Python L1 Assignments:

1. What will be the output of 'seclist' in print commands of below code?

mylist = range(4)

seclist = mylist

print seclist

mylist.append(4)

print seclist

seclist = mylist[:]

print seclist

mylist.append(5)

print seclist

Answer:

[0, 1, 2, 3]

[0, 1, 2, 3, 4]

[0, 1, 2, 3, 4]

[0, 1, 2, 3, 4]

2. What is the output of following code:

def f(n):

for x in range(n):

yield x\*\*3

for x in f(6):

print x

Answer:

0

1

8

27

64

125

3. Write a program to receive a string from keybord and check if the string has two 'e' in the characters.

If yes return True else False.

str = input("Enter a string:::")

count=0

for i in str:

if(i=='e'):

count+=1

if(count==2):

print("True")

else:

print("False")

4. What is the output of following code:

counter = 1

def dolots(count):

global counter

for i in (1, 2, 3):

counter = count + i

print dolots(4)

print counter

Answer:

None

7

1. Write a code to read the data from input file called input.txt and count the number of characters per line, number of words per line and write these into output file called as output.txt

Answer:

with open("input.txt") as f:

line=f.readline()

while line:

line\_count = 1

a=line.split()

count=len(a)

f2=open("output.txt","a")

f2.write("Number of characters in Line{} \t".format(line\_count))

f2.write(str(count))

f2.write("\n")

f2.close()

line=f.readline()

line\_count+=1

f.close()

with open("input.txt") as nf:

line=nf.readline()

while line:

line\_count=1

nf1=open("output.txt","a")

nf1.write("Number of words in Line{} \t".format(line\_count))

nf1.write(str(len(line.replace(" ",""))-1))

nf1.write("\n")

nf1.close()

line=nf.readline()

line\_count+=1

nf.close()

1. Create 3 Lists ( list1,list2,list3) with numbers and perform following operations

a) Create Maxlist by taking 2 maximum elements from each list.

b) Find average value from all the elements of Maxlist.

c) Create a MinlIst by taking 2 minimum elements from each list

d) Find the average value from all the elements of Minlist

def lmax(lst,l):

a=max(lst)

l.append(a)

lst.remove(a)

b=max(lst)

l.append(b)

def lmin(lst,n1):

a=min(lst)

n1.append(a)

lst.remove(a)

b=min(lst)

n1.append(b)

def Average(lst):

return (sum(lst)/len(lst))

list1=[1,3,5,13,88,12]

list2=[11,421,22,53,12,20]

list3=[40,89,90,11]

mx1=[]

mx2=[]

mx3=[]

mn1=[]

mn2=[]

mn3=[]

lmax(list1,mx1)

lmax(list2,mx2)

lmax(list3,mx3)

lmin(list1,mn1)

lmin(list2,mn2)

lmin(list3,mn3)

Maxlist = mx1+mx2+mx3

print("MAXLIST",Maxlist)

Minlist=mn1+mn2+mn3

print("MINLIST",Minlist)

print("AVERAGE MAXLIST",Average(Maxlist))

print("AVERAGE MINLIST",Average(Minlist))

1. Write program to convert prefix/net mask to IP

eg: input:16 output: 255.255.0.0

from netaddr import IPAddress

print(IPAddress('255.255.255.0').netmask\_bits())

1. Create a suitable data construct to read the data from an xml document as shown below:

<bookstore shelf="New Arrivals">

<book category="COOKING">

<title lang="en">Everyday Italian</title>

<author>Giada De Laurentiis</author>

<year>2005</year>

<price>30.00</price>

</book>

<book category="CHILDREN">

<title lang="en">Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book category="WEB">

<title lang="en">Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>

Answer:

from bs4 import BeautifulSoup

with open('SAMPLE.xml', 'r') as f:

data = f.read()

Bs\_data = BeautifulSoup(data, "xml")

b\_bookstore = Bs\_data.find\_all('bookstore')

print(b\_bookstore)

1. Create a suitable object type and check for file size of 0 bytes of the directory contents as shown below

02/15/2016 10:49 PM 962 switchfinal.py

02/15/2016 10:49 PM 943 switchfinal.py.bak

01/27/2016 11:46 AM 15 t.py

03/31/2016 12:39 PM 840 t1.py

01/25/2016 10:34 AM 2,407 tc1.py

02/14/2017 09:13 AM 0 teat.py

03/15/2016 05:52 PM 5 tes.py

import os

path=os.path.abspath(input("Enter path::"))

size = 0

min\_file=""

for folder,subfolders,files in os.walk(path):

for file in files:

size = os.stat(os.path.join(folder, file)).st\_size

if size==0:

min\_file=os.path.join(folder,file)

print("The smallest file is:",min\_file)

10.Create a suitable object type to eliminate the duplicate elements

list=[1,1,1,3,4,2,44,22,21,33,22]

print("Original list::", list)

temp=[]

for i in list:

if i not in temp:

temp.append(i)

print("After removing duplicates:::",temp)