Question1

Create a function that takes three integer arguments (a, b, c) and returns the amount of integers which are of equal value.

**Examples**

equal(3, 4, 3) ➞ 2

equal(1, 1, 1) ➞ 3

equal(3, 4, 1) ➞ 0

**Notes**

Your function must return 0, 2 or 3.

ANSWER 1:

def equal( a , b, c):

result = 0

if(a == b):

result = 2

if( b ==c ):

result =3

elif( a == c):

result =2

elif( b == c ):

result = 2

else:

result = 0

return result

Question2

Write a function that converts a **dictionary** into a **list** of keys-values **tuples**.

### Examples

dict\_to\_list({

"D": 1,

"B": 2,

"C": 3

}) ➞ [("B", 2), ("C", 3), ("D", 1)]

dict\_to\_list({

"likes": 2,

"dislikes": 3,

"followers": 10

}) ➞ [("dislikes", 3), ("followers", 10), ("likes", 2)]

### Notes

Return the elements in the list in alphabetical order.

ANSWER 2

def dict\_to\_list( dict):

lst1 = []

lst2 = []

for i in dict.keys():

lst1.append( i )

lst1.sort()

for i in lst1:

tmp = ( i , dict[i] )

lst2.append( tmp)

return lst2

Question3

Write a function that creates a dictionary with each **(key, value)** pair being the **(lower case, upper case)** versions of a letter, respectively.

### Examples

mapping(["p", "s"]) ➞ { "p": "P", "s": "S" }

mapping(["a", "b", "c"]) ➞ { "a": "A", "b": "B", "c": "C" }

mapping(["a", "v", "y", "z"]) ➞ { "a": "A", "v": "V", "y": "Y", "z": "Z" }

### Notes

All of the letters in the input list will always be lowercase.

ANSWER3

def mapping( input\_list ):

dict = {}

for item in input\_list:

dict[ item ] = item.upper()

return dict

Question4

Write a function, that replaces all vowels in a string with a specified vowel.

### Examples

vow\_replace("apples and bananas", "u") ➞ "upplus und bununus"

vow\_replace("cheese casserole", "o") ➞ "chooso cossorolo"

vow\_replace("stuffed jalapeno poppers", "e") ➞ "steffed jelepene peppers"

### Notes

All words will be lowercase. Y is not considered a vowel.

ANSWER4

def vow\_replace( str , vowel ):

res\_str = ""

for i in str:

if i in ( 'a' , 'e' , 'i' , 'o' , 'u '):

res\_str = res\_str + vowel

else:

res\_str = res\_str + i

return res\_str

Question5

Create a function that takes a string as input and capitalizes a letter if its ASCII code is even and returns its lower case version if its ASCII code is odd.

### Examples

ascii\_capitalize("to be or not to be!") ➞ "To Be oR NoT To Be!"

ascii\_capitalize("THE LITTLE MERMAID") ➞ "THe LiTTLe meRmaiD"

ascii\_capitalize("Oh what a beautiful morning.") ➞ "oH wHaT a BeauTiFuL moRNiNg."

ANSWER5

def ascii\_capitalize( str ):

res\_str = ""

for i in str:

if (ord(i))%2 !=0:

res\_str = res\_str + i.lower()

else:

res\_str = res\_str + i.upper()

return res\_str