Question 1:

Please write a program using generator to print the numbers which can be divisible by 5 and 7 between 0 and n in comma separated form while n is input by console.

Example:  
If the following n is given as input to the program:

100

Then, the output of the program should be:

0,35,70

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def find\_factors():

numb = int(input("Enter your number upto where you want to find multiples of both 5 and 7: "))

for i in range(numb+1):

if i%(5\*7) == 0:

yield i

try:

for special in find\_factors():

print(special,",", end="")

lg.info("Class comp\_and\_decomp() has been called")

except Exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

Question 2:

Please write a program using generator to print the even numbers between 0 and n in comma separated form while n is input by console.

Example:  
If the following n is given as input to the program:

10

Then, the output of the program should be:

0,2,4,6,8,10

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def find\_even():

numb = int(input("Enter your number upto where you want to find multiples of both 5 and 7: "))

for i in range(numb+1):

if i%2 == 0:

yield i

try:

for even in find\_even():

print(even,",", end="")

lg.info("Class comp\_and\_decomp() has been called")

except Exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

Question 3:

The Fibonacci Sequence is computed based on the following formula:

f(n)=0 if n=0  
f(n)=1 if n=1  
f(n)=f(n-1)+f(n-2) if n>1

Please write a program using list comprehension to print the Fibonacci Sequence in comma separated form with a given n input by console.

Example:  
If the following n is given as input to the program:

7

Then, the output of the program should be:

0,1,1,2,3,5,8,13

Ans

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def fibo():

my\_list = [0,1]

n = int(input("give the number of terms needed in "))

[my\_list.append(my\_list[-2]+my\_list[-1]) for n in range(n-1)]

print(my\_list)

try:

fibo()

lg.info("Class fibo() has been called")

except Exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

Question 4:

Assuming that we have some email addresses in the "[username@companyname.com](mailto:username@companyname.com)" format, please write program to print the user name of a given email address. Both user names and company names are composed of letters only.

Example:  
If the following email address is given as input to the program:

[john@google.com](mailto:john@google.com)

Then, the output of the program should be:

john

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

def get\_user(email):

user\_names = ""

for i in email:

if i == "@":

break

else:

user\_names += i

print( user\_names)

try:

get\_user("john@gmail.com")

lg.info("""Class get\_user("john@gmail.com") has been called""")

except Exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass

Question 5:

Define a class named Shape and its subclass Square. The Square class has an init function which takes a length as argument. Both classes have a area function which can print the area of the shape where Shape's area is 0 by default.

Ans.

import logging as lg

# importing logging so every function call of

lg.basicConfig(filename ='C:\\Users\\Home\\Johns python talent\\logging\\testlog1.log', level =lg.INFO , format = '%(asctime)s %(message)s')

class shape:

def \_\_init\_\_(self, length):

self.length = length

def area(self):

print("The area of the shape is :", self.length\*\*3)

# raising to the power of 3 just to show difference between area of shape and area of square

class square(shape):

def \_\_init\_\_(self, length1= None):

if length1 == None:

self.length1 = 0

else:

self.length1 = length1

def area(self):

print("The area of the square is :", self.length1\*\*2)

try:

j = shape(15)

j.area()

p = square(20)

p.area()

lg.info("""Class shape and square has been called has been called""")

except Exception as e:

print("There was an error called: ",e)

else:

pass

finally:

pass