

ASSIGNMENT-9

# PYTHON

# **NAME : MADA SRAGVIN KUMAR**

# **MIS NO : 112315097**

# **GROUP : 3**

**YEAR : 2**

**SECTION : A**

# **1:**

import math

class Circle:

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

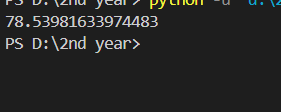
self.r=r

def area(self):

return (math.pi)\*((self.r)\*\*2)

c=Circle(5)

print(c.area())



# **2:**

import math

class Circle:

def \_\_init\_\_(self,l,b):

self.l=l

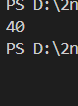
self.b=b

def area(self):

return self.l\*self.b

c=Circle(5,8)

print(c.area())



# **3:**

class Myclass:

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

self.n=n

def iterator(self):

for i in range(self.n):

if i%7==0:

yield i

c=Myclass(64)

for i in c.iterator():

print(i,end=" ")



# **4:**

class Shape:

def area(self):

return 0

class Square(Shape):

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

self.l=l

def area(self):

return self.l\*\*2

sq=Square(5)

print(sq.area())



# **5:**

class My:

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

self.a=a

def getString(self,k):

self.a=k

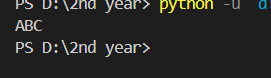
def printString(self):

print(self.a.upper())

m=My("a")

m.getString("abc")

m.printString()



# **6:**

class Person():

def getgender(self):

pass

class Male(Person):

def getgender(self):

print("Male")

class Female(Person):

def getgender(self):

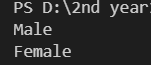
print("Female")

m=Male()

f=Female()

m.getgender()

f.getgender()



# **7:**

class Package:

def \_\_init\_\_(self,package\_id,destination,weight,status):

self.package\_id=package\_id

self.destination=destination

self.weight=weight

self.status=status

def deliver(self):

self.status="delivered"

class Vehicle:

def \_\_init\_\_(self,vehicle\_id,capacity,current\_packages):

self.vehicle\_id=vehicle\_id

self.capacity=capacity

self.current\_packages=current\_packages

def load\_package(self,pack\_obj):

total\_weight=sum([i.weight for i in self.current\_packages])

if total\_weight+pack\_obj.weight<=self.capacity:

self.current\_packages.append(pack\_obj)

else:

print("Vehicle is full")

def deliver\_packages(self):

for i in self.current\_packages:

i.deliver()

class Truck(Vehicle):

def deliver\_packages(self):

print("Vehicle Type:Truck")

deli\_count=0

for i in self.current\_packages:

if i.status=="delivered":

deli\_count=deli\_count+1

print(f"No. of packages delivered:{deli\_count}")

class Drone(Vehicle):

def load\_package(self):

for i in self.current\_packages:

if i.weight>5:

raise ValueError("This package is more than 5kg")

def deliver\_packages(self):

print("Vehicle Type:Drone")

for i in self.current\_packages:

if i.status=="delivered":

deli\_count=deli\_count+1

print(f"No. of packages delivered:{deli\_count}")

class DeliverySystem:

def assign\_vehicle(self,veh\_obj,list\_pack):

self.veh\_obj=veh\_obj

self.veh\_obj.current\_packages=list\_pack

def dispatch(self,veh\_obj):

self.veh\_obj=veh\_obj

for i in self.veh\_obj.current\_packages:

yield i.deliver()

def main():

pack\_list1=[Package("P1","Mumbai",2,"pending"),Package("P2","Chennai",3,"intransit"),Package("P3","Pune",4,"intransit")]

pack\_list2=[Package("P4","Mumbai",2,"pending"),Package("P5","Chennai",3,"pending"),Package("P6","Pune",4,"pending")]

tru\_obj=Truck("V1-Truck",1000,pack\_list1)

dron\_obj=Drone("V2-Drone",10,pack\_list2)

deli\_obj=DeliverySystem()

for i in deli\_obj.dispatch(tru\_obj):

pass

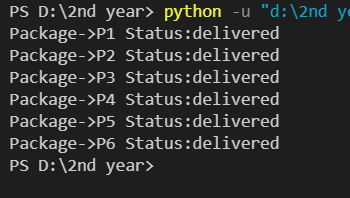
for i in deli\_obj.dispatch(dron\_obj):

pass

for i in pack\_list1+pack\_list2:

print(f"Package->{i.package\_id} Status:{i.status}")

main()



# 

# **8:**

from datetime import \*

from random import \*

count=0

def sensor\_data():

while True:

yield (f"{datetime.now()}",randint(-10,50))

def filter\_by\_threshold():

for i in sensor\_data():

if i[1]>=0 and i[1]<=40:

yield i

mylist=[]

def smooth\_temperature():

for i in filter\_by\_threshold():

mylist.append(i[1])

if(len(mylist)<3):

smoothed\_temperature=sum(mylist)/len(mylist)

else:

smoothed\_temperature=(mylist[-1]+mylist[-2]+mylist[-3])/3

yield (f"{datetime.now()}",smoothed\_temperature)

def convert\_to\_farenheit():

for i in smooth\_temperature():

temperature\_farenheit=(1.4\*i[1])+32

yield (f"{datetime.now()}",temperature\_farenheit)

def main():

for i in convert\_to\_farenheit():

print(i)

global count

count=count+1

if(count==20):

break

main()

