DAY-1

Day-1

1.Input your name into a variable called $name and then print "Hello, <your name here>".

name=input("enter your name:")

print("Hello",name)

o/p:

enter your name:komali

Hello,komali

----- ------------------------------------------------------------

2. Write a program that adds two numbers and then prints out whether the sum of those two numbers is positive or negative.

num1=int(input("enter a number:"))

num2=int(input("enter another number:"))

sum=num1+num2

x=0

if(sum>x):

print("sum is possitive")

else:

print("sum is negative")

o/p:

enter a number:54

enter another number:12

sum is positive

-----------------------------------------------------------------

3. Write a program that stores a number and keeps trying to get user input until the user enters the number correctly. As soon as the correct number is entered, it prints: Correct!

i=9

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

while True:

if(n!=i):

n=int(input("enter another number:"))

else:

print("correct")

break

o/p:

enter a number:6

enter another number:3

correct

-----------------------------------------------------------------

4. Input your first name and last name as two separate variables, labeled as $firstname and $lastname respectively. Concatenate them together using the dot operator '.' into a new variable called $wholename. Then print out the $wholename.

first\_name=input("enter firstname:")

last\_name=input("enter lastname:")

whole\_name = "{0} {1}".format(first\_name,last\_name)

print(whole\_name)

o/p:

enter firstname: komali

enter lastname:pavuluri

komali pavuluri

-----------------------------------------------------------------

5. Write a program to accept an input string from the user and toggle the character cases.

For example, $str=” Hello How Are You?”

O/p : hELLO hOW aRE yOU

string=input("enter string :")

print(string.swapcase())

o/p:

enter string :How are YOU?

hOW ARE you?

------------------------------------------------------------------------------

6. Write a program which will perform sum and multiplication ,that sums and multiplies (respectively) all the numbers in a list of numbers. For example, sum([1, 2, 3, 4]) should return 10, and multiply([1, 2, 3, 4]) should return 24.

a=[]

x=int(input("enter the number of numbers u want to enter into the list"))

i=0

for i in range(x):

a.append(int(input("enter number:")))

b=0

for i in range(x):

b+=a[i]

print(b)

b=1

for i in range(x):

b\*=a[i]

print(b)

o/p:

enter the number of numbers u want to enter into the list 3

enter numbers:1

enter numbers:3

enter numbers:6

10

18

-----------------------------------------------------------------------------

7. Write a program that takes a value (i.e. a number, string, etc) x and a list of values a, and returns True if x is a member of a, False otherwise. (Note that this is exactly what the in operator does, but for the sake of the exercise you should pretend Python did not have this operator.)

list=input("enter element into list:").split()

print(list)

f=0

x=input("enter value:")

for y in list:

if x == y :

f=1

break

if f==1:

print('present')

else:

print('not present')

o/p:

enter element into list:6 5 7 8 3

['6', '5', '7', '8' , '3' ]

enter value:5

present

-----------------------------------------------------------------------------

8. Write a program that has two lists and print True if they have at least one member in common, False otherwise. You may use your is\_member() function, or the in operator, but for the sake of the exercise, you should (also) write it using two nested for-loops.

l1=['a','b','c','d'];

l2=['e','f','b','g'];

l3=['a','b','c','d'];

l4=['e','f','g','h'];

def is\_member1(x):

if (x in l2):

return('True')

def is\_member1(y):

if (y in l4):

return('True')

print('using Is\_member function')

print()

print('l1= ',l1);

print();

print('l2 = ',l2);

for i in l1:

m=is\_member1(i)

if m=='True':

print(i,' is present')

break;

if m=='True':

print(m)

else:

print('False')

print()

print()

o/p:

using Is\_member function

l1=['a' , 'b', 'c' , 'd' ]

l2=['e' , 'f' , 'b' , 'g' ]

b is present

True

l3=['a' , 'b' , 'c' , 'd' ]

l4=['e' , 'f' , 'g' , 'h' ]

False

------------------------------------------------------------------------------

9. Write a program for histogram that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:

\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

def histogram(lista):

for b in lista:

line=" "

c=0

while c<b:

c+=1

line+="\*"

print(line)

arr=[]

print("input the number of \* you want in each line")

print("input end to finish")

while True:

n=input("input the number of \* in this line:")

if n=="end":

break

else:

n=int(n)

arr.append(n)

histogram(arr)

o/p:

input the number of \* you want in each line

input end to finish

input the number of \* in this line:5

input the number of \* in this line:2

input the number of \* in this line:end

\*\*\*\*\*

\*\*

---------------------------------------------------------------------------------------------------------------------

Day 2

DAY-2

1.Define a function generate\_n\_chars() that takes an integer n and a character c and returns a string, n characters long, consisting only of c:s. For example, generate\_n\_chars(5,"x") should return the string "xxxxx". (Python is unusual in that you can actually write an expression 5 \* "x" that will evaluate to "xxxxx". For the sake of the exercise you should ignore that the problem can be solved in this manner.)

char1=input("enter a character :")

int1=int(input("enter a number:"))

def generate\_n\_chars(int1,char1):

for i in range(0,int1):

print(char1,end=" ")

gnerate\_n\_chars(int1,char1)

o/p:

enter a character:u

enter a number:6

u u u u u u

---------------------------------------------------------------------------------------------------------------------

2.The function max() from exercise 1) and the function max\_of\_three() from exercise 2) will only work for two and three numbers, respectively. But suppose we have a much larger number of numbers, or suppose we cannot tell in advance how many they are? Write a function max\_in\_list() that takes list of numbers and returns the largest one.

def max\_in\_list(lista):

return max(lista)

lista=[]

print("write break to finish")

while True:

num=input("introduce a num:")

if num=='break':

break

else:

num=int(num)

lista.append(num)

print("max num in list:",max\_in\_list(lista))

o/p:

write break to finish

introduce a num:5

introduce a num:4

introduce a num:100

introduce a num:break

max num in list:100

---------------------------------------------------------------------------------------------------------------------

3. Write a program that maps a list of words into a list of integers representing the lengths of the correponding words.

list1=input("enter list of words:").split()

print(list1)

for i in list1

print(len(i),end=" ")

o/p:

enter list of words:komali krupa

['komali' , 'krupa' ]

6 5

--------------------------------------------------------------------------------------------------------------------

4. Write a function find\_longest\_word() that takes a list of words and returns the length of the longest one. Modify the same to do with lambda expression.

def find\_longest\_word(list1):

c=0

for i in list1:

if(len(i)>c):

c=len(i)

return c

list1=input("enter words").split()

print(list1)

print("largest word length:", find\_longest\_word(list1))

print("using lamda")

length = 0

word=lamda x:True,max(list1, key = len)

length = len(word[1])

print("The longest length is :",length)

o/p:

enter words komali krupa

['komali' , 'krupa' ]

largest word length: 6

using lamda

The longest length is :6

------------------------------------------------------------------------------

5. Write a function filter\_long\_words() that takes a list of words and an integer n and returns the list of words that are longer than n.

Modify the same to do with lambda expression.

def filter\_long\_words(string, number):

return list(filter(lambda word:len(word)>number, string.split()))

words = input("Enter list of words:")

num = int(input("Enter an integer:"))

print("Words with length greater than input integer are:", filter\_long\_words(words, num))

output:

Enter list of words: capgemini training

Enter an integer:8

Words with length greater than input integer are: ['capgemini']

----------------------------------------------------------------------------------------------------

6. Write a version of a palindrome recognizer that also accepts phrase palindromes such as

"Go hang a salami I'm a lasagna hog.",

"Was it a rat I saw?",

"Step on no pets",

"Sit on a potato pan, Otis",

"Lisa Bonet ate no basil",

"Satan, oscillate my metallic sonatas",

"I roamed under it as a tired nude Maori",

"Rise to vote sir", or the exclamation

"Dammit, I'm mad!".

Note that punctuation, capitalization, and spacing are usually ignored.

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

string = string.replace(" ","")

string=''.join(i for i in string if i.isalnum())

string=string.lower()

revstring=string[::-1]

if(string == revstring):

print("Palindrome")

else:

print("Not Palindrome")

----------------------------------------------------------------------------------------------------------

7. A pangram is a sentence that contains all the letters of the English alphabet at least once,

for example: The quick brown fox jumps over the lazy dog.

Your task here is to write a function to check a sentence to see if it is a pangram or not.

def pangran(arg):

if len(set('abcdefghijklmnopqrstuvwxyz')-set(arg.lower()))==0:

return True

else:

return False

str = input("Enter a string to check pangram:")

if(pangram(str)):

print("it is pangram")

else:

print("it is not pangram")

output:

Enter a string to check pangram: The quick brown fox jumps over the lazy dog

it is pangram

----------------------------------------------------------------------------------------------

8. Represent a small bilingual lexicon as a Python dictionary in the following fashion

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

and use it to translate your Christmas cards from English into Swedish.

That is, write a function translate() that takes a list of English words and returns a list of Swedish words.

def translate():

b=input("Enter the words you want to translate:").split(" ")

c=[]

for i in b:

c.append(dic[i])

print(c)

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

translate()

output:

happy christmas

['gott', 'jul']

-------------------------------------------------------------------------------------------------

9. Write a function char\_freq() that takes a string and builds a frequency listing of

the characters contained in it.

Represent the frequency listing as a Python dictionary.

Try it with something likechar\_freq("abbabcbdbabdbdbabababcbcbab").

def char\_freq(inp):

char\_dict={}

for c in inp:

if c in char\_dict:

char\_dict[c]+=1

else:

char\_dict[c]=1

return char\_dict

inp=input("Enter a nstring:")

print(char\_freq(inp))

output:

Enter a String: saikrupa

{'s': 1, 'a': 2, 'i': 1, 'k': 1, 'r': 1, 'u': 1, 'p': 1}

--------------------------------------------------------------------------------------

10. Create a module called mathematics.py and provide subroutines

(should be defined generally and should work for any number of arguments)

such as:

Add Sub Sort the values Max Sort

Use the module in a program and apply the functions on two array variables

(say a and b) to:

Add two arrays (and store it in c)

Subtract two arrays(and store it in d)

Find the minimum and maximum value of the

resultant array (c or d) Sort the resultant array(c or d)

imort mathematics as m

c=m.Add([1,2,3,4],[3,2,1,4])

d=m.Sub([1,2,3,4],[1,5,2,4])

print(c)

print(d)

print(m.Max(c))

print(m.Min(d))

print(m.Sort(c))

print(m.Sort(d))

output:

[4,4,8,11] [-2,0,-2,-3]

[4,4,8,11]

11

-3

------------------------------------------------------------------------

11. Try above programe with package.

import math1

a= [1,2,3,4]

b= [3,2,5,7]

c= Add(a,b)

d= Sub(a,b)

print(c,d)

print(Sort(c))

print(Max(c))

print(Min(d))

output:

[4,4,8,11] [-2,0,-2,-3]

[4,4,8,11]

11

-3

---------------------------------------------------------------------------------

12. Create a Date class, which represents the Date with its attributes.

Write a UseDate class, which makes use of the Date class to instantiate,

and call methods on the object.

class DateClass:

def\_\_init\_\_(self,date,month,year):

self.date=date

self.month=month

self.year=year

def mon(self):

print("Months=",self.month)

def datee(self):

print("Date=",sellf.date)

def yeaar(self):

print("Year=",sellf.year)

dc=DateClass(10,'December',2018)

dc.datee()

dc.mon()

dc.yeaar()

output:

Date= 10

Month= December

Year= 2018

-------------------------------------------------------------

13. WAP to read data from one file and writes in second file.

f0=open('D\day1\_13\in.txt','w+')

f0.write("Capgemini training")

f0.close

f0=open('D\day1\_13\in.txt','r')

f1=open('D\day1\_13\out.txt','w+')

for line in f0:

f1.werite(line)

f1=open('D\day1\_13\out.txt','r')

str=f1.read()

print(str)

f1.close()

f0.close()

output:

Capgemini training

-------------------------------------------------------------------------

14. WAP which will display diffrenent function of math and numpy library.

import numpy as np

list = [1,2,3,4,5,6,7]

matrix = [[5,10,15,20,25],[30,35,40,45,50],[55,60,65,70,75]]

print('array='+str(np.array((list))))

print('array of zeros='+str(np.zeros([5,5])))

print('unit matrix='+str(np.eye(4)))

print('arange function='+str(np.arange(1,10,2)))

print('random function='+str(np.random.rand(4)))

print('random function='+str(np.random.radiant(1,100,10)))

a=np.arange(1,10)

print('max value='+str(np.max(a)))

print('min value='+str(np.min(a)))

print('mean value='+str(np.mean(a)))

print('sqr root='+str(np.sqrt(a)))

print('power='+str(np.power(a,2)))

import math as m

print('ceiling function='+str(m.ceil(100.32)))

print('floor fun='+str(m.floor(100.45)))

print('exponent fun='+str(m.exp(100.12)))

print('absolute fun='+str(m.fabs(-35.13)))

print('fraction fun='+str(m.modf(100.15)))

print('power fun='+str(m.pow(2,4)))

print('square root='+str(m.sqrt(10)))

print('sine fun='+str(m.sin(9)))

print('cosine fun='+str(m.cos(15)))