**Python LC’s and CC’s**

**Variables and Operators/ CC**

**1.Spherical Cargos**

a=int(input())

d= int(input())

n= int(input())

vc=a\*a\*a

r=float(d)/2

vs=(4./3)\*3.14\*(r\*r\*r)

tv=vs\*n

rs=vc-tv

print(format(rs,'.2f'))

**2.Celsius to Fahrenheit Converter**

c=int(input("Enter the Celsius : "+'\n'))

f=(1.8\*c)+32

print('%0.1f degree Celsius is equal to %0.1f degree Fahrenheit' %(c,f))

**Decision Making / CC**

**1.** **Discount Tuesdays**

a=int(input())

b=int(input())

c=int(input())

d=int(input())

e=int(input())

f=int(input())

g=int(input())

h=int(input())

wt=(a\*g)+(b\*h)

wb=(c\*g)+(d\*h)

wf=(e\*g)+(f\*h)

if (wt<wb and wt<wf):

print ("Train Transportation")

elif (wb<wt and wb<wf):

print ("Bus Transportation")

elif (wf<wt and wf<wb):

print ("Flight Transportation")

2. **Offer to Valuable Customers**

i=input("Enter the type of relationship with customer\n")

s=input("Enter the slot of the customer\n")

if(i=='New') and (s=='S1'):

print "Amount to be paid =",int(200-(200\*10)/100)

elif(i=='New') and (s=='S3'):

print "Amount to be paid =",int(200-(200\*10)/100)

elif(i=='New') and (s=='S2'):

print "Amount to be paid =",int(800-(800\*10)/100)

elif(i=='New') and (s=='S4'):

print "Amount to be paid =",int(800-(800\*10)/100)

elif(i=='Gold') and (s=='S1'):

print "Amount to be paid =",int(200-(200\*25)/100)

elif(i=='Gold') and (s=='S3'):

print "Amount to be paid =",int(200-(200\*25)/100)

elif(i=='Gold') and (s=='S2'):

print "Amount to be paid =",int(800-(800\*25)/100)

elif(i=='Gold') and (s=='S4'):

print "Amount to be paid =",int(800-(800\*25)/100)

elif(i=='Platinum') and (s=='S1'):

print "Amount to be paid =",int(200-(200\*40)/100)

elif(i=='Platinum') and (s=='S3'):

print "Amount to be paid =",int(200-(200\*40)/100)

elif(i=='Platinum') and (s=='S2'):

print "Amount to be paid =",int(800-(800\*40)/100)

elif(i=='Platinum') and (s=='S4'):

print "Amount to be paid =",int(800-(800\*40)/100)

else:

print (“Invalid Input”)

**Creating and Executing Loops / CC**

1.Cargo Id and Container Id

x=int(input())

for i in range(1,x+1):

if x%i==0:

print(i, end=’ ‘),

2.The Container Contains

k=1

while k>0:

x=int(input("Enter the number of items in the box"+" "+str(k)+'\n'))

if x%8==0:

k=k+1

else:

break

print("Number of boxes stored in container is"+" "+str(k-1))

**Working with Lists / CC**

**1.All Products**

data1=input("Enter the Numbers in List-1\n")

list1=data1.split(" ")

data2=input("Enter the Numbers in List-2\n")

list2=data2.split(" ")

while '' in list1:

list1.remove('')

while '' in list2:

list2.remove('')

result=list()

for x in range (len(list1)):

num=int(list1[x])

for y in range (len(list2)):

mu=num\*int(list2[y])

if mu%2!=0:

result.append(mu)

if len(result)>0:

print(" ".join(str(z) for z in result))

else:

print ("No such Elements in the list")

**2.Rotate the List**

import collections

data=input("Enter the Numbers in List\n")

myList=data.split(" ")

print("Before Rotating : ")

print(" ".join(str(y) for y in myList)+" ")

d=collections.deque(myList)

d.rotate(-1)

print ("After Rotating : ")

print (" ".join(str(y) for y in d))

**Working with Strings / CC**

1.**The Chatbot**

i=input()

if i.find("medicine")>=0 or i.find("tablet")>=0 or i.find("drugs")>=0:

print "C-Cargo"

if i.find("chocolate")>=0 or i.find("meat")>=0 or i.find("fruit")>=0:

print "F-Cargo"

if i.find("electronics")>=0 or i.find("mobile")>=0 or i.find("pc")>=0:

print "E-Cargo"

**Working with Tuples / CC**

1.Duplicate Cargo

data=input()

nu=data.split(",")

count=0

myList=[]

while count<len(nu):

newInt=int(nu[count])

count=count+1

if newInt not in myList:

myList.append(newInt)

print(tuple(myList))

**Declaring Functions and Passing Values / CC**

1.Overlapping Cities

s1=input()

s2=input()

def ports(s):

s=s.split(',')

len1=len(s)

str=[]

for i in range(0,len1):

var=s[i]

var1=var.strip()

str.append(var1)

return str

def port(s1,s2):

count=0

s1=s1.split(',')g

s2=s2.split(',')

a=[]

b=[]

for c in s1:

for d in s2:

if(c.lower()==d.lower()):

count=count+1

break

if(count>0):

return "Overlapping"

else:

return "Non Overlapping"

str1=ports(s1)

str2=ports(s2)

print(str1)

print(str2)

str3=port(s1,s2)

print(str3)

**Working with Dictionary / CC**

1.Translating a Commodity Name to German

in1=input()

in2=input()

myList=in1.split(",")

dict={}

for x in range(len(myList)):

sp=myList[x].split()

key=sp[0]

value=sp[1]

dict[key]=value

def convert(param):

val=''

p=param.split()

for y in range(len(p)):

if p[y] not in dict.keys():

print("The sentence cannot be translated")

exit()

else:

val=val+' '+dict[p[y]]

if val!='':

val=val.lstrip()

print(val)

convert(in2)

**File Handling / CC**

1.Smallest Cargo Name

import string

def length():

with open("text.txt","r") as f:

content=f.readlines()

for item in content:

if len(item)==len(min(content,key=len)):

item=item.strip()

print(item)

length()

2.Eligible Candidate

import xml.etree.ElementTree as ET

tree = ET.parse('candidate.xml')

root=tree.getroot()

for country in root.findall('candidate'):

if int((country.find('age').text))>=25:

print(country.find('candidateName').text+" "+":"+" "+(country.find('age').text))