*# ------------------------------ Multiple Inheritance --------------------- #  
'''  
Python Multiple Inheritance  
A class can be derived from more than one base class in Python, similar to C++. This is called multiple inheritance.  
In multiple inheritance, the features of all the base classes are inherited into the derived class.  
The syntax for multiple inheritance is similar to single inheritance.  
  
Example  
class Base1:  
 pass  
  
class Base2:  
 pass  
  
class MultiDerived(Base1, Base2):  
 pass  
'''  
  
  
#***class** School:  
 no\_of\_leaves = 8  
  
 **def** \_\_init\_\_(self, name, standard, faculty):  
 self.Name = name  
 self.Standard = standard  
 self.Faculty = faculty  
  
 **def** printdetails(self):  
 **return f'Name is {**self.Name**}. Standard is {**self.Standard**} and Faculty is {**self.Faculty**}. '** @classmethod  
 **def** change\_leaves(cls, newleaves):  
 cls.no\_of\_leaves = newleaves  
  
 @classmethod  
 **def** from\_dash(cls, string):  
 **return** cls(\*string.split(**'-'**))  
  
 @staticmethod  
 **def** simple\_func(string):  
 **return f'{**string**} This is Simple Function, for this simple func we use @staticmethod'  
  
  
class** Player:  
 no\_of\_games\_allow = 4  
  
 **def** \_\_init\_\_(self, name, \*games):  
 self.Name = name  
 self.Games = games  
  
  
 **def** print\_details(self):  
 **return f'Player Name is {**self.Name**}, Games Allow = {**self.no\_of\_games\_allow**}'***# ---------------------------------- Multiple Inheritance -------------- #***class** Cool\_programmer(School, Player): *# In multiple inheritance, orders matter* **pass***# multiple inheritance ma jo order ham derived class ko deta ha to wo order k mutabiq \_\_init\_\_ leta ha*Sherry = School(**'Sheheryar'**, 14, **'AI'**)  
Hamza = School(**'Hamza Rehman'**, 11, **'Learner'**)  
  
  
Furkan = Cool\_programmer(**'Furkan'**, 9, **'Science'**)  
  
print(Furkan.Faculty)