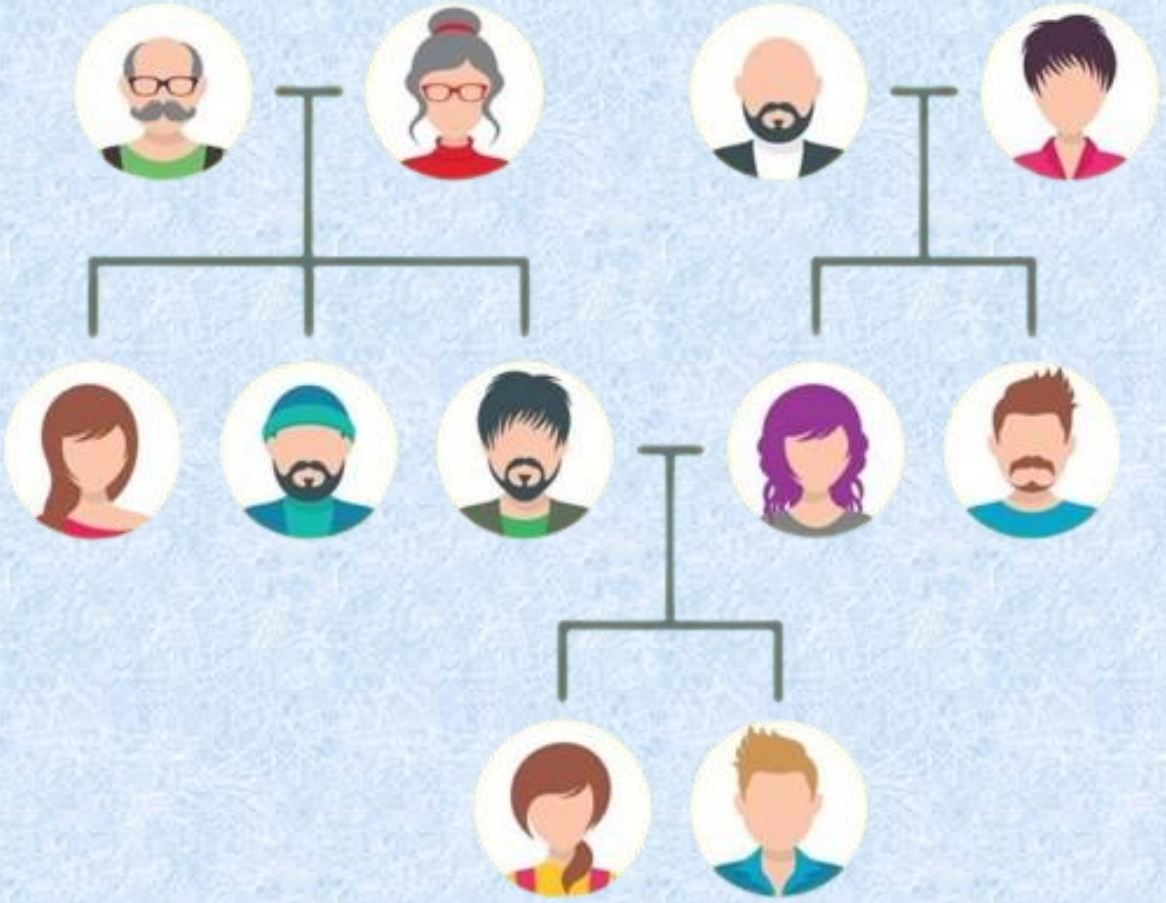
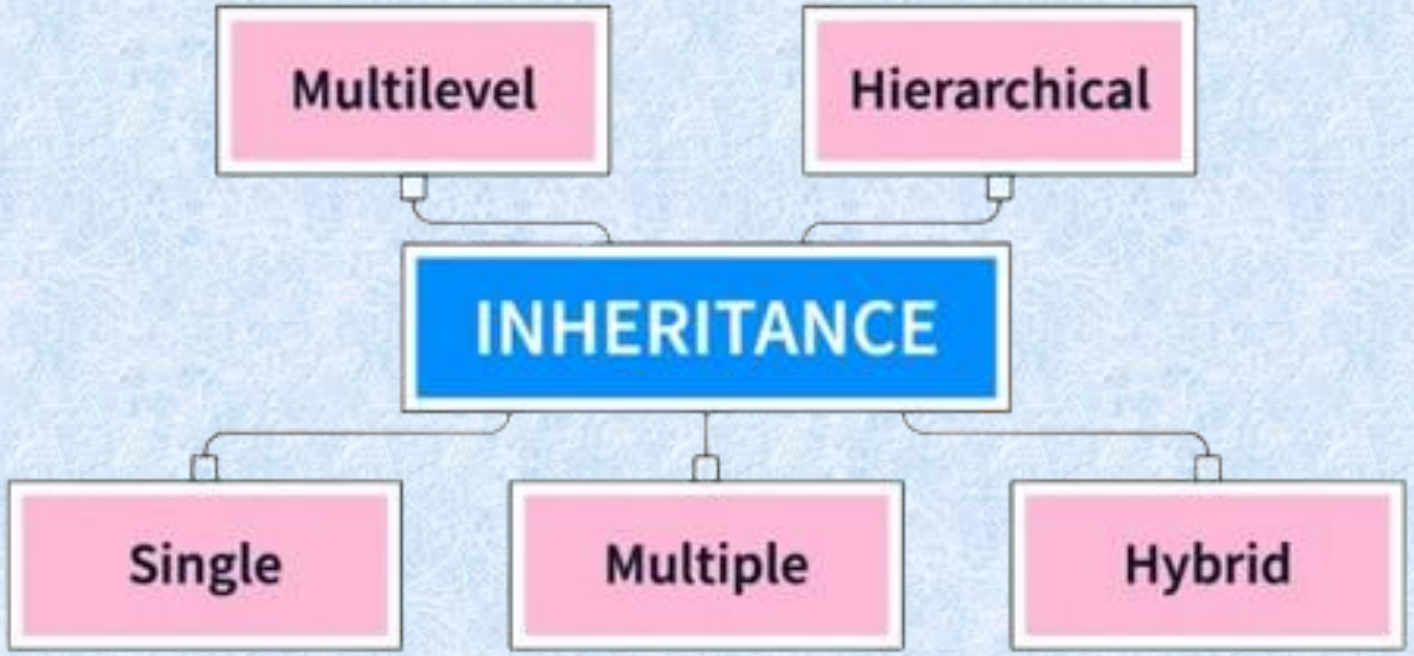
Inheritance in Python



- It’s a Mechanism in Python OOP where a class (derived/child) inherits attributes and methods from another class (base/parent).
- Class whose attributes and methods are inherited by another class is called as Parent class.
- Class that inherits from another class is called as Child class.
- It Facilitates code reuse allowing a class to build on existing features.
- Promotes code reuse and organization by allowing a new class to build upon the existing features of another class.
- The super() function is a built-in function that allows to access the parent class’s methods and attributes in the child class.

Types of Inheritance in Python

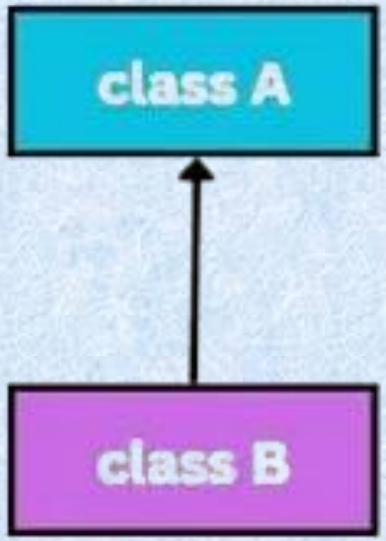


1. Single Inheritance

- Single inheritance enables a derived class to inherit properties from a single parent class only.

Example Code :

```
class Animal:
    def speak(self):
        print("Animal speaks")

class Dog(Animal):
    def bark(self):
        print("Dog barks")

my_dog = Dog()
my_dog.speak()
my_dog.bark()
```

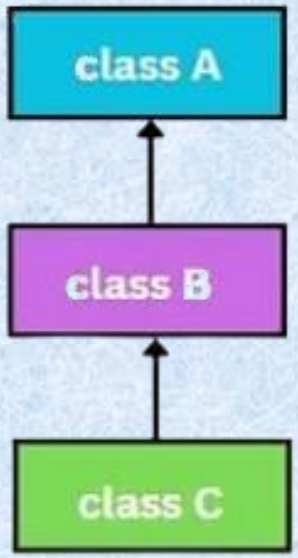


2. Multilevel Inheritance

- In multilevel inheritance, features of the base class and the derived class are further inherited into the new derived class.

Example Code :

```
class Vehicle:
    def start_engine(self):
        print("Engine started")

class Car(Vehicle):
    def drive(self):
        print("Car is driving")

class SportsCar(Car):
    def race(self):
        print("Sports car is racing")

my_sports_car = SportsCar()
my_sports_car.start_engine()
my_sports_car.drive()
my_sports_car.race()
```

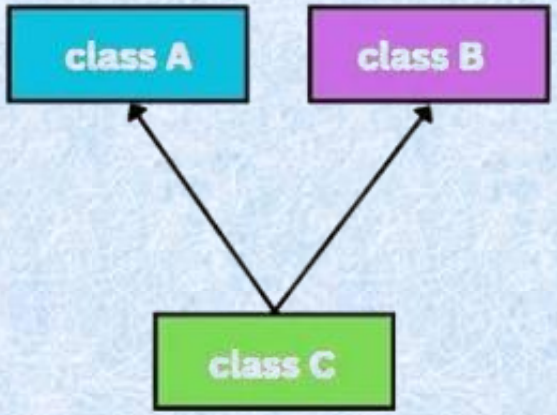


3. Multiple Inheritance

- When a class can be derived from more than one base class this type of inheritance is called multiple inheritances.

Example Code :

```
class Electric:
   def charge_battery(self):
      print("Battery charged")

class Gasoline:
   def refill_gas(self):
      print("Gas tank refilled")

class HybridCar(Electric,  Gasoline):
   def drive(self):
      print("Hybrid car is driving")

my_hybrid_car = HybridCar()
my_hybrid_car.charge_battery()
my_hybrid_car.refill_gas()
my_hybrid_car.drive()
```

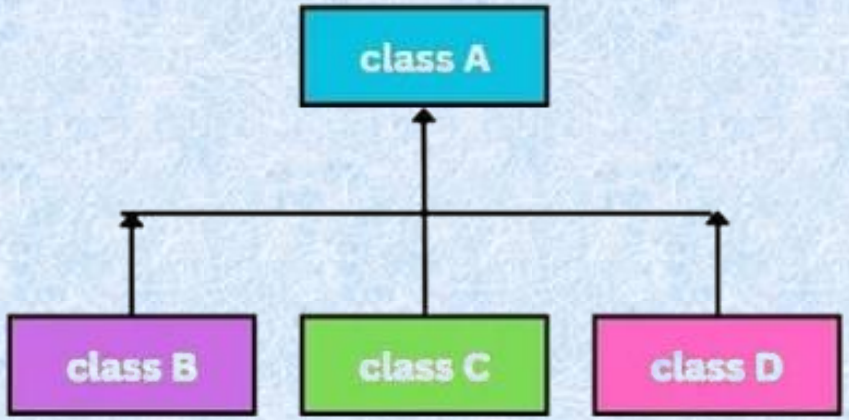


4. Hierarchical Inheritance

- When more than one derived class are created from a single base this type of inheritance is called hierarchical inheritance. .

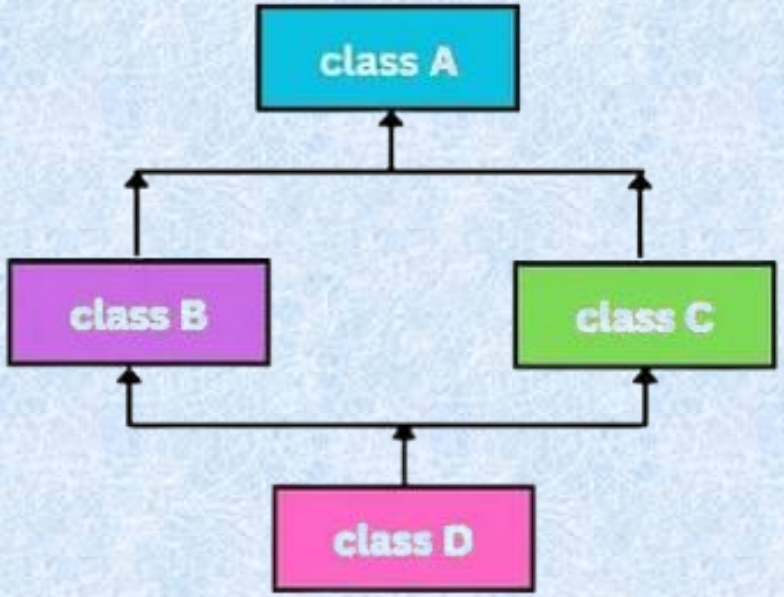
Example Code :

```
class Vehicle:
    def start_engine(self):
        print("Engine started")

class Car(Vehicle):
    def drive(self):
        print("Car is driving")

class Truck(Vehicle):
    def transport_goods(self):
        print("Truck is transporting goods")

my_car = Car()
my_car.start_engine()
my_car.drive()

my_truck = Truck()
my_truck.start_engine()
my_truck.transport_goods()
```



5. Hybrid Inheritance

- Inheritance that consist of multiple types of inheritance is called hybrid inheritance.

Example Code :

```
class Engine:
    def start(self):
        print("Engine started")

class ElectricCar(Engine):
    def charge_battery(self):
        print("Battery charged")

class GasolineCar(Engine):
    def refill_gas(self):
        print("Gas tank refilled")

class HybridCar(ElectricCar, GasolineCar):
    def drive(self):
        print("Hybrid car is driving")

my_hybrid_car = HybridCar()
my_hybrid_car.start()
my_hybrid_car.charge_battery()
my_hybrid_car.refill_gas()
my_hybrid_car.drive()
```