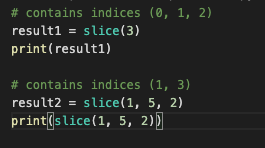
TASK THREE:

DATA STRUCTURES

1. Create a list of the 10 elements of four different types of Data Types like int, string, complex, and float.
2. my\_list = [1,2,4,4.3,’india’,(3,4),’usa’,1j,4j,5.65]
3. Create a list of size 5 and execute the slicing structure



1. Write a program to get the sum and multiply of all the items in a given list.

# Python program to multiply and sum all values in the

def multiplyList(myList) :

# Multiply elements one by one

result = 1

for x in myList:

result = result \* x

return result

# Driver code

list1 = [1, 2, 3]

res=multiplyList(list1)

print('product of list is:',res)

# def function to write sum of all elemnets in given list

list1 = [11, 5, 17, 18, 23]

# creating sum\_list function

def sumOfList(list, size):

if (size == 0):

return 0

else:

return list[size - 1] + sumOfList(list, size - 1)

# Driver code

total = sumOfList(list1, len(list1))

print("Sum of all elements in given list: ", total)

1. Find the largest and smallest number from a given list.
2. ls = []
3. num = int(input('How many numbers: '))
4. for n in range(num):
5. numbers = int(input('Enter number '))
6. ls.append(numbers)
7. print("Maximum element in the list is :", max(ls))
8. print("Minimum element in the list is :", min(ls))

5. Create a new list that contains the specified numbers after removing the even numbers from a predefined list.

#Remove even number in given list

num = [1, 2, 5, 8, 16, 35, 38, 96, 29, 39]

num = [x for x in num if x%2!=0]

print(num)

6. Create a list of first and last 5 elements where the values are square of numbers between 1 and 30 (both included).

def inRange():

ls= list()

for i in range(1,30):

ls.append(i\*\*2)

print(ls[:5])

print(ls[-5:])

inRange()

7. Write a program to replace the last element in a list with another list.

Sample data: [[1,3,5,7,9,10],[2,4,6,8]]

Expected output: [1,3,5,7,9,2,4,6,8]

list1 = [1, 3, 5, 7, 9, 10]

list2 = [2, 4, 6, 8]

list1[-1:] = list2

print(list1)

8. Create a new dictionary by concatenating the following two dictionaries:

a={1:10,2:20}

b={3:30,4:40}

Expected Result: {1:10,2:20,3:30,4:40}

dict1={1:10, 2:20}

dict2={3:30, 4:40}

dict3 = {}

for d in (dict1, dict2): dict3.update(d)

print(dict3)

9. Create a dictionary that contains a numbers in the form(x:x\*x) where x takes all the values between 1 and n

Sample data : n=5

Expected Output: {1:1,2:4,3:9,4:16,5:25}

num=int(input("Input a number "))

d = dict()

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

d[x]=x\*x

print(d)

10. Write a program which accepts a sequence of comma-separated numbers from the console and generate a list and a tuple which contains every number. Suppose the following input is supplied to the program:

34,67,55,33,12,98

The output should be:

[‘34’,’67’,’55’,’33’,’12’,’98’]

(‘34’,’67’,’55’,’33’,’12’,’98’)

l = str(input('enter comma separated values:'))

li = l.split(',')

tup = tuple(li)

li = list(li)

print(li)

print(tup)

Output:

chaithras-MacBook-Air:~ chaithraanjikumar$ python /Users/chaithraanjikumar/Desktop/test.py

enter comma separated values:34,67,55,33,12,98

['(34', ' 67', ' 55', ' 33', ' 12', ' 98)']

('(34', ' 67', ' 55', ' 33', ' 12', ' 98)')

----TASK 4 CONTINUES--

TASK FOUR:

TRADITIONAL FUNCTIONS, ANONYMOUS FUNCTIONS &

HIGHER ORDER FUNCTION.

1.Write a program to reverse a string.

Sample data: “1234abcd”

Expected Output: “dcba4321”

def string\_reverse(str1):

rstr1 = ''

index = len(str1)

while index > 0:

rstr1 += str1[ index - 1 ]

index = index - 1

return rstr1

print('reverse string :', string\_reverse('1234abcd'))

Output:



2. Write a function that accepts a string and calculate the number of uppercase letters and lowercase letters.

Expected Output:

No. of Upper case characters : 3

No. of Lower case Characters : 12

def string\_test(s):

d={"UPPER\_CASE":0, "LOWER\_CASE":0}

for c in s:

if c.isupper():

d["UPPER\_CASE"]+=1

elif c.islower():

d["LOWER\_CASE"]+=1

else:

pass

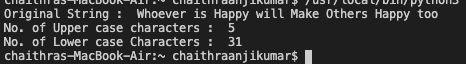
print ("Original String : ", s)

print ("No. of Upper case characters : ", d["UPPER\_CASE"])

print ("No. of Lower case Characters : ", d["LOWER\_CASE"])

string\_test('Whoever is Happy will Make Others Happy too')

Output:



3. Create a function that takes a list and returns a new list with unique elements of the first list.

def unique\_list(l):

x = []

for a in l:

if a not in x:

x.append(a)

return x

print(unique\_list([2, 3, 4, 5, 5, 5, 8, 10]))

Output:



4. Write a program that accepts a hyphen-separated sequence of words as input and prints the words in a hyphen-separated sequence after sorting them alphabetically.

words=[n for n in input().split('-')]

words.sort()

print('-'.join(words))

Output:



5. Write a program that accepts a sequence of lines as input and prints the lines after making all characters in the sentence capitalized.

Sample input:

Hello world

Practice makes perfect

Expected Output:

HELLO WORLD

PRACTICE MAKES PERFECT

sequ = []

while True:

ls = input()

if ls:

sequ.append(ls.upper())

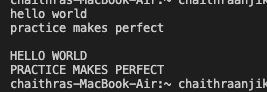
else:

break;

for ls in sequ:

print(ls)

Output:



6. Define a function that can receive two integral numbers in string form and compute their sum and print it in console.

def calculateSum (a,b):

s = int(a) + int(b)

return s

# two integral numbers as strings

num1 = "40"

num2 = "60"

# calculate sum

sum = calculateSum (num1, num2)

print ("Sum = ", sum)

7. Define a function that can accept two strings as input and print the string with maximum length in console. If two strings have the same length, then the function should print all strings line by line.

def length\_of\_string(str1, str2):

if (len(str1) == len(str2)):

print(str1)

#print("\n")

print(str2)

elif (len(str1) < len(str2)):

print(str2)

else:

print(str1)

stri1 = input(str("enter First String: "))

stri2 = input(str("enter Second String: "))

print("\n")

length\_of\_string(stri1, stri2)

8. Define a function which can generate and print a tuple where the value are square of numbers between 1 and 20.

def printTuple():

li=list()

for i in range(1,21):

li.append(i\*\*2)

print (tuple(li))

printTuple()

Output:



9. Write a function called showNumbers that takes a parameter called limit. It should print all the numbers between 0 and limit with a label to identify the even and odd numbers.

Example: If the limit is 3 , it should print:

0 EVEN

1 ODD

2 EVEN

3 ODD

limit = int(input("Enter the limit: "))

def showNumbers(limit):

for i in range(0,limit +1):

if i % 2 == 0:

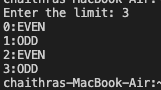
print(str(i) + ":" +'EVEN')

else:

print(str(i) + ":"+ 'ODD')

showNumbers(limit)

Output:



10.Write a program which can filter() to make a list whose elements are even number between 1 and 20 ( both included)

ls1 = range(1,20)

is\_even = lambda x: x % 2 == 0

# using filter

ls2 = list(filter(is\_even, ls1))

# Printing output

print(ls2)

Output:



11. Write a program which can map() and filter() to make a list whose elements are square of even number in [1,2,3,4,5,6,7,8,9,10]

Hints: Use map() to generate a list.

Use filter() to filter elements of a list

Use lambda to define anonymous functions

li = [1,2,3,4,5,6,7,8,9,10]

eve\_num = list(map(lambda x: x\*\*2, filter(lambda x: x%2==0, li)))

print(eve\_num)

Output:



12. Write a function to compute 5/0 and use try/except to catch the exceptions

try:

result = 5/0

print("Result: ", result)

except ZeroDivisionError:

print("Exception Handler for ZeroDivisionError")

print("We cant divide a number by 0")

13. Flatten the list [[1,2,3].,[4,5],[6,7,8]] into [1,2,3,4,5,6,7,8] using reduce

Goal : Turn [1,2,3,4,5,6,7] to 1234567

import operator

from functools import reduce

lists = [[1, 2, 3], [4, 5], [6, 7, 8]]

joinedlist = reduce(operator.add, lists)

print(joinedlist)

Output:



14.

(i) def foo():

try:

return 1

finally:

return 2

k = foo()

print(k)

Output: 2

(ii) def a():

try:

f(x, 4)

finally:

print('after f')

print('after f?')

a()

1. The above code throughs Name Error, because ‘f’ is not defined.