#1

def generate\_n\_chars(n,c):

print(c\*n)

generate\_n\_chars(int(input("enter number: ")),(input("enter char: ")))

#2

def maxInList(lt):

max = int(lt[1])

print(type(max))

for x in lt:

y = int(x)

if y>max:

max = y

return max

a = input("enter list of values separated by comma: ").split(",")

print(maxInList(a));

#3

length = lambda str : (str, len(str))

print(list(map(length,["Hirak", "Jyoti","ATP","Bangalore"])))

#4

def maxInList(lt):

max = int(lt[1])

for x in lt:

y = int(x)

if y>max:

max = y

return max

string = input("enter list of values separated by spaces: ").split(" ")

l = [len(x) for x in string]

print(maxInList(l))

#5

n = int(input("enter length: "))

filterLongWords = lambda x : x if len(x)>n else ""

string = input("enter list of values separated by spaces: ").split(" ")

newList = (list(map(filterLongWords,string)))

for x in newList:

if x == '':

newList.remove(x)

print(newList)

#6

string = input("enter list of values separated by spaces: ").lower().split(" ")

s=""

for x in string:

s = s+x

if s == s[::-1]:

print("palindrome")

else:

print("not palindrome")

#7

string = input("enter list of values separated by spaces: ").lower().split(" ")

s=""

for x in string:

s = s+x

lt = [x for x in s]

if len(set(lt))==26:

print("panagram")

else:

print("not panagram")

#8

def translate(key):

dic = {"merry":"god", "christmas":"jul", "and":"och", "happy":gott", "new":"nytt", "year":"år"}

return dic.get(key)

print(translate(input("enter English word to translate to Swedish : ")))

#9

def char\_freq(str):

s = set(str.split(‘’))

for x in s:

count = 0;

for i in len(str):

if x==str[i]:

count = count+1

print("frequency of x: " count)

str = int(input("enter string: "))

#10

arr=input("enter the numbers seperated by spaces")

add(arr)

add(arr)

def add(innum):

array=()

array = list(map(int, array))

print("result \t",sum(array))

def sub(innum):

array=()

array = list(map(int, array))

print("result\t",array[0]-array[1])

def sort(innum):

array=()

array = list(map(int, array))

array.sort()

print("result",array)

def maximum(innum):

array=()

array = list(map(int, array))

print("max result\t",max(array))

def addarr(a,b):

return list(map((lambda a,b:a+b),a,b))

def subarr(a,b):

return list(map((lambda a,b:a-b),a,b))

def maxarr(a,b):

return list(map((lambda a,b:max(a,b)),a,b))

def minarr(a,b):

return list(map((lambda a,b:min(a,b)),a,b))

#11

######test.py######module

import mathematics.py

arr=input("enter the numbers seperated by spaces")

mathematics.add(arr)

mathematics .add(arr)

######module######mathematics.py

def add(innum):

array=()

array = list(map(int, array))

print("result \t",sum(array))

def sub(innum):

array=()

array = list(map(int, array))

print("result\t",array[0]-array[1])

def sort(innum):

array=()

array = list(map(int, array))

array.sort()

print("result",array)

def maximum(innum):

array=()

array = list(map(int, array))

print("max result\t",max(array))

def addarr(a,b):

return list(map((lambda a,b:a+b),a,b))

def subarr(a,b):

return list(map((lambda a,b:a-b),a,b))

def maxarr(a,b):

return list(map((lambda a,b:max(a,b)),a,b))

def minarr(a,b):

return list(map((lambda a,b:min(a,b)),a,b))

#12

Import datetime

class Date:

def fun(self):

print(datetime.date.today())

class UseDate :

obj = Date()

obj.fun();

#13

file = open('C:\\Users\\hirdutta\\Desktop\\Python day2\\download.jpg','rb')

a = file.read()

newFile = open('newFile.jpg','wb')

newFile.write(a)

file.close()

newFile.close()