**尋找單字中重複最多次的字母**

**目的:**傳入一個單字,傳回重複最多次的單字與次數

**程式碼:**

import operator

def most\_repeated\_letter(word):

count\_=[]#儲存字母重複的次數

letters=list(**set(**word**)**)#轉串列

for letter in letters:

count\_.append((letter,word.**count(**letter**)**))

return max(count\_,key=operator.itemgetter(1))

a=most\_repeated\_letter('independence')

print(f'{a[0]}重複了{a[1]}次')

**取得字母中所有不重複的字母 : set(集合)**

**set的元素(鍵)決不會重複,不管你加多少鍵都一樣。所以我們藉此建立單字的『字母表』。**

eg: s = ‘ independence’

print(s) #{ ‘e’ , ’p’ , ‘d’ , ’c’ , ‘i’ , ’n’}

**計算元素在容器出現的次數 :** 可用list 或 str 物件的 **count()** method 來計算某元素出現次數

s= ’independence’

print(s.count(‘e’)) # 4

**延伸技巧 : 使用Counter計數器來統計**

**from collections import Counter**

**s= ’independence’**

**letter\_count = Counter(s)**

**print(letter\_count)**

# Counter({‘e’: 4 , ‘n’: 3 , ‘d’: 2 , ‘I’: 1 , ‘p’: 1 , ‘c’: 1})

most\_common()也可以填入參數 , 指定要傳回幾筆結果 :

print(letter\_counts.most.commond(1))

# [(‘e’ , 4)]

**程式碼 :**

**from collections import Counter**

def most\_repeated\_letter(word):

result=**Counter(word).most\_common(1)[0]** # result=[('e',4)] , result[0]=('e',4)

print(f'{result[0]}重複了{result[1]}次')

most\_repeated\_letter('independence')