**OBJECT ORIENTED PROGRAMMING PROJECT**

**INTRODUCTION**

Object-oriented programming is a programming paradigm that provides a means of structuring programs so that properties and behaviours are bundled into individual objects.

**DEFINITION**

Object Oriented Programming is a way of computer programming using the idea of “objects” to represents data and methods.

**IMPORTANCE**

Object-oriented programming provides flexibility for your codebase through inheritance and polymorphism. Classes and objects (depending on the language) can also share properties and methods through inheritance. The child class or object inherits everything from its parent.

**FEATURES**

Top Features of OOPS

* Inheritance.
* Encapsulation.
* Abstraction.
* Polymorphism.
* Method Overriding.
* Method Overloading.
* Objects.
* Classes.

**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.

## 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.

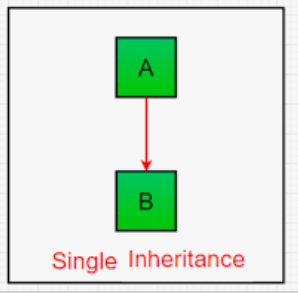
**INHERITANCE**

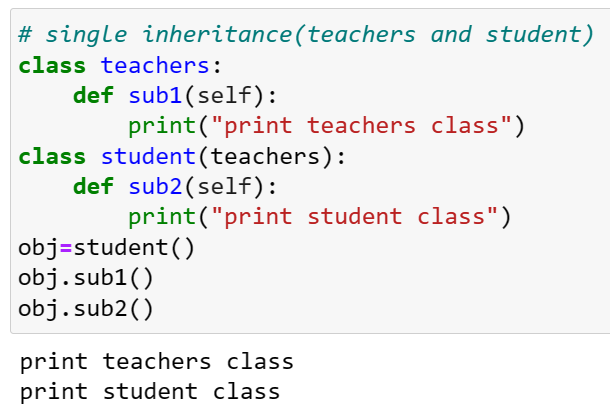
Inheritance allows us to define a class that inherits all the methods and properties from another class. Parent class is the class being inherited from, also called base class. Child class is the class that inherits from another class, also called derived class.

**TYPES OF INHERITANCE**

**1)SINGLE INHERITANCE**

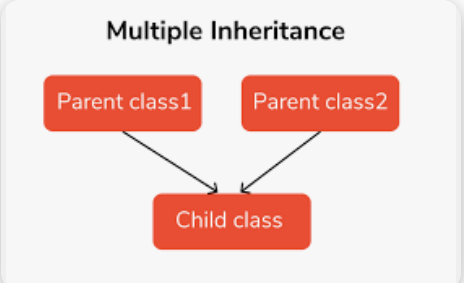
In this type, a child class inherits from a single parent class as shown below. To summarize this example: There is a subclass called Cat that inherits the attributes and methods of the base class Pet. In addition, the class Cat also has its own method, sounds.

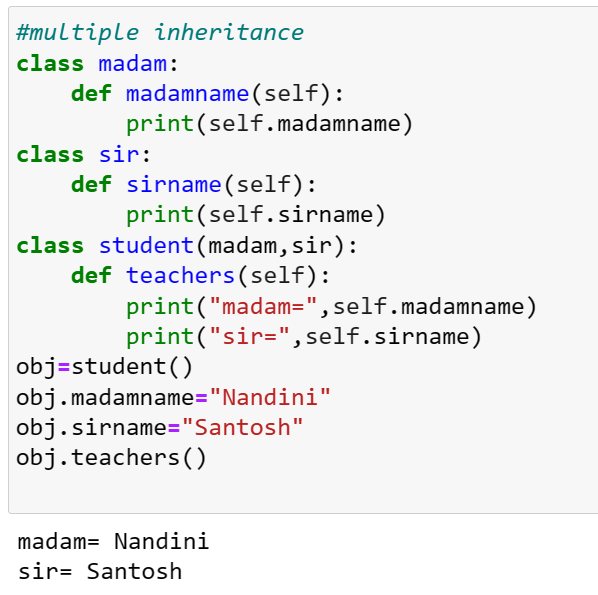




**2)MULTIPLE INHERITANCE**

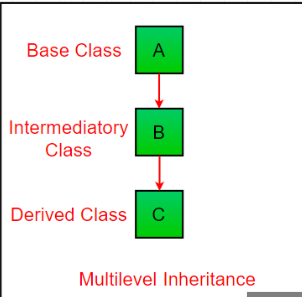
A class can be derived from more than one superclass in Python. This is called multiple inheritance. For example, A class Bat is derived from super classes Mammal and Winged Animal .

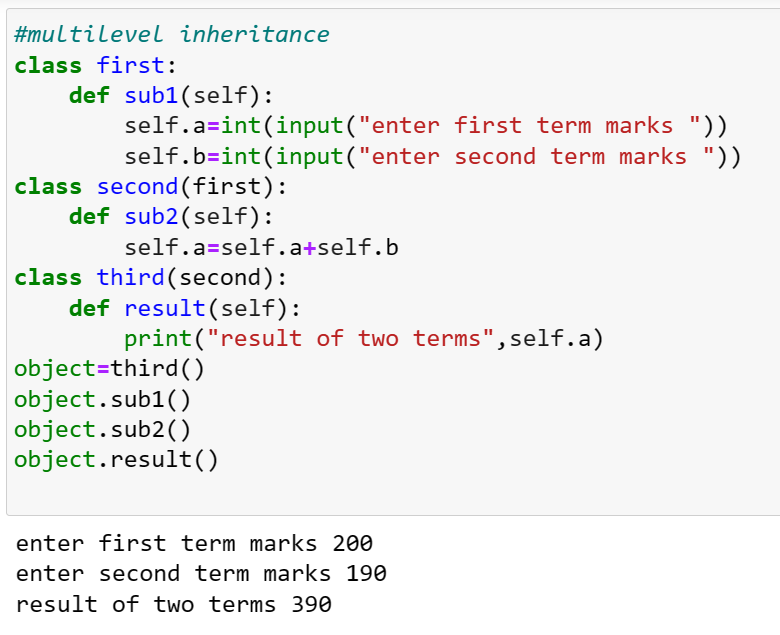




**3)MULTILEVEL INHERITANCE**

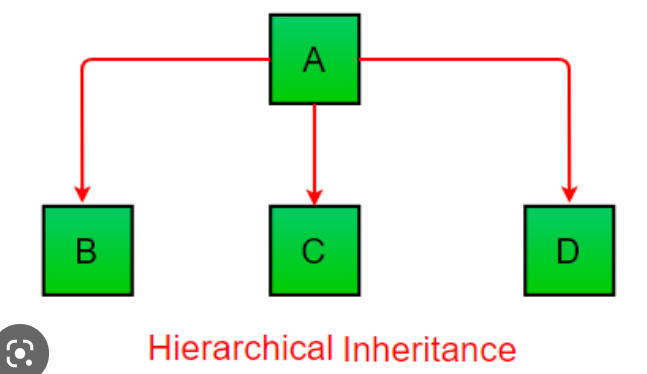
Multilevel Inheritance in Python is a type of Inheritance that involves inheriting a class that has already inherited some other class.

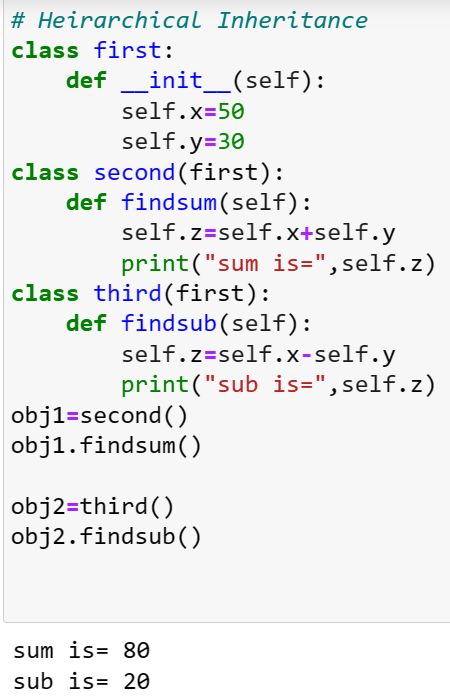




**4)HIERARCHICAL INTERITANCE**

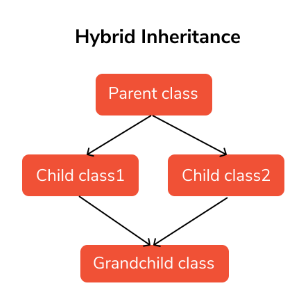
Hierarchical Inheritance If multiple derived classes are created from the same base, this kind of Inheritance is known as hierarchical inheritance. In this instance, we have two base classes as a parent (base) class as well as two children (derived) classes.

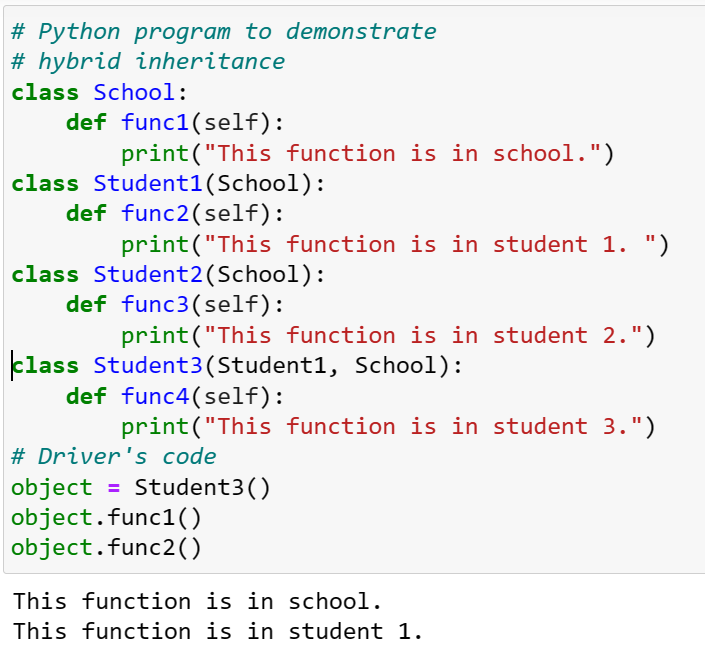




**5)HYBRID INHERITANCE**

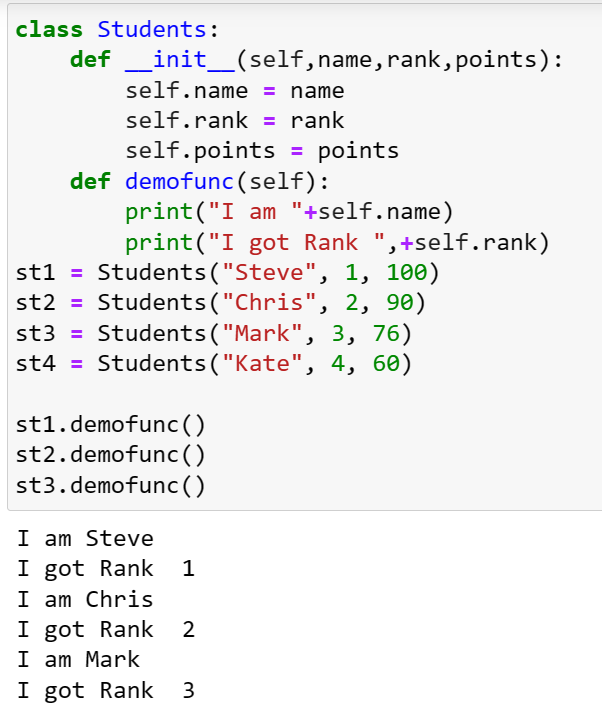
Hybrid inheritance is a combination of multiple inheritance and multilevel inheritance. A class is derived from two classes as in multiple inheritance. However, one of the parent classes is not a base class. It is a derived class.





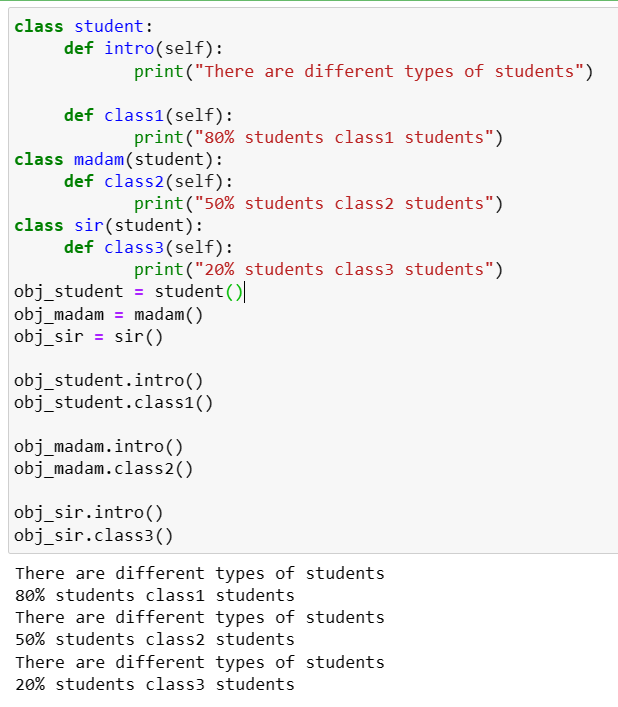
**ENCAPSULATION**

Encapsulation is a mechanism of wrapping the data (variables) and code acting on the data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class.



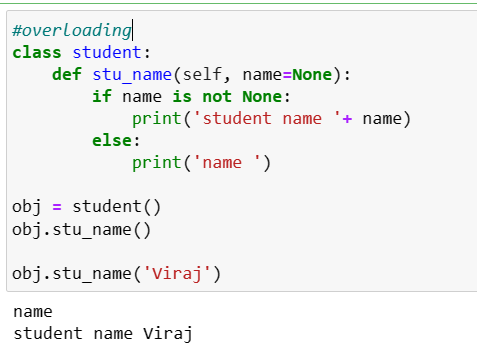
**POLYMORPHISM**

Polymorphism defines the ability to take different forms. Polymorphism in Python allows us to define methods in the child class with the same name as defined in their parent class.



**OVERLOADING**

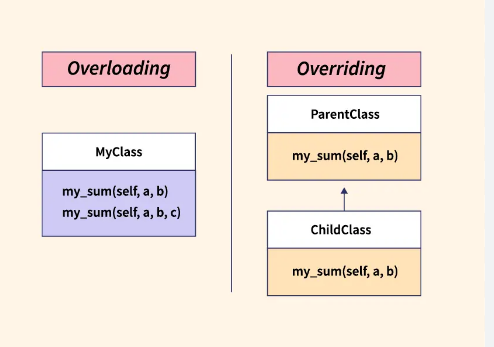
It is one of the important concepts in OOP. Two methods cannot have the same name in Python; hence method overloading is a feature that allows the same operator to have different meanings. Overloading is a method or operator that can do different functionalities with the same name.

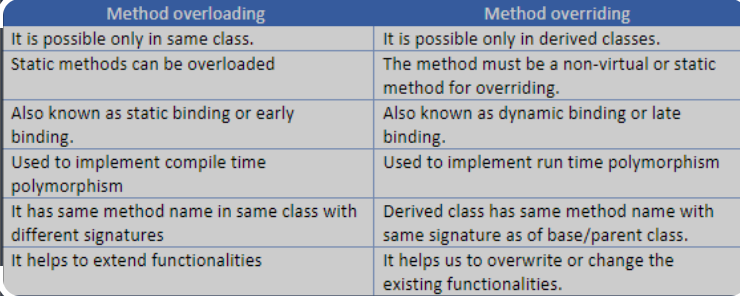


**OVERRIDING**

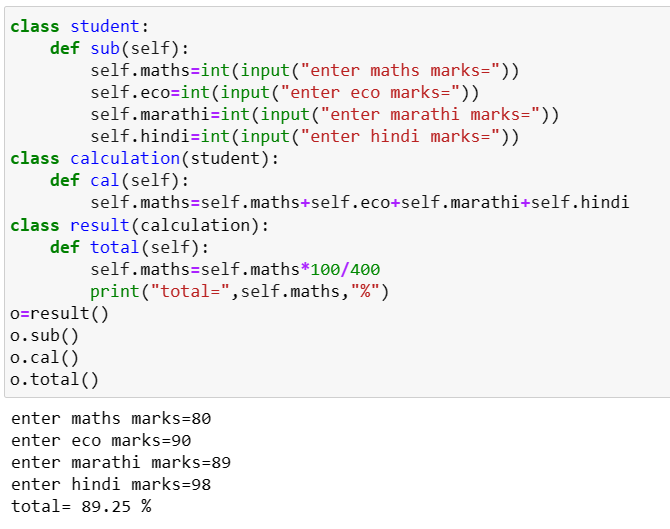
Method overriding in Python is when you have two methods with the same name that each perform different tasks. This is an important feature of inheritance in Python. In method overriding, the child class can change its functions that are defined by its ancestral classes.







**Student total marks and their percentage**



**CLASS SUPER**

The super() function is used to give access to methods and properties of a parent or sibling class. The super() function returns an object that represents the parent class.

