

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

Ans:

```
In [11]: 1 class divisible_7:
2         n = int(input('Enter a number: '))
3         def __init__(self, num):
4             self.num = num
5         def generate(self):
6             for i in range(0, self.num+1):
7                 if i%7==0:
8                     print(i, ' ')
9         d = divisible_7(n)
10        d.generate()

Enter a number: 49
0
7
14
21
28
35
42
49
```

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

Ans:

```
In [1]: 1 string = input('Enter a setence:')
2         words = string.split(' ')
3         words.sort()
4         count = 0
5         for i in range(0, len(words)):
6             count = 1
7             for j in range(i+1, len(words)):
8                 if words[i] == words[j]:
9                     count += 1
10                    words = words[:j] + [0] + words[j+1:]
11            print(words[i], ': ', count, end=" ")

Enter a setence:New to Python or choosing between Python 2 and Python 3
2 : 2 0 : 1 3 : 1 3? : 1 New : 1 Python : 5 0 : 4 0 : 3 0 : 2 0 : 1 Read : 1 and : 1 between : 1 choosing : 1 or : 2 0 : 1 to :
1
```

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. ?

Ans:

```
In [2]: 1 class Person():
2         def getGender():
3             pass
4     class Male(Person):
5         def getGender():
6             print("Male")
7     class Female(Person):
8         def getGender():
9             print("Female")
10    Male.getGender()
11    Female.getGender()
```

Male
Female

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"] ?

Ans:

```
In [4]: 1 def generate():
2         subject = ['I','You']
3         verb = ['Play','Love']
4         object = ['Hockey','Football']
5         for s in subject:
6             for v in verb:
7                 for o in object:
8                     print(s,v,o)
9     generate()
```

I Play Hockey
I Play Football
I Love Hockey
I Love Football
You Play Hockey
You Play Football
You Love Hockey
You Love Football

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

Ans:

```
In [2]: 1 import gzip
2     text = b'hello world!hello world!hello world!hello world!'
3     comp=gzip.compress(text)
4     print("Compressed: ", comp)
5     decomp=gzip.decompress(comp)
6     print("Decompressed: ", decomp)
```

Compressed: b'\x1f\x8b\x08\x00\x00\x00\x00\xff\xcbH\xcd\x09\x09W(\xc9\xcaIQ\xcc \x82\r\x00\xd3\xe6\xcf\xae0\x00\x00\x00'
Decompressed: b'hello world!hello world!hello world!hello world!'

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.

Ans:

```
In [4]: 1 string = input('Enter a setence:')
        2 words = string.split(' ')
        3 words.sort()
        4 word = input('Enter word to be found:')
        5 def binary_search(w):
        6     for i in range(0,len(words)):
        7         if words[i] == word:
        8             return i
        9     binary_search(word)
```

```
Enter a setence:hello world
Enter word to be found:world
```

```
Out[4]: 1
```