1. Write a program to reverse a string.

txt = "1234abcd"[::-1]

print(txt)

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

Sample input: “abcSdefPghijQkl”

*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"])*

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([1,2,3,3,3,3,4,5]))*

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.

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

*items.sort()*

*print('-'.join(items))*

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 man perfect

*# Hello world*

*# Practice makes perfect*

*# Then, the output should be:*

*# HELLO WORLD*

*# PRACTICE MAKES PERFECT*

*from string import \**

*lines = []*

*print "Enter sequence of lines: "*

*while True:*

*line = raw\_input("> ")*

*if not line:*

*break*

*lines.append(line.upper())*

*print lines*

Expected output: HELLO WORLD PRACTICE MAKES MAN PERFECT

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

*def calculateSum (a,b):*

*s = int(a) + int(b)*

*return s*

*# Main code*

*# take two integral numbers as strings*

*num1 = "10"*

*num2 = "20"*

*# calculate sum*

*sum = calculateSum (num1, num2)*

*# print sum*

*print "Sum = ", sum*

7. 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 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 values are square of numbers between 1 and 20 (both 1 and 20 included).

*def printTuple():*

*li=list()*

*for i in range(1,21):*

*li.append(i\*\*2)*

*print tuple(li)*

*printTuple()*

9. Write a function called show Numbers 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)

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(4))*

Expected output:

0 EVEN

1 ODD

2 EVEN

3 ODD

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

li=list()

for i in range(1,21):

li.append(i\*\*2)

print li

printList()

11. 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.

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

def throws():

return 5/0

*throws()*

*except ZeroDivisionError:*

*print "division by zero!"*

*except Exception, err:*

*print 'Caught an exception'*

*finally:*

*print 'In finally block for cleanup'*

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

# importing the module

import itertools

data = [[1, 2, 3,4,5]]

flat\_list = []

# iterating over the data

for item in data:

# appending elements to the flat\_list

flat\_list += item

# printing the resultantn flat\_list

print(flat\_list)

14. 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.

15. 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.

16. 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()