**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**

def divisible\_by\_5\_and\_7(n):

for num in range(n + 1):

if num % 5 == 0 and num % 7 == 0:

yield num

n = int(input("Enter a value for n: "))

result = list(divisible\_by\_5\_and\_7(n))

result\_str = ",".join(map(str, result))

print(result\_str)

**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**

def even\_numbers(n):

for num in range(n + 1):

if num % 2 == 0:

yield num

n = int(input("Enter a value for n: "))

result = list(even\_numbers(n))

result\_str = ",".join(map(str, result))

print(result\_str)

**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**

def fibonacci\_sequence(n):

fib = [0, 1]

[fib.append(fib[-1] + fib[-2]) for \_ in range(2, n)]

return fib

n = int(input("Enter a value for n: "))

result = fibonacci\_sequence(n)

result\_str = ",".join(map(str, result))

print(result\_str)

**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**

def extract\_username(email):

parts = email.split('@')

if len(parts) == 2:

return parts[0]

else:

return None

email = input("Enter an email address: ")

username = extract\_username(email)

if username:

print(username)

else:

print("Invalid email format")

**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.**

class Shape:

def area(self):

return 0

class Square(Shape):

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

self.length = length

def area(self):

return self.length \* self.length

length = float(input("Enter the length of the square: "))

square = Square(length)

print("Area of the square:", square.area())