OOPS using python

1.

class student:

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

self.name="mohit"

self.age=12

s1=student() #creating object

s2=student()

s3=student()

s1.name = "krishna" # initialization

s1.age= 9

s2.name="test"

s2.age=29

print(s1.name,s1.age) # call using . operator

print(s2.name,s2.age)

print(s3.name,s3.age)

2. https://www.datacamp.com/community/tutorials/finance-python-trading

class employee:

def \_\_init\_\_(self,id,name,salary,dept):

self.id = id

self.name = name

self.salary=salary

self.dept=dept

def show(self):

print(self.name)

def buddy(self,friend):

self.friend = friend

friend.friend = self

pass

obj1 = employee(1,'krishna',70000,'IT')

obj2 = employee(2,'Ankit',70000,'Sales')

obj1.show()

obj2.show()

obj1.buddy(obj2)

# obj2.buddy(obj1)

print("krishna's friend is : ",obj1.friend.name)

print("Ankit's friend is : ",obj2.friend.name)

3.

class employee:

def \_\_init\_\_(self,x,y):

self.x = x

self.y = y

def show(self):

print(self.x+self.y)

obj1 = employee(5,5)

obj1.show()

4.Inheritance

class Calculator:

def \_\_init\_\_(self,x,y):

self.x = x

self.y = y

def show(self):

print(self.x+self.y)

class calc2(Calculator):

def \_\_init\_\_(self,x,y,x1,y1):

self.x1=x1

self.y1=y1

Calculator.\_\_init\_\_(self,x,y)

def show2(self):

print(self.x,self.y,self.x1,self.y1)

obj1 = calc2(5,5,5,5)

obj1.show()

obj1.show2()

Res:

10

5 5 5 5

4.2

class Calculator:

def \_\_init\_\_(self,x,y):

self.x = x

self.y = y

def show(self):

print(self.x+self.y)

class calc2(Calculator):

def \_\_init\_\_(self,x,y,x1,y1):

self.x1=x1

self.y1=y1

Calculator.\_\_init\_\_(self,x,y)

def show2(self):

print(self.x+self.y+self.x1+self.y1)

obj1 = calc2(5,5,5,5)

obj1.show()

obj1.show2()

Res:

10

20

4.3

class Calculator:

def \_\_init\_\_(self,x,y):

self.x = x

self.y = y

def show(self):

print(self.x+self.y)

class calc2(Calculator):

def \_\_init\_\_(self,x,y,x1,y1):

self.x1=x1

self.y1=y1

Calculator.\_\_init\_\_(self,x,y)

def show2(self):

print(self.x\*self.y)

obj1 = calc2(5,5,5,5)

obj1.show()

obj1.show2()

RES:

10

25