|  |
| --- |
| Question 1: |
|  |

Define a class with a generator which can iterate the numbers, which are divisible by 7, between a given range 0 and n.

class Divisible:

def divide\_by\_seven(self,num):

for i in range(0, num):

if i % 7 ==0:

yield i

d= Divisible()

t= iter(d.divide\_by\_seven(70))

next(t)

Question 2:

|  |
| --- |
| Write a program to compute the frequency of the words from the input. The output should output after sorting the key alphanumerically. |
|  |

|  |
| --- |
| Suppose the following input is supplied to the program: |
|  |

|  |
| --- |
| New to Python or choosing between Python 2 and Python 3? Read Python 2 or Python 3. |
|  |

|  |
| --- |
| Then, the output should be: |
|  |

|  |
| --- |
| 2:2 |
|  |

|  |
| --- |
| 3.:1 |
|  |

|  |
| --- |
| 3?:1 |
|  |

|  |
| --- |
| New:1 |
|  |

|  |
| --- |
| Python:5 |
|  |

|  |
| --- |
| Read:1 |
|  |

|  |
| --- |
| and:1 |
|  |

|  |
| --- |
| between:1 |
|  |

|  |
| --- |
| choosing:1 |
|  |

|  |
| --- |
| or:2 |
|  |

to:1

ss = "New to Python or choosing between Python 2 and Python 3? Read Python 2 or Python 3"

splitted = ss.split()

new\_list =[]

for i in splitted:

new\_list.append(str(i) +' ' + str(ss.count(i)))

set(sorted(new\_list))

|  |
| --- |
| Question 3: |
|  |

|  |
| --- |
|  |
|  |

Define a class Person and its two child classes: Male and Female. All classes have a method "getGender" which can print "Male" for Male class and "Female" for Female class.

class Person:

pass

class Male(Person):

def getGender():

print('Male class method')

class Female(Person):

def getGender():

print('Female class method')

Question 4:

Please write a program to generate all sentences where subject is in ["I", "You"] and verb is in ["Play", "Love"] and the object is in ["Hockey","Football"].

list1 = ["I", "You"]

list2=["Play", "Love"]

list3=["Hockey","Football"]

for i in list1:

for j in list2:

for k in list3:

print("{} {} {}".format(i,j,k) )

Question 5:

Please write a program to compress and decompress the string "hello world!helloworld!helloworld!hello world!".

import gzip

s = b"hello world!helloworld!helloworld!hello world!"

s = gzip.compress(s)

s

t = gzip.decompress(s)

print(t)

Question 6:

Please write a binary search function which searches an item in a sorted list. The function should return the index of element to be searched in the list.

from bisect import bisect\_left

def BinarySearch(a, x):

i = bisect\_left(a, x)

if i != len(a) and a[i] == x:

return i

else:

return -1

a = [1, 2, 4, 4, 8]

x = int(1)

res = BinarySearch(a, x)

if res == -1:

print(x, "is absent")

else:

print("First occurrence of", x, "is present at", res)