|  |
| --- |
| **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 DivisibleBySeven:

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

self.n = n

def generate\_divisible\_by\_seven(self):

for num in range(self.n + 1):

if num % 7 == 0:

yield num

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

divisible\_by\_seven = DivisibleBySeven(n)

for num in divisible\_by\_seven.generate\_divisible\_by\_seven():

print(num)

**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**

input\_text = input("Enter a sentence: ")

words = input\_text.split()

word\_freq = {}

for word in words:

word = word.strip('.,?')

word\_freq[word] = word\_freq.get(word, 0) + 1

sorted\_word\_freq = sorted(word\_freq.items())

for word, freq in sorted\_word\_freq:

print(f"{word}:{freq}")

|  |
| --- |
| **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:

def getGender(self):

return "Unknown"

class Male(Person):

def getGender(self):

return "Male"

class Female(Person):

def getGender(self):

return "Female"

male = Male()

female = Female()

print(male.getGender())

print(female.getGender())

**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"].**

subjects = ["I", "You"]

verbs = ["Play", "Love"]

objects = ["Hockey", "Football"]

sentences = [(subject, verb, obj) for subject in subjects for verb in verbs for obj in objects]

for sentence in sentences:

print(f"{sentence[0]} {sentence[1]} {sentence[2]}")

**Question 5:**

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

import zlib

text = "hello world!hello world!hello world!hello world!"

compressed\_text = zlib.compress(text.encode('utf-8'))

decompressed\_text = zlib.decompress(compressed\_text).decode('utf-8')

print("Original text:", text)

print("Compressed text:", compressed\_text)

print("Decompressed text:", decompressed\_text)

**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.**

def binary\_search(sorted\_list, target):

low = 0

high = len(sorted\_list) - 1

while low <= high:

mid = (low + high) // 2

if sorted\_list[mid] == target:

return mid

elif sorted\_list[mid] < target:

low = mid + 1

else:

high = mid - 1

return -1

sorted\_list = [1, 3, 5, 7, 9, 11, 13]

target = 7

index = binary\_search(sorted\_list, target)

if index != -1:

print(f"Element {target} found at index {index}")

else:

print(f"Element {target} not found in the list")