1. 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

    {'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.

2. Given a dictionary of students and their favourite colours:

    people={'Arham':'Blue','Lisa':'Yellow','Vinod:'Purple','Jenny':'Pink'}
    1. Find out how many students are in the list
    2. Change Lisa's favourite colour
    3. Remove 'Jenny' and her favourite colour
    4. Sort and print students and their favourite colours alphabetically by name

3. Write a menu driven program to practice Dictionary functions.
    Write a program to accept name of a person and their vehicle and store it in a dictionary.
    Ask user if they want to continue to accept multiple values.
    Display following menu:
    1. Add new person name and a vehicle name.
    2. Delete a person name and vehicle name from the dictionary.
        ----Accept person name from user.
        ----Check whether person name exists in the dictionary.
        ----If exists show person name and vehicle name to the user.
        ----Confirm for deletion, if user enters y
            then delete otherwise no. Display appropriate message.
    3. Modify vehicle name for the person
        ----Accept a person name from user.
        ----Check whether the person's name exists.
        ----If the name exists, show the person's name and vehicle name to user.
            Ask for new value and then overwrite the old value.
    4. Search vehicle for the given person's name.
    5. Search list of people, who have given a vehicle
    6. Display all person names.
    7. Display all vehicle names.
    8 Exit

4. Write a program to display following menu and do the following:
   1. Add new city and trees commonly found in the city.
   2. Display all cities and the list of trees for all cities.
   3. Display list of trees of a particular city.
      ---- Accept a city from user search city and if found display list of trees otherwise
      ---- Display message not found
   4. Display cities which have the given tree.
      ---- Accept a tree name from user and display all cities in which the tree is found.
   5. Delete city
      ---- Accept city from user and delete the city if found.
      ---- Prompt user before deletion
   6. Modify tree list
      ---- Accept city and trees to be added in the city. if city exist add trees at the end
        of the list
      ---- Otherwise add city and list
   7. Exit