**EXPERIMENT 2.2**

**Aim :** To study functions in Python.

**Theory :**

1. **Program to display the powers of 2 using anonymous function.**

terms = int(input("Enter the upper limit on number of terms: "))

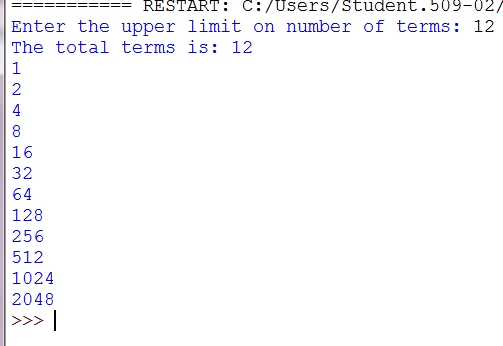
result = list(map(lambda x: 2 \*\* x, range(terms)))

print("The total terms is:",terms)

for i in range(terms):

print(result[i])

**OUTPUT :**

****

1. **Program to find the numbers divisible by another number (Factors of a number).**

mylist = [0,1050,2,30,43,455,16,75,85,99,100,111,120,13,143,135,106,37]

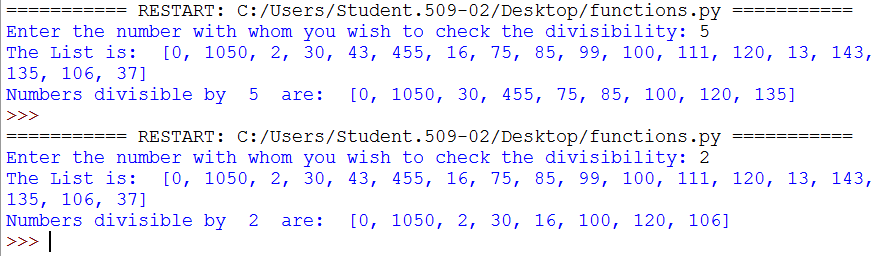
term = int(input("Enter the number with whom you wish to check the divisibility: "))

result = list(filter(lambda x:x%term == 0,mylist))

print("The List is: ",mylist)

print("Numbers divisible by ",term," are: ",result)

**OUTPUT :**

****

1. **Program to convert Binary to decimal,octal and hexadecimal.**

def convert(dec):

print("In Binary: ",bin(dec))

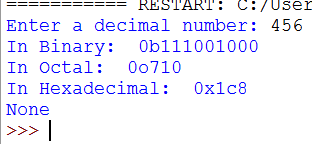
print("In Octal: ",oct(dec))

print("In Hexadecimal: ",hex(dec))

dec = int(input("Enter a decimal number: "))

print(convert(dec))

**OUTPUT :**

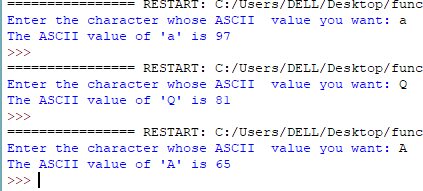
****

1. **Program to find the ASCII value of a number.**

c = input("Enter the character whose ASCII value you want: ")

print("The ASCII value of '" + c + "' is",ord(c))

**OUTPUT:**

****

1. **Program to find HCF or GCD.**

def computeHCF(x, y):

if x > y:

smaller = y

Else:

smaller = x

for i in range(1, smaller+1):

if((x % i == 0) and (y % i == 0)):

hcf = i

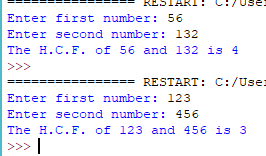
return hcf

num1 = int(input("Enter first number: "))

num2 = int(input("Enter second number: "))

print("The H.C.F. of", num1,"and", num2,"is", computeHCF(num1, num2))

**OUTPUT:**

****

1. **Program to find LCM.**

def lcm(x, y):

if x > y:

greater = x

else:

greater = y

while(True):

if((greater % x == 0) and (greater % y == 0)):

lcm = greater

break

greater += 1

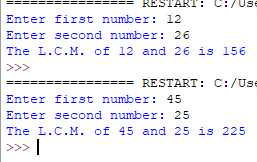
return lcm

num1 = int(input("Enter first number: "))

num2 = int(input("Enter second number: "))

print("The L.C.M. of", num1,"and", num2,"is", lcm(num1, num2))

**OUTPUT:**

****

1. **Program to find the factors of a number.**

def factors():

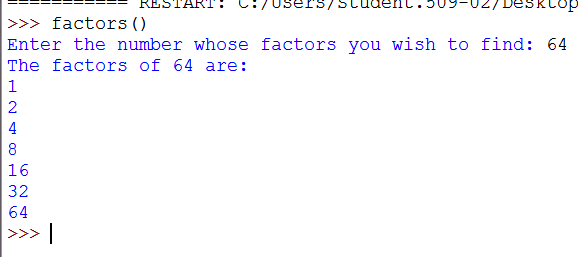
term = int(input("Enter the number whose factors you wish to find: "))

print("The factors of",term,"are:")

for i in range(1, term + 1):

if term % i == 0:

print(i)

****

1. **Program to make a simple calculator.**

def add(x, y):

return x + y

def subtract(x, y):

return x - y

def multiply(x, y):

return x \* y

def divide(x, y):

return x / y

def mod(x, y):

return x % y

print("Select operation.")

print("1.Add")

print("2.Subtract")

print("3.Multiply")

print("4.Divide")

print("5.Modulus")

choice = input("Enter choice(1/2/3/4):")

num1 = int(input("Enter first number: "))

num2 = int(input("Enter second number: "))

if choice == '1':

print(num1,"+",num2,"=", add(num1,num2))

elif choice == '2':

print(num1,"-",num2,"=", subtract(num1,num2))

elif choice == '3':

print(num1,"\*",num2,"=", multiply(num1,num2))

elif choice == '4':

print(num1,"/",num2,"=", divide(num1,num2))

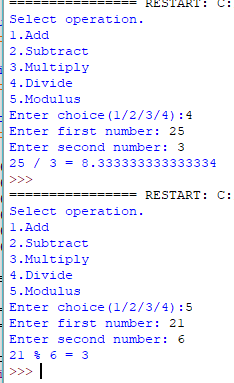
elif choice == '5':

print(num1,"%",num2,"=", mod(num1,num2))

else:

print("Invalid input")

**OUTPUT:**

****

1. **Program to shuffle deck of cards.**

import itertools, random

deck = list(itertools.product(range(1,14),['Spade','Heart','Diamond','Club']))

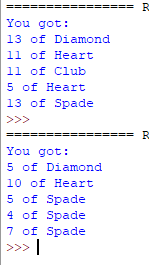
random.shuffle(deck)

print("You got:")

for i in range(5):

print(deck[i][0], "of", deck[i][1])

**OUTPUT:**

****

1. **Program to display calendar.**

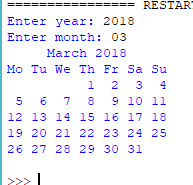
import calendar

yy = int(input("Enter year: "))

mm = int(input("Enter month: "))

print(calendar.month(yy, mm))

**OUTPUT:**

****

1. **Program to display fibonacci series using recursion.**

def recur\_fibo(n):

if n <= 1:

return n

else:

return(recur\_fibo(n-1) + recur\_fibo(n-2))

nterms = int(input("How many terms? "))

if nterms <= 0:

print("Please enter a positive integer")

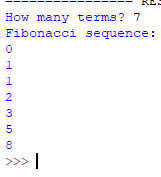
else:

print("Fibonacci sequence:")

for i in range(nterms):

print(recur\_fibo(i))

**OUTPUT:**



1. **Program to find the sum of natural numbers using recursion.**

def recur\_sum(n):

if n <= 1:

return n

else:

return n + recur\_sum(n-1)

num = int(input("Enter a number: "))

if num < 0:

print("Enter a positive number")

else:

print("The sum of first",num," natural numbers is",recur\_sum(num))

**OUTPUT:**

