'''

this is demonstration of has attribute.

'''

class Employee:

'''this is demonstration of class'''

empCount = 0

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

self.name=name

self.salary=salary

Employee.empCount +=1

def displayCount(self):

print("Total Employee %d" % Employee.empCount)

def displayEmployee(self):

print("Name",self.name,"Salary",self.salary)

emp1=Employee("Nikhil",9999)

emp1.displayEmployee()

print("is salary an attribute:",hasattr(emp1,'salary'))

print(getattr(emp1,'salary'))

setattr(emp1,'salary',7000)

print("Modified salary",getattr(emp1,'salary'))

print(hasattr(emp1,'desg'))

setattr(emp1,'desg','manager')

print(hasattr(emp1,'desg'))

print("Added attribute",getattr(emp1,'desg'))

delattr(emp1,'salary')

print("is salary an attribute",hasattr(emp1,'salary'))



class A:

pass

class B:

pass

class Employee(A,B):

'''

this is employee

'''

empCount=0

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

self.name=name

self.salary=salary

Employee.empCount+=1

def displayCount(self):

print("Total Employee %d" % Employee.empCount)

def displayEmployee(self):

print("Name",self.name,"Salary:",self.salary)

if \_\_name\_\_=="\_\_main\_\_":

print("Employee.\_\_doc\_\_:",Employee.\_\_doc\_\_)

print("Employee.\_\_name\_\_:",Employee.\_\_name\_\_)

print("Emplopyee.\_\_module\_\_:",Employee.\_\_module\_\_)

print("Employee.\_\_bases\_\_:",Employee.\_\_bases\_\_)

print("Employee.\_\_dict\_\_:",Employee.\_\_dict\_\_)



class Point:

def \_\_init\_\_(self,x=0,y=0):

self.x=x

self.y=y

def \_\_del\_\_(self):

class\_name=self.\_\_class\_\_.\_\_name\_\_

print(class\_name,"destroyed")

pt1=Point()

pt2=pt1

pt3=pt1

print(id(pt1),id(pt2),id(pt3))

del pt1

del pt2

del pt3



def printinfo(arg1, \*vartuple):

print("Output is:",arg1)

print("Contents of variable length tuple is:")

for var\_temp in vartuple:

print(var\_temp,end=' ')

print()

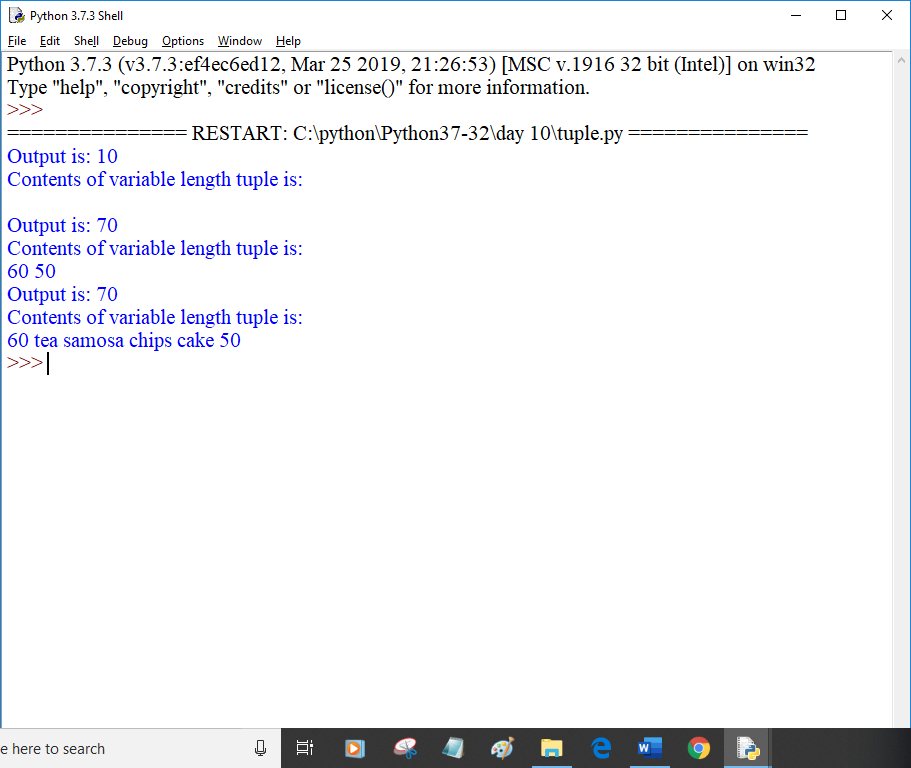
return

if \_\_name\_\_=='\_\_main\_\_':

printinfo(10)

printinfo(70,60,50)

printinfo(70,60,'tea','samosa','chips','cake',50)



def test\_fun(stream,course,fee):

print("arg1:",stream)

print("arg2:",course)

print("arg3:",fee\*2)

return

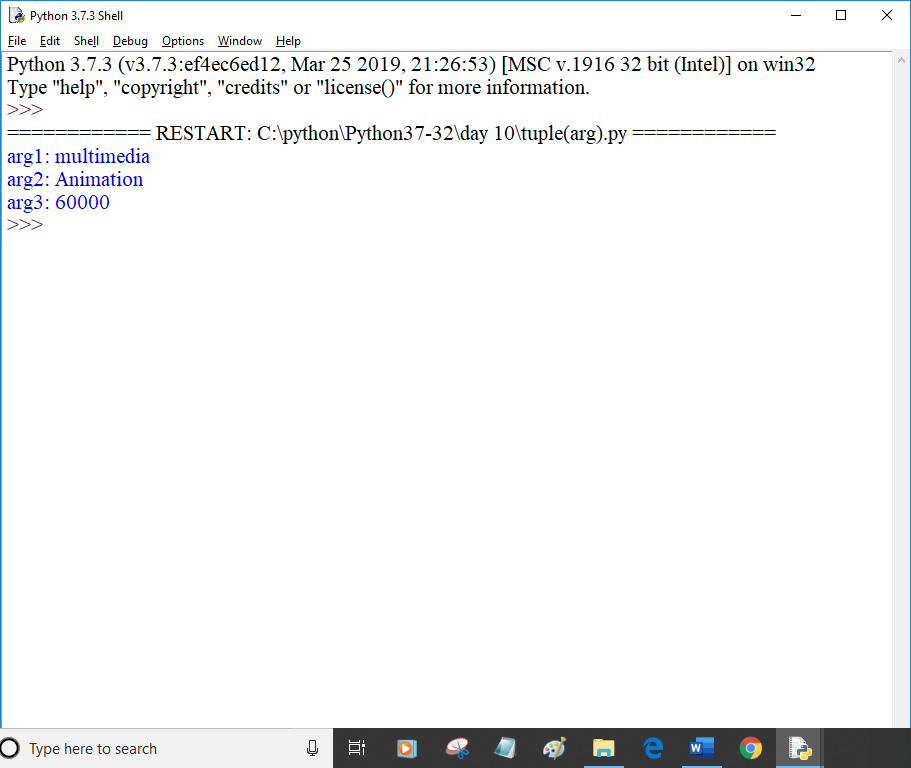
##

##tup1=('Programming','python',20000)

##test\_fun(\*tup1)

dict1={"fee":30000,"course":"Animation","stream":"multimedia"}

test\_fun(\*\*dict1)



def printEmpSkill(name,\*skillset):

print("{0} is skilled in {1} language".format(name,skillset))

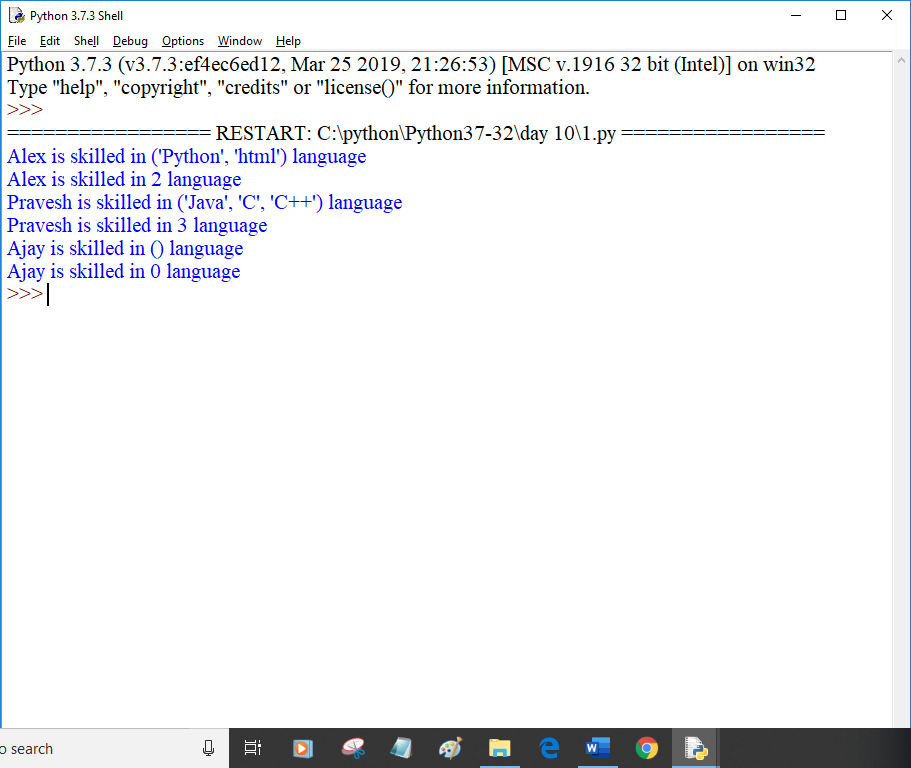
print("{0} is skilled in {1} language".format(name,len(skillset)))

return None

printEmpSkill('Alex','Python','html')

printEmpSkill('Pravesh','Java','C','C++')

printEmpSkill('Ajay')



total=100

def add(arg1,arg2):

global total

total=arg1+arg2

print("Inside the function local total:",total)

return total

a=add(10,20)

print("Outside the function global total:",total)

