg2) – this is all the special self is about.
The __init__ 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.

Object-oriented Programming (OOP) in Python

Inheritance

- Inheritance is the capability of one class to derive or inherit the properties from another class.
- The class that derives properties is called the derived class or **child class**.
- The class from which the properties are being derived is called the **base class** or **parent class**.



Object-oriented Programming (OOP) in Python

A notable tip of inheritance

- It is transitive in nature, which means that if class B inherits from another class A, then all the subclasses of B would automatically inherit from class A.