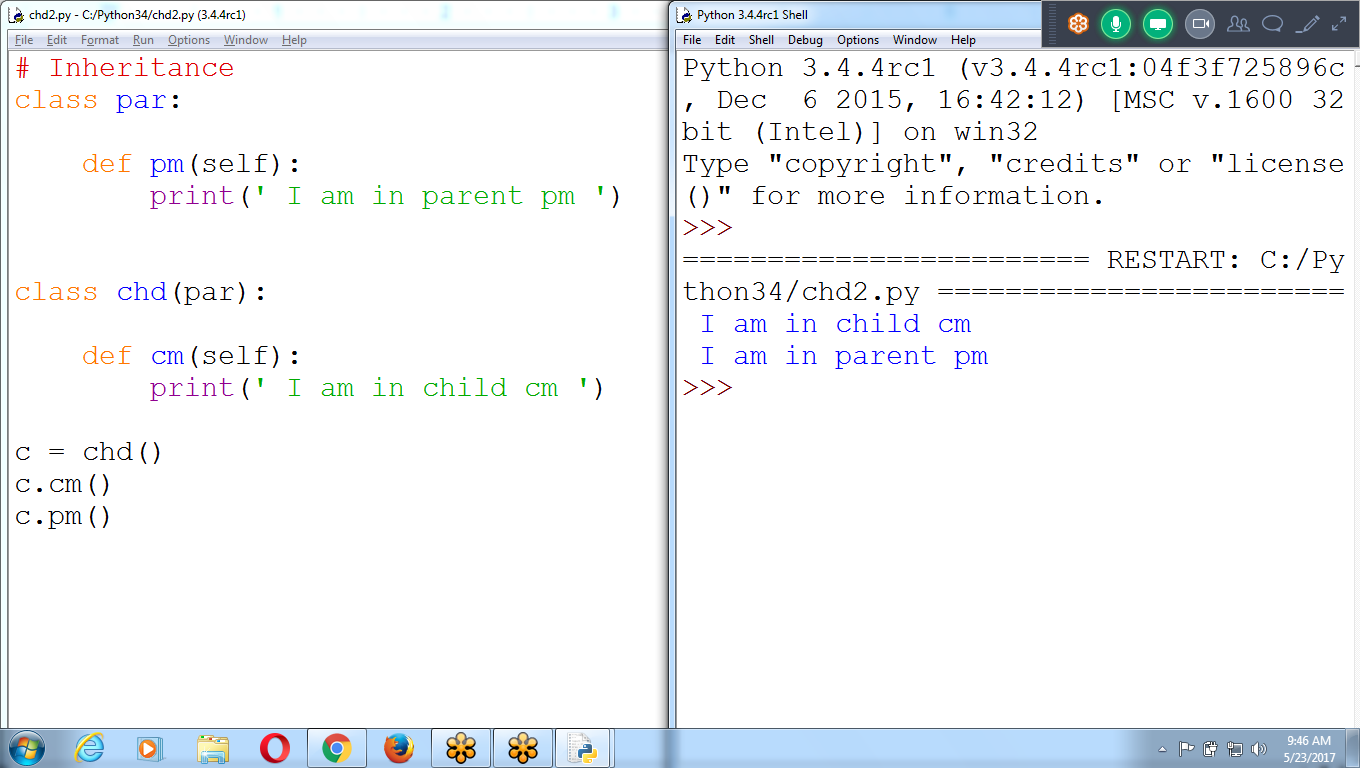
**INheritance : Extending or Modifying Existing class Features**

**Newly Formed classes are derived classes, the classes that we drive from are called base classes.**

**Derived classes override or extend the functionality of base classes**

**Advantage : Code Reusability**



# Inheritance

class par: # parent Class

def pm(self):

print(' I am in parent pm ')

class chd(par): # Child Class

def cm(self):

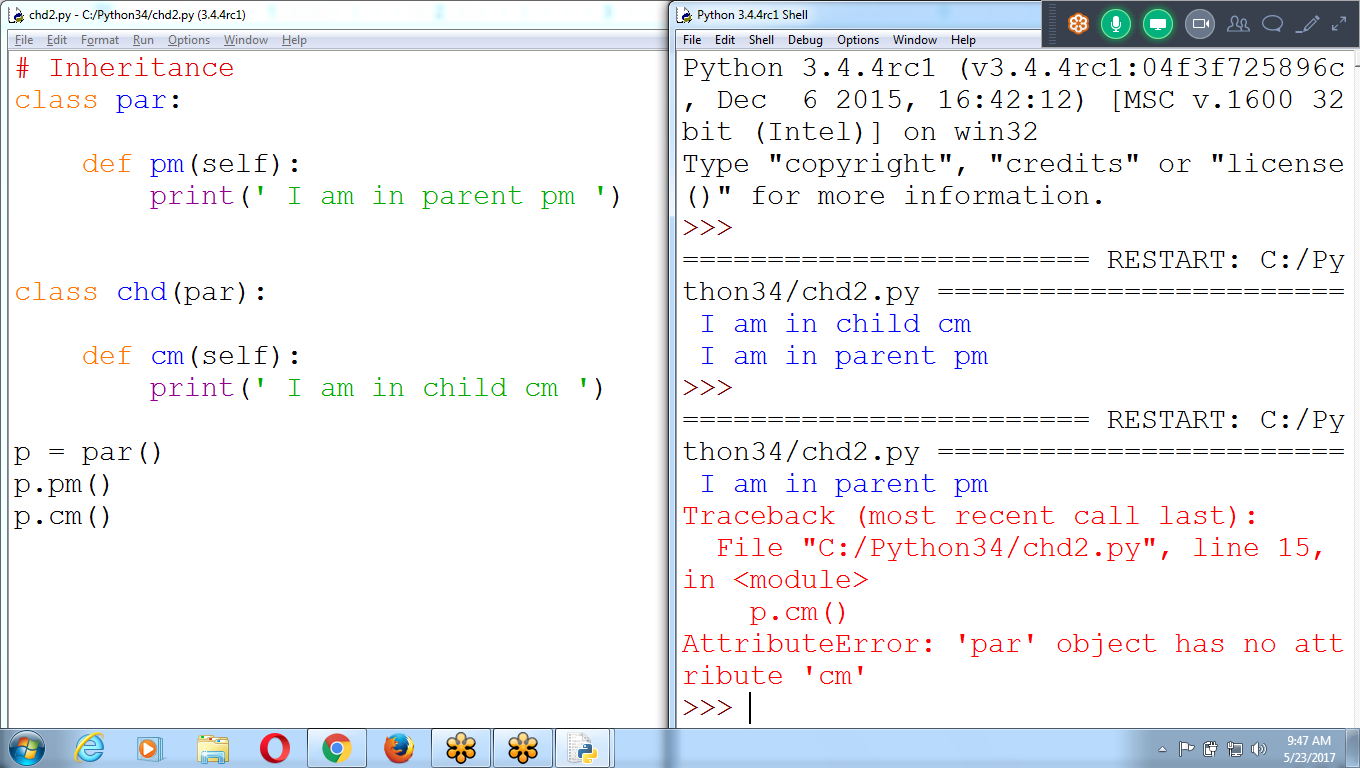
print(' I am in child cm ')

c = chd()

c.cm()

c.pm()

**Parent Can NOT Call CHILD methods**



# Inheritance

class par:

def pm(self):

print(' I am in parent pm ')

class chd(par):

def cm(self):

print(' I am in child cm ')

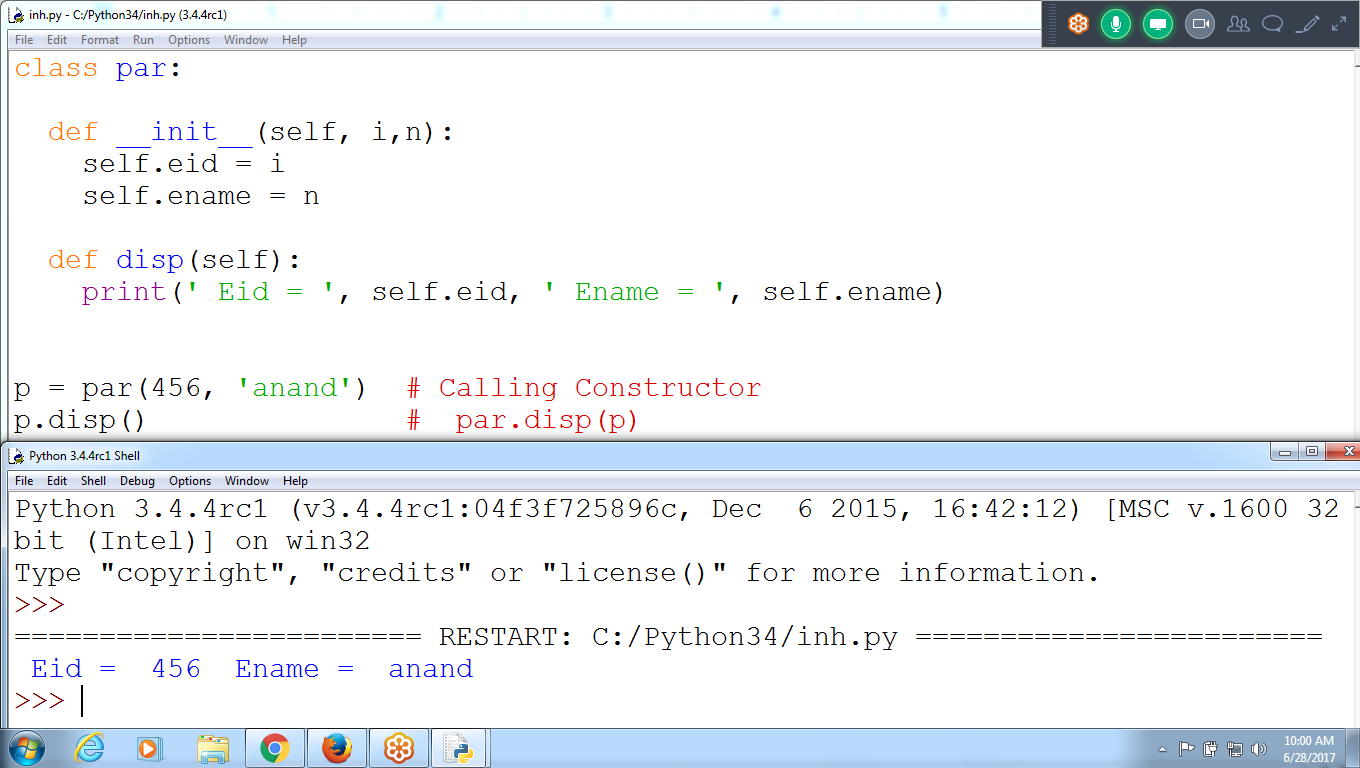
p = par()

p.pm()

**p.cm()**

Class with Constructor

**Parent class contains Constructor and disp()**



class par:

def \_\_init\_\_(self, i,n):

self.eid = i

self.ename = n

def disp(self):

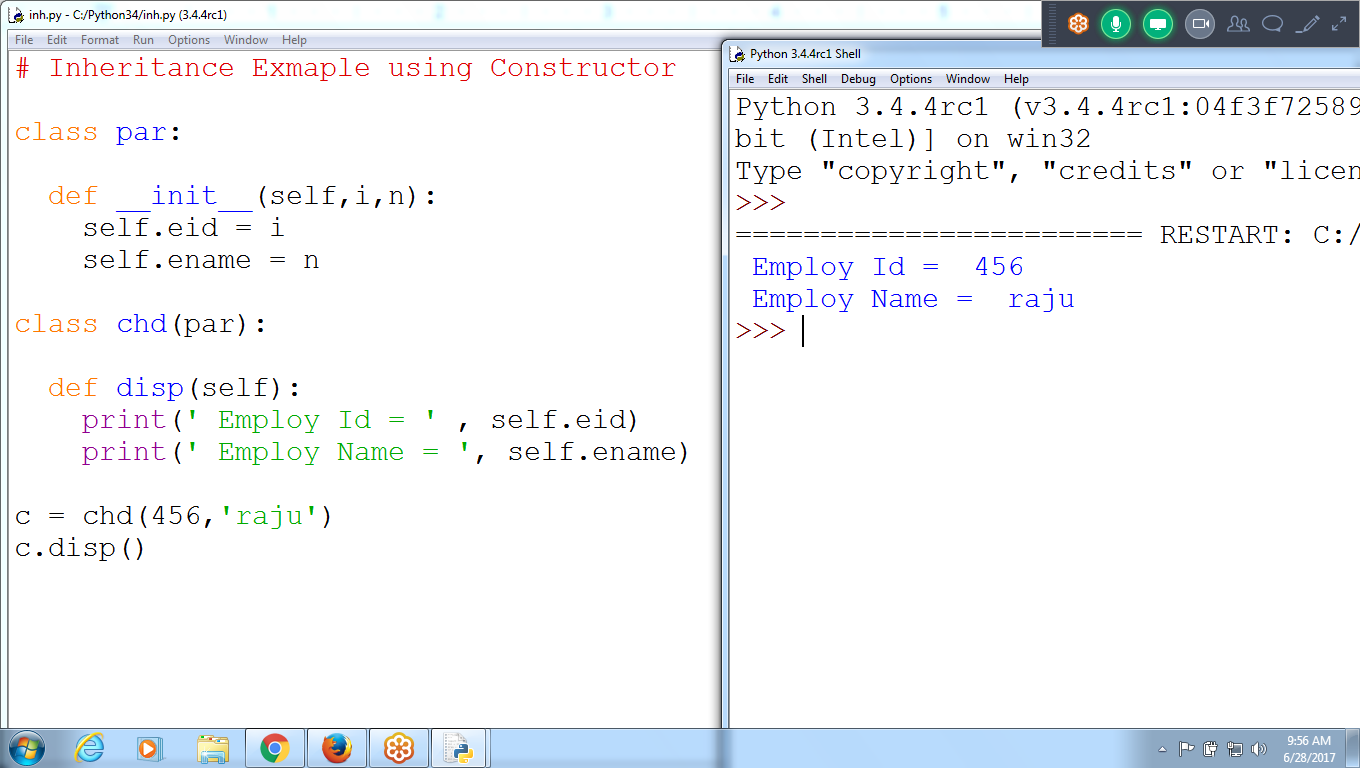
print(' Eid = ', self.eid, ' Ename = ', self.ename)

p = par(456, 'anand') # Calling Constructor

p.disp() # par.disp(p)

**Parent Contains Constructor**

Child contains parent Constructor and disp()

****

**# Inheritance Example using Constructor**

**class par:**

**def \_\_init\_\_(self,i,n):**

**self.eid = i**

**self.ename = n**

**class chd(par):**

**def disp(self):**

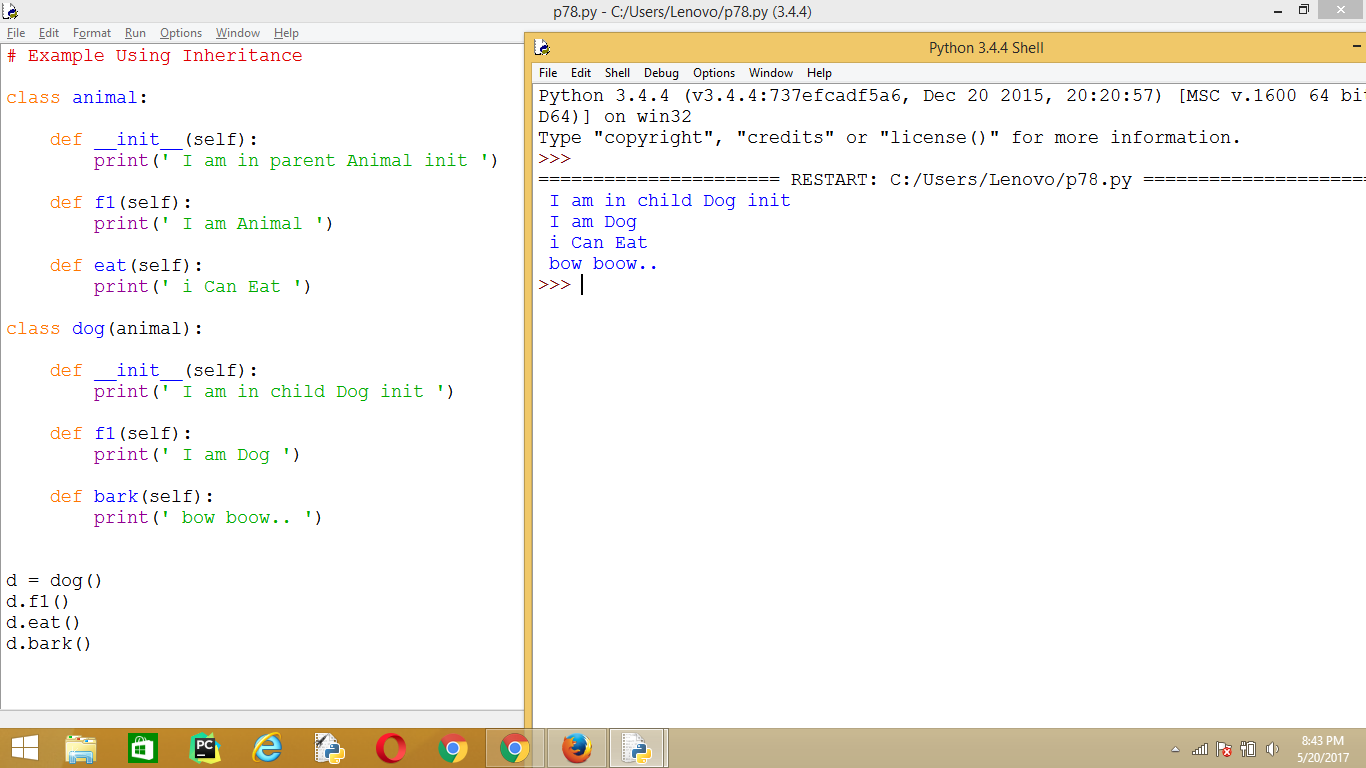
**print(' Employ Id = ' , self.eid)**

**print(' Employ Name = ', self.ename)**

**c = chd(456,'raju')**

**c.disp()**

Example Using Inheritance



# Example Using Inheritance

class animal:

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

print(' I am in parent Animal init ')

def f1(self):

print(' I am Animal ')

def eat(self):

print(' i Can Eat ')

class dog(animal):

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

print(' I am in child Dog init ')

def f1(self):

print(' I am Dog ')

def bark(self):

print(' bow boow.. ')

d = dog()

d.f1()

d.eat()

d.bark()

Inheritance TYPES

**Single Level Inheritance**

C1

C2

**Multilevel Inheritance**

**C1**

**C2**

**C3**

**C4**

**Hierarchy Inheritance (Classification, Tree Like)**

**C1**

**C2 c3**

**C4 c5 c6 c7**

**C1 is parent**

**C2 and c3 are child for parent c1**

**MULTIPLE INheritance : class having more than one parent**

**C1**

**C2 c3 c5**

**C4**

**C2,c3 and c5 are child for parent c1**

**C4 is child for c2,c3 and c5 parents**

**C1(m1:100 Lines Code)**

**C2(m2:200 Lines) C3 (m2:300 Lines)**

**C4(m1:100, m2:200, m2:300)**

**C4 having two parents c2 and c3**

**Obj = c4()**

**obj.m2() ???????**

**# Conflicts which m2() either c2 or c3 to call**

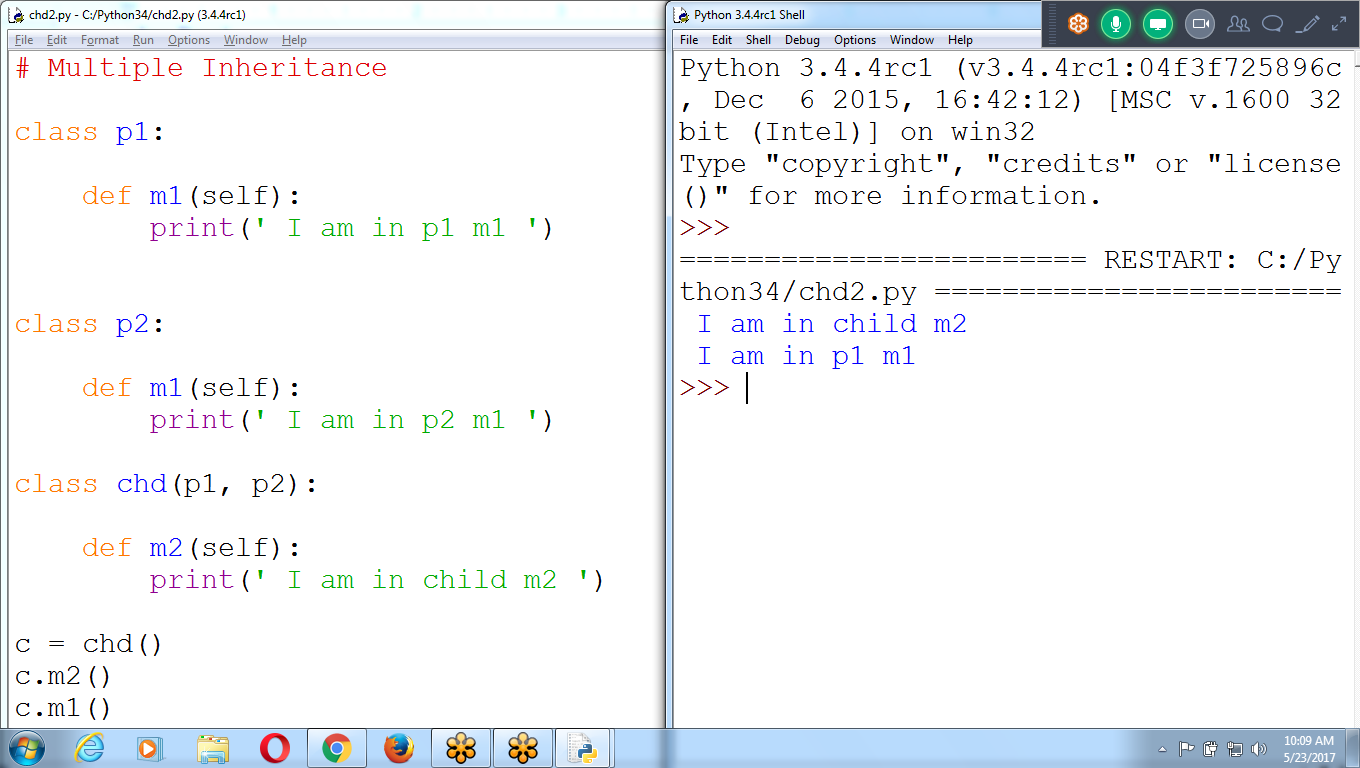
**# C++, Java are NOT allowing Multiple Inheritance**

**Multiple Inheritance allowed**

C++, a class can be derived from more than one base classes in Python. This is called multiple inheritance.

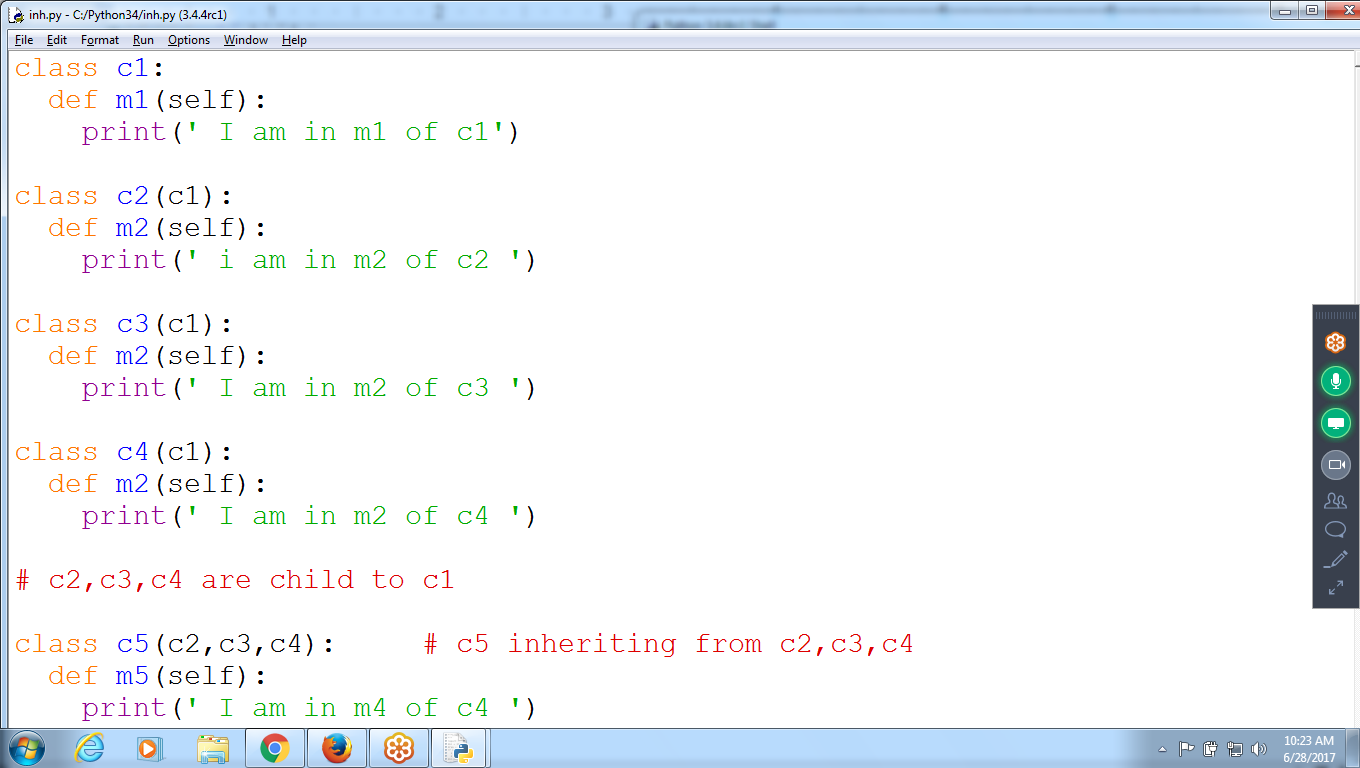
In multiple inheritance, the features of all the base classes are inherited into the derived class. The syntax for multiple inheritance is similar to single inheritance.

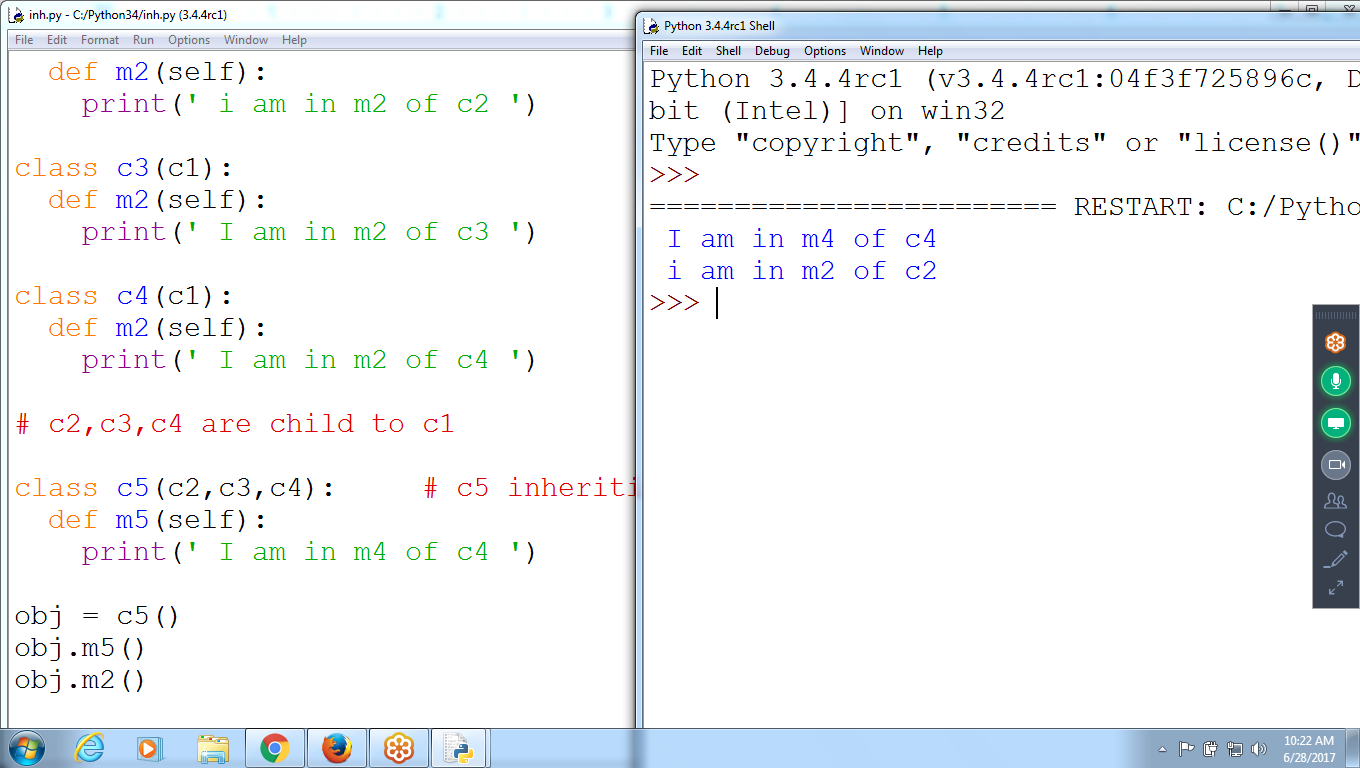
class Base1:  
 pass  
  
class Base2:  
 pass  
  
class MultiDerived(Base1, Base2):  
 pass



Calling first Class p1 : p1 of m1

Multiple Inheritance





Calls First class m2()

**obj.m2() # calling c2 class m2()**

class c1:

def m1(self):

print(' I am in m1 of c1')

class c2(c1):

def m2(self):

print(' i am in m2 of c2 ')

class c3(c1):

def m2(self):

print(' I am in m2 of c3 ')

class c4(c1):

def m2(self):

print(' I am in m2 of c4 ')

# c2,c3,c4 are child to c1

**class c5(c2,c3,c4): # c5 inheriting from c2,c3,c4**

def m5(self):

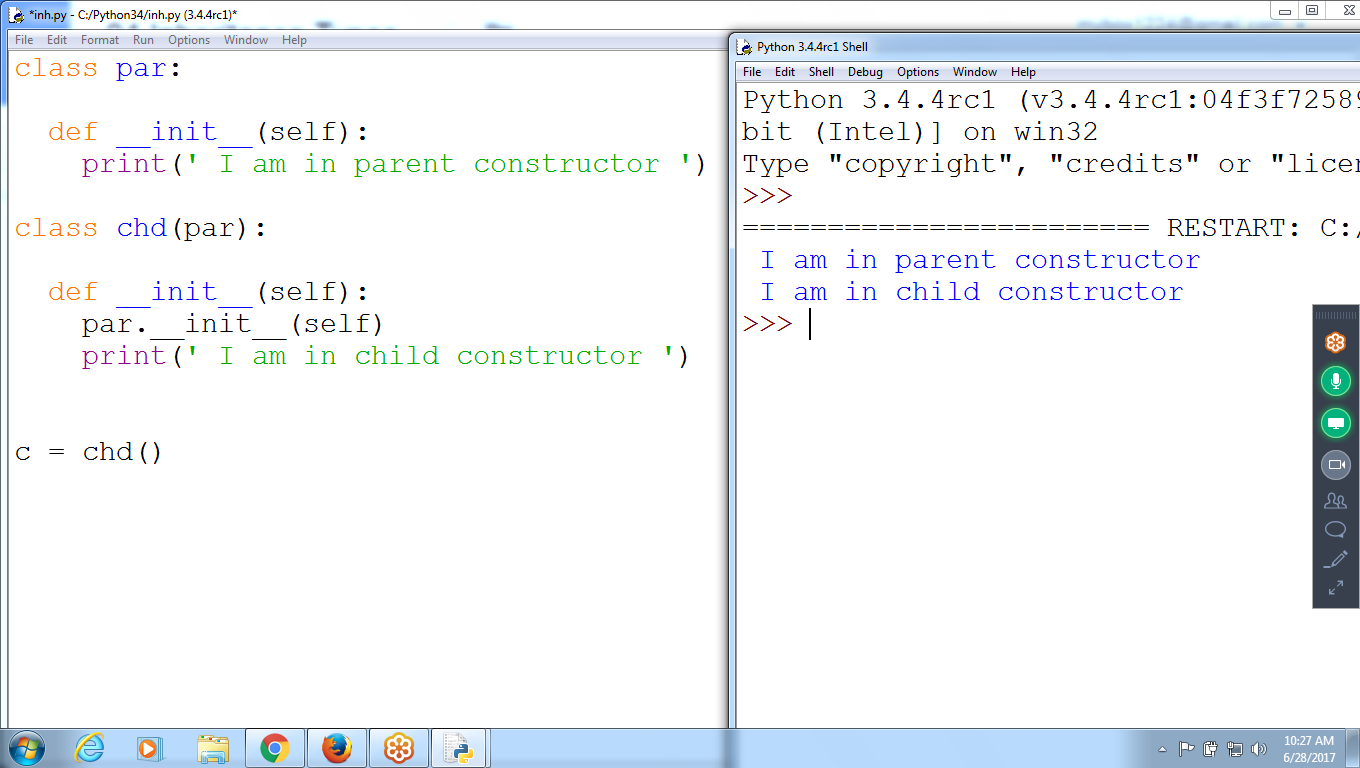
print(' I am in m4 of c4 ')

obj = c5()

obj.m5()

obj.m2()

Child can Call Parent Constructor



class par:

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

print(' I am in parent constructor ')

class chd(par):

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

**par.\_\_init\_\_(self)**

print(' I am in child constructor ')

c = chd()

**New Course Training GIS**

Course Duration : 3 Weeks

**Placements:: WIPRO, Google, APPLE, Conctrics**

**Location : 6 months in Bangalore, later Hyderabad**

**FEE** : FREE Training : 30 Members

REgistratio : 1k

Qualification : Any Graduate

Salary : 15 to 20k ( 1.8 lac to 2.5 lac)

HOurs : 9 Hours

Batch : Aug 1st

Daily : 2 to 3 Hours

Timing: 11.30 to 2 pm

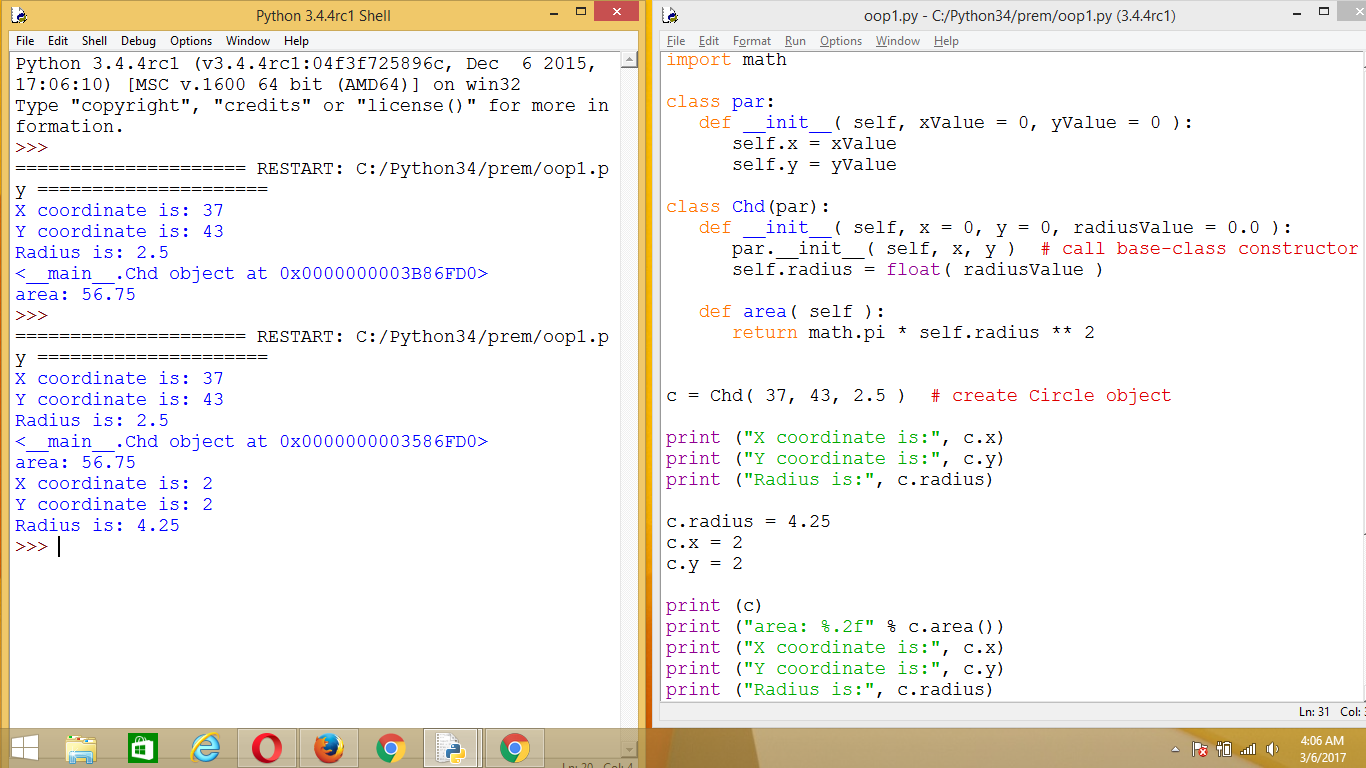
100 Members : marketing

Salary : 15 to 30k

Domain name :: Good Package

Advanced Examples

**Using Parent, child, \_\_init\_ constructor from base class**

****