

Programming Exercises (Dictionaries)

1. Write a function called `removekeys(mydict, keylist)` that accepts two parameters: a dictionary called `mydict` and a list called `keylist`. `removekeys(mydict, keylist)` should remove all the keys contained in `keylist` from `mydict` and return the dictionary.
2. Write a function called `accept_login(users, username, password)` with three parameters: `users` a dictionary of username keys and password values, `username` a string for a login name and `password` a string for a password. The function should return `True` if the user exists and the password is correct and `False` otherwise.
3. Write a function called `findvalue(mydict, val)` that accepts a dictionary called `mydict` and a variable of any type called `val`. The function should return a list of keys that map to the value `val` in `mydict`.
4. Write a function called `wordfrequencies(mylist)` that accepts a list of strings called `mylist` and returns a dictionary where the keys are the words from `mylist` and the values are the number of times that word appears in `mylist`.
5. In bioinformatics a k-mer is a substring of k characters from a string that is longer than k (lookup k-mer in the web for details). Write a function with two parameters: a string containing DNA and the value of k. Return a dictionary of k-mer counts.
6. 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.