Python programming sheet - 2

1. The function max() from exercise 1) and the function max_of_three() from exercise 2) will only work for two and three numbers, respectively. But suppose we have a much larger number of numbers, or suppose we cannot tell in advance how many they are? Write a function max_in_list() that takes a *list* of numbers and returns the largest one.

2. Dictionary and List Methods.
   i. Create a dictionary and display its keys alphabetically.
   ii. Now display both the keys and values sorted in alphabetical order by the key.
   iii. Same as part (b), but sorted in alphabetical order by the value. (Note: this generally has no practical purpose in dictionaries or hash tables in general because most access and ordering [if any] is based on the keys. This is merely an exercise.)

3. Inverting Dictionaries. Take a dictionary as input and return one as output, but the values are now the keys and vice versa.

4. Create a dictionary of 20 random values in the range 1–99. Determine whether there are any duplicate values in the dictionary.

5. In cryptography, a *Caesar cipher* is a very simple encryption techniques in which each letter in the plain text is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 3, A would be replaced by D, B would become E, and so on. The method is named after Julius Caesar, who used it to communicate with his generals. *ROT-13* ("rotate by 13 places") is a widely used example of a Caesar cipher where the shift is 13. In Python, the key for ROT-13 may be represented by means of the following dictionary:
   key = {'a':'n', 'b':'o', 'c':'p', 'd':'q', 'e':'r', 'f':'s', 'g':'t', 'h':'u', 'i':'v', 'j':'w', 'k':'x', 'l':'y', 'm':'z', 'n':'a', 'o':'b', 'p':'c', 'q':'d', 'r':'e', 's':'f', 't':'g', 'u':'h', 'v':'i', 'w':'j', 'x':'k', 'y':'l', 'z':'m', 'A':'N', 'B':'O', 'C':'P', 'D':'Q', 'E':'R', 'F':'S', 'G':'T', 'H':'U', 'I':'V', 'J':'W', 'K':'X', 'L':'Y', 'M':'Z', 'N':'A', 'O':'B', 'P':'C', 'Q':'D', 'R':'E', 'S':'F', 'T':'G', 'U':'H', 'V':'I', 'W':'J', 'X':'K', 'Y':'L', 'Z':'M'}
   Your task in this exercise is to implement an encoder/decoder of ROT-13. Once you're done, you will be able to read the following secret message:
   Pnrfne pvcure? V zhpu cersre Pnrfne fnynq!
   Note that since English has 26 characters, your ROT-13 program will be able to both encode and decode texts written in English.
   Output :
   ```
   >>>rot13.py
   >>>Enter string to rot13: This is a short sentence.
   >>>Your string to en/decrypt was: [This is a short sentence.].
   >>>The rot13 string is: [Guvf vf n fubeg fragrapr.].
   ```

6. Write a method rand divis 3 that takes no parameters, generates and prints a random number, and finally returns True if the randomly generated number is divisible by 3, and False otherwise. For this method we'll use a new module, the random module. At the top of your code, underneath import math, add the line import random. We'll use this module to generate a random integer using the function randint, which works as follows:

   random.randint(lo, hi) ,where lo and hi are integers that tell the code the range in which to generate a random integer (this range is inclusive). 0 to 100 is probably a decent range.

7. Write a method roll dice that takes in 2 parameters -the number of sides of the die, and the number of dice to roll -and generates random roll values for each die rolled. Print out each roll and then return the string “That’s all!” An example output:

   ```
   >>> roll_dice(6, 3)
   4
   1
   6
   That’s all!
   ```

8. For this exercise, you will be coding your very first class, a Queue class. Create a new file called queue.py to make your Queue class. In your Queue class, you will need three methods:

   init : to initialize your Queue (think: how will you store the queue’s elements? You’ll need to initialize an appropriate object attribute in this method)

   insert: inserts one element in your Queue

   remove: removes one element from your Queue and returns it. If the queue is empty, return a message that says it is empty.