**Types of inheritance in Python**

Types of Inheritance depends upon the number of child and parent classes involved. There are four types of inheritance in Python

**Single Inheritance**

Single inheritance enables a derived class to inherit properties from a single parent class, thus enabling code reusability and the addition of new features to existing code.

# Python program to demonstrate

# single inheritance

# Base class

class Parent:

     def func1(self):

          print("This function is in parent class.")

# Derived class

class Child(Parent):

     def func2(self):

          print("This function is in child class.")

# Driver's code

object = Child()

object.func1()

object.func2()

**Multiple Inheritance**

When a class can be derived from more than one base class this type of inheritance is called multiple inheritance. In multiple inheritance, all the features of the base classes are inherited into the derived class.

# Python program to demonstrate

# multiple inheritance

# Base class1

class Mother:

mothername = ""

def mother(self):

print(self.mothername)

# Base class2

class Father:

fathername = ""

def father(self):

print(self.fathername)

# Derived class

class Son(Mother, Father):

def parents(self):

print("Father :", self.fathername)

print("Mother :", self.mothername)

# Driver's code

s1 = Son()

s1.fathername = "SHAM"

s1.mothername = "SHARVARI"

s1.parents()

**Multilevel Inheritance**

In multilevel inheritance, features of the base class and the derived class are further inherited into the new derived class. This is similar to a relationship representing a child and grandfather.

# Python program to demonstrate

# multilevel inheritance

# Base class

class Grandfather:

def \_\_init\_\_(self, grandfathername):

self.grandfathername = grandfathername

# Intermediate class

class Father(Grandfather):

def \_\_init\_\_(self, fathername, grandfathername):

self.fathername = fathername

# invoking constructor of Grandfather class

Grandfather.\_\_init\_\_(self, grandfathername)

# Derived class

class Son(Father):

def \_\_init\_\_(self,sonname, fathername, grandfathername):

self.sonname = sonname

# invoking constructor of Father class

Father.\_\_init\_\_(self, fathername, grandfathername)

def print\_name(self):

print('Grandfather name :', self.grandfathername)

print("Father name :", self.fathername)

print("Son name :", self.sonname)

# Driver code

s1 = Son('Prince', 'Rampal', 'Lal mani')

print(s1.grandfathername)

s1.print\_name()