*#program- self*class check:  
 def \_\_init\_\_(self):  
 print("Address of self = ", id(self))  
  
  
obj = check()  
print("Address of class object = ", id(obj))  
  
*# #PROGRAM*class student:  
 def check\_pass\_fail(self):  
 if self.marks >=40:  
 return True  
 else:  
 return False  
  
 def \_\_init\_\_(self,name,marks):  
 self.name = name  
 self.marks=marks  
  
student1 = student("harry" , 85)  
student2 = student("janet" , 30)  
did\_pass=student1.check\_pass\_fail()  
print(did\_pass)  
  
did\_pass=student2.check\_pass\_fail()  
print(did\_pass)  
  
*#program*class student:  
 def \_\_init\_\_(self,name,rollno,dob,city):  
 self.name = name  
 self.dob = dob  
 self.rollno = rollno  
 self.city = city  
  
 def address(self):  
 addr = f"Name: {self.name}\nDOB : {self.dob}\nRollno : {self.rollno}\nCity: {self.city} "  
 return addr  
  
stu1= student("Anandh",100,1998,"chennai")  
stu2 = student("Ram",200,1999,"Thanjavur")  
print(stu1.address())  
print(stu2.address())  
  
*#program -Array*import array  
balance=array.array('i', [300,200,100])  
print(balance[1])  
  
*#program*import array as myarray  
abc = myarray.array('d', [2.5,4.5,6.7])  
print("Array first element is :" , abc[0])  
print("Array last element is :", abc[-1])  
  
*#program - Array slicing*import array as myarray  
abc = myarray.array('q',[3,9,6,5,20,13,19,22,30,25])  
print(abc[1:4])  
print(abc[7:10])  
  
*#Program - class*class person:  
 def \_\_init\_\_(self,name,age):  
 self.name=name  
 self.age=age  
  
 def printfun(self):  
 print("My name is " ,self.name, ",age is ", self.age)  
  
p1= person ("John",37)  
p1.printfun()  
  
*#Program-delete*class person:  
 def \_\_init\_\_( self,name,age):  
 self.name=name  
 self.age=age  
 def printfun(self):  
 print("My name is ",self.name,",age is ",self.age)  
  
p2=person("Mary",35)  
p2.printfun()  
*#del p2.age*print(p2.age)  
p2.name ="Ram"  
p2.printfun()  
  
*#program - Create a class named Person, with firstname and lastname properties, and a printname method:*class person:  
 def \_\_init\_\_(self,fname,lname):  
 self.firstname=fname  
 self.lastname=lname  
 def printname(self):  
 print(self.firstname,self.lastname)  
  
p2=person("Amenda","johnson")  
p2.printname()  
  
*#Program- create base and derived class*class person: *#baseclass* def \_\_init\_\_(self,fname,lname):  
 self.firstname=fname  
 self.lastname=lname  
 def printname(self):  
 print(self.firstname,self.lastname)  
  
class student(person): *#Derived class* pass  
  
p3=student("john","Robinson")  
p3.printname()  
  
*#Program*class person: *#baseclass* def \_\_init\_\_(self,fname,lname):  
 self.firstname=fname  
 self.lastname=lname  
 def printname(self):  
 print(self.firstname,self.lastname)  
  
class student(person): *#Derived class* def \_\_init\_\_(self,fname,lname):  
 person.\_\_init\_\_(self,fname,lname)  
  
x=student("Mike", "Olsen")  
x.printname()  
  
*#Program - using super function and add properties.*class person:  
 def \_\_init\_\_(self,fname,lname):  
 self.firstname=fname  
 self.lastname=lname  
 def printname(self):  
 print(self.firstname,self.lastname)  
  
class student(person):  
 def \_\_init\_\_(self,fname,lname):  
 super().\_\_init\_\_(fname,lname)  
 self.graduationyear = 2019  
  
x=student("mike","Olsen")  
print(x.graduationyear)  
  
*#program-using constructor*class person:  
 def \_\_init\_\_(self,name):  
 print("Inside constructor")  
 self.name=name  
 print("All variables are initialized")  
 def show(self):  
 print("My name is ",self.name)  
  
x=person("Emili")  
x.show()  
  
*#program-constructor*class Employee:  
 def show(self):  
 print("Inside show method")  
  
d=Employee()  
d.show()  
  
*#Program-Non-Parametrized Constructor*class Employee:  
 def \_\_init\_\_(self):  
 self.name="guna"  
 self.address="Krishna nagar"  
 def display(self):  
 print("My name is ",self.name,",I am reside at ",self.address)  
  
emp=Employee()  
  
emp.display()  
  
*#Program-Parameterized constructor*class industry:  
 def \_\_init\_\_(self,name,age,address):  
 self.name=name  
 self.age=age  
 self.address=address  
  
 def showmethod(self):  
 print("Name- ",self.name,"Age- ", self.age,"Address- ",self.address)  
  
i = industry("Shyam ",34,"Mangalapuram")  
i.showmethod()  
i1 = industry("Siva", 23," Natarajapuram")  
i1.showmethod()  
  
*#Program*class student:  
 def \_\_init\_\_(self,name,age=12, std=12):  
 self.name=name  
 self.age=age  
 self.std=std  
  
 def disp(self):  
 print(f"Name:{self.name}\nAge: {self.age}\nStandard: {self.std}")  
  
s1=student("Ram")  
s1.disp()  
  
s2=student("Anjana" , 13,8)  
s2.disp()

Output:

Address of self = 2616864684016

Address of class object = 2616864684016

True

False

Name: Anandh

DOB : 1998

Rollno : 100

City: chennai

Name: Ram

DOB : 1999

Rollno : 200

City: Thanjavur

200

Array first element is : 2.5

Array last element is : 6.7

array('q', [9, 6, 5])

array('q', [22, 30, 25])

My name is John ,age is 37

My name is Mary ,age is 35

35

My name is Ram ,age is 35

Amenda johnson

john Robinson

Mike Olsen

2019

Inside constructor

All variables are initialized

My name is Emili

Inside show method

My name is guna ,I am reside at Krishna nagar

Name- Shyam Age- 34 Address- Mangalapuram

Name- Siva Age- 23 Address- Natarajapuram

Name:Ram

Age: 12

Standard: 12

Name:Anjana

Age: 13

Standard: 8