1. Write a program to reverse a string. Sample input: “1234abcd” Expected output: “dcba4321”

text="Catenation"[::-1]

print(text)

1. Write a function that accepts a string and prints the number of uppercase letters and lowercase letters. Sample input: “abcSdefPghijQkl” Expected Output: No. of Uppercase characters : 3 No. of Lower case Characters : 12

a=input("Enter your string:")

countUpperCase=countl=0

for i in a:

    if i.isupper():

        countUpperCase+=1

print("Number of upper case letters:",countUpperCase)

for i in a:

    if i.islower():

        countl+=1

print ("Number of lower case letters:",countl)

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

a=[1,1,2,2,3,3]

print(list(set(a)))

1. 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.

print("Enter a hyphen separated sequence of words:")

list=[n for n in input().split('-')]

list.sort()

print("Sorted:")

print('-'.join(list))

1. 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 man perfect Expected output: HELLO WORLD PRACTICE MAKES MAN PERFECT

a=input("Enter your input: Hello world practice makes man perfect")

b=a.upper()

print(b)

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

def func(a,b):

    c=int(a)

    d=int(b)

    sum=c+d

    print(sum)

a=input("Enter first number:")

b=input("Enter second number:")

func(a,b)

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

def long(a,b):

    if len(a)>len(b):

        print(a)

    elif len(a)==len(b):

        print(a)

        print(b)

    else:

        print(b)

a=input("Enter first string:")

b=input("Enter second string:")

long(a,b)

1. Define a function which can generate and print a tuple where the values are square of numbers between 1 and 20 (both 1 and 20 included).

square of numbers = [elem \*\* 2 for elem in range(1,20)]

print(square of numbers)

1. 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. Sample input: show Numbers(3) (where limit=3) Expected output: 0 EVEN 1 ODD 2 EVEN 3 ODD

def showNumber(limit):

    for i in range(0,limit):

        if i ==0:

            print(i, end="")

            print("Even")

        elif i % 2 ==0:

            print(i, end="")

            print("Even")

        else:

            print(i, end="")

            print("Odd")

print(showNumber(10))

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

x=filter(lambda x:x%2==0, range(1,20))

print("Even numbers are: ",list(x))

1. Write a program which uses map() and filter() to make a list whose elements are squares of even numbers in [1,2,3,4,5,6,7,8,9,10]. Hints: Use filter() to filter even elements of the given listUse map() to generate a list of squares of the numbers in the filtered list. Use lambda() to define anonymous functions.

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

number=map(lambda x:x \*\* 2, filter(lambda x:x %2 ==0, li))

print(list(number))

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

def divide():

    return 5/0

try: divide()

print(“Exception)

1. Flatten the list [1,2,3,4,5,6,7] into 1234567 using reduce().

from functools import reduce

x=reduce(lambda x,y: 1\*x+y,[1,2,3,4,5])

print(x)

1. Write a program in Python to find the values which are not divisible by 3 but are a multiple of 7. Make sure to use only higher order functions.

def mul\_div():

    x=list(range(0,20))

for n in list(range(0,20)):

    if n % 3!=0 and n % 7==0:

        print(n)

1. Write a program in Python to multiply the elements of a list by itself using a traditional function and pass the function to map() to complete the operation.

def function(x):

    return (x\*x)

y=map(function,(1,2,3))

print(tuple(y))

1. What is the output of the following codes: (i) def foo(): try: return 1 finally: return 2 k = foo() print(k) (ii) def a(): try: f(x, 4) finally: print('after f') print('after f?') a()

NameError: name 'f' is not defined