

고급통계프로그래밍 # 6

2017580034 통계학과 이유민

(#1) Ex12.1

In [1]:

```
def sumall(*x):          #가변길이 인자 받아서
    x = tuple(sorted(x)) #들어오는 숫자를 정렬하고
    ans = 0              #for문을 돌면서 합치자
    for i in x:
        ans+=i           #숫자 크기만큼 ans에 추가시켜서
    return ans           #return
```

In [2]:

```
sumall(6, 324, 32, 3, 49, 4, 34, 12, 3)
```

Out[2]:

467

(#2) Ex12.2

In [3]:

```
import random
```

In [4]:

```
# 원래 함수
def sort_by_length(words):
    t = []
    for word in words:
        t.append((len(word), word))

    t.sort(reverse=True) #내림차순 정렬
    #같은 길이의 단어는 역순으로 출력

    res = []
    for length, word in t:
        res.append(word)
    return res
```

In [5]:

```
# 수정한 함수 (답안)
def sort_by_length(words):
    t = []
    for word in words:
        t.append((len(word), random.random(), word))
        # 원래 함수의 append에 random 추가 > 임의의 순서로 나타나도록

    t.sort(reverse=True) #내림차순 정렬

    res = []
    for length, _, word in t:
        res.append(word)
    return res
```

In [6]:

```
t=['John', 'Eric', 'Graham', 'Terry', 'terry', 'Michael']
sort_by_length(t)
```

Out[6]:

```
['Michael', 'Graham', 'Terry', 'terry', 'John', 'Eric']
```

(#3) Ex12.3

In [7]:

```
def most_frequent(s):
    d = dict() # 빈 딕셔너리를 정의하고
    for i in s:
        dict_val = d.get(i, 0) # fro문을 돌면서 d에 value가 없는 경우 추가해주고
        d[i] = dict_val + 1 #있으면 count를 1 증가
    return d #최종 딕셔너리 반환
```

In [8]:

```
s = 'It is a truth universally acknowledged that a single man in possession of a good fortune must be in want of money'
most_frequent(s)
```

```
'n': 9,  
'o': 76,  
'w': 20,  
'd': 38,  
'g': 20,  
'm': 24,  
'p': 7,  
'f': 25,  
'b': 10,  
'.': 16,  
'H': 1,  
',': 11,  
'x': 2,  
' ': 19,  
'M': 10,  
'B': 5,  
'N': 2,  
'P': 1,  
'?': 4,  
';': 3,  
'!': 2
```

In [9]:

```
# 비교를 보다 쉽게 하기 위해 key값으로 쓸 함수를 추가로 정의  
# 그냥 most_frequent(s)도  
def f1(x):  
    return x[1]
```

오만과 편견 챕터1의 영어, 스페인어 글자 수를 세자

In [10]:

```
# english
s = 'It is a truth universally acknowledged that a single man in possession of a good fortune must be in want of money'
s = most_frequent(s)
sorted(s.items(), key=f1, reverse=True)
```

Out[10]:

```
[(' ', 243),
 ('e', 115),
 ('t', 90),
 ('o', 76),
 ('a', 75),
 ('n', 74),
 ('i', 69),
 ('s', 63),
 ('h', 63),
 ('r', 54),
 ('d', 38),
 ('l', 31),
 ('u', 26),
 ('f', 25),
 ('m', 24),
 ('w', 20),
 ('g', 20),
 ('v', 19),
```

In [11]:

```
# spanish
s = 'Es una verdad universalmente reconocida que un hombre soltero en posesión de una buena fortuna debe ser rico'
s = most_frequent(s)
sorted(s.items(), key=f1, reverse=True)
```

Out[11]:

```
[(' ', 225),
 ('e', 142),
 ('a', 103),
 ('n', 85),
 ('o', 84),
 ('s', 72),
 ('r', 64),
 ('i', 57),
 ('u', 54),
 ('d', 51),
 ('t', 44),
 ('l', 42),
 ('c', 31),
 ('m', 23),
 ('q', 19),
 ('p', 19),
 ('.', 15),
 ('h', 13),
```

당연한 결과인 것 같긴 하지만 두 언어 모두 공백이 많다. 또한 e, a와 같은 모음의 빈도가 두드러지게 나타나는 것 같다.

(#4) Ex14.2

In [12]:

```
#14.2
import os
def sed(pattern, repl, a, b):
    try :
        fout = open(a, 'r') #읽기 모드로 open
        ww = open(b, 'w') # 쓰기 모드로 open
        lines = fout.readlines()
        for line in lines: #for문을 돌면서 확인하자
            if pattern in line:
                line = line.replace(pattern, repl) #패턴과 일치하면 repl로 바꾸고
                ww.write(line) #바꾼 문자 작성
            else:
                ww.write(line) #일치하지 않는 경우는 그대로 작성

        #파일닫기
        fout.close()
        ww.close()

    except:
        print ('Something went wrong.') #오류 출력
```

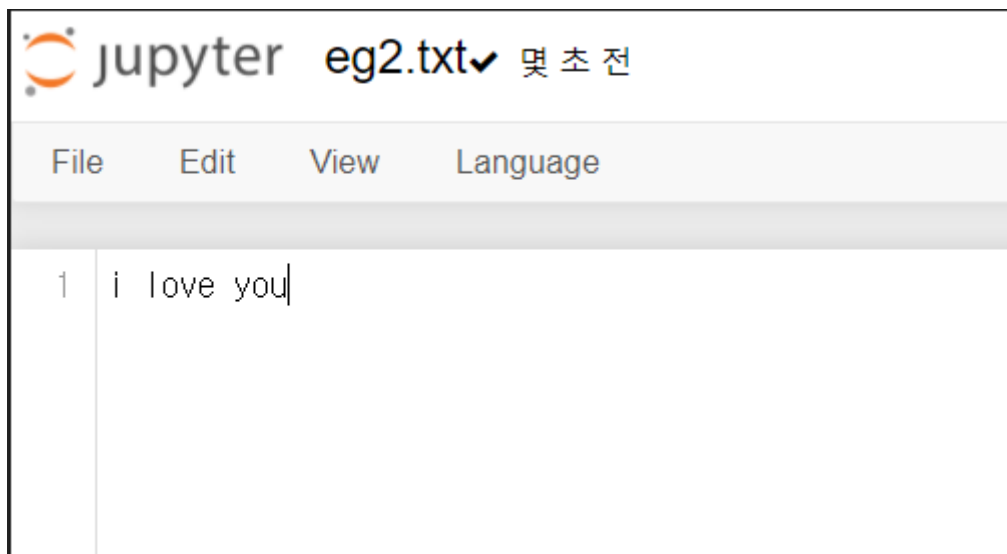
In [13]:

```
sed( 'love', 'like' , 'eg1.txt', 'eg2.txt')
```

In [14]:

```
from IPython.display import Image
Image('C:/Users/82104/Desktop/before.png') #실행전
```

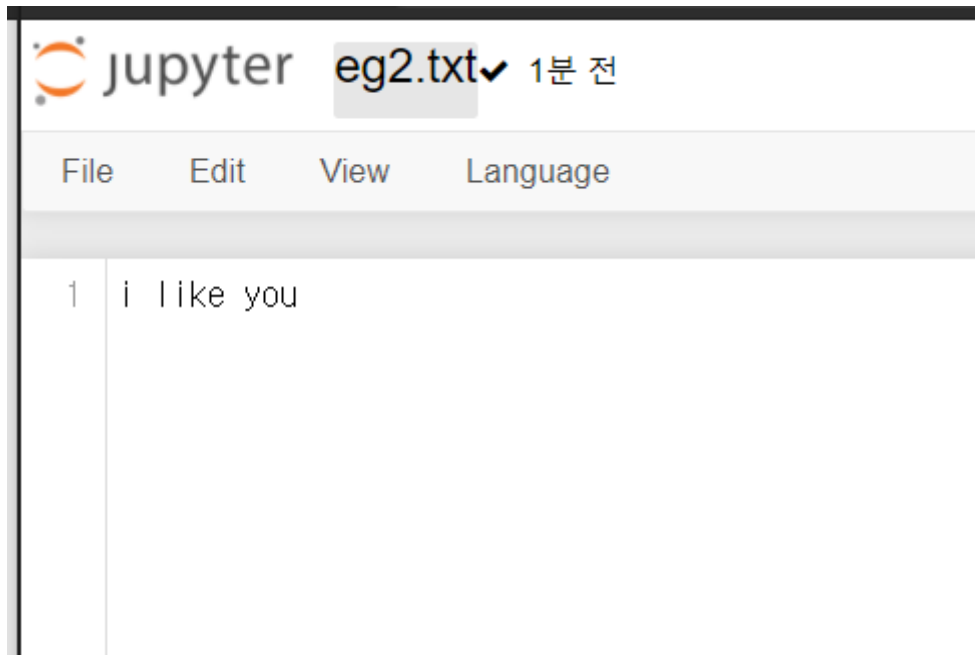
Out[14]:



In [15]:

```
Image('C:/Users/82104/Desktop/after.png') #실행후
```

Out[15]:



(#5)

In [16]:

```
def fun_abs():
    while True:
        try:
            x = float(input("Please enter a number: ")) #입력받아서
            print(abs(x)) #절댓값 출력
            break #탈출
        except ValueError: #입력값이 숫자가 아니면 (ValueError)
            print("Oops! That was no valid number. Try again...") #경고출력
```

In [17]:

```
fun_abs()
```

```
Please enter a number: rfg
Oops! That was no valid number. Try again...
Please enter a number: -----
Oops! That was no valid number. Try again...
Please enter a number: -4.3
4.3
```

