

고급통계프로그래밍 # 5

2017580034 통계학과 이유민

Ex10.6

In [1]:

```
def is_sorted(l):      # list l을 받은 후
    if l == sorted(l): # 오름차순으로 정렬한 리스트와 같은지 확인
        return True    # 같은 경우는 True,
    else:
        return False   # 아니면 False 반환
```

In [2]:

```
is_sorted([1,2,2])
```

Out[2]:

True

In [3]:

```
is_sorted(['b', 'a']) # != ['a','b']
```

Out[3]:

False

Ex10.7

In [4]:

```
def is_anagram(x, y):      # 문자열 x, y를 받아서
    if sorted(x) == sorted(y): # 각각을 정렬한 리스트가 동일한지 확인
        return True          # 동일하다면 : 애너그램 > True를,
    else:
        return False         # 그렇지 않으면 : 애너그램X > False 반환
```

In [5]:

```
is_anagram('parrot', 'raptor')
```

Out[5]:

True

In [6]:

```
is_anagram('parrot', 'carrot')
```

Out[6]:

False

Ex11.2

In [7]:

```
# original function
def histogram(s):
    d = dict()
    for c in s:
        if c not in d:
            d[c] = 1
        else:
            d[c] += 1
    return d
```

In [8]:

```
histogram('carrot')
```

Out[8]:

```
{'c': 1, 'a': 1, 'r': 2, 'o': 1, 't': 1}
```

In [9]:

```
# modified function
# 조건 1)get 사용, 2)if문 제거

def histogram(s):
    d = dict()    # 딕셔너리 생성
    for c in s:   # for문을 돌면서
        d[c] = d.get(c, 0) + 1 # 각 키를 발견할 때마다 +1 한 결과를 저장해서
    return d      # 반환
```

In [10]:

```
histogram('carrot')
```

Out[10]:

```
{'c': 1, 'a': 1, 'r': 2, 'o': 1, 't': 1}
```

Ex11.3

In [11]:

```
# original function
# 딕셔너리의 키를 탐색하고, 각 키와 대응하는 값 인쇄
def print_hist(h):
    for c in h:
        print(c, h[c])
```

In [12]:

```
# modified function
# 키와 값을 알파벳순으로 인쇄

def print_hist(h):
    new = sorted(h.items()) # items()로 키-값 쌍을 모두 가져온 후 정렬
    for i in new:
        print(i) # for문을 돌면서 정렬한 순서(알파벳순)대로 반환
```

In [13]:

```
print_hist(histogram('parrot'))
```

```
('a', 1)
('o', 1)
('p', 1)
('r', 2)
('t', 1)
```

Ex11.4

In [14]:

```
def reverse_lookup(d,v):
    ans = []
    for i in d:
        if d[i] == v:
            ans.append(i)
    return ans # ans 반환, ans에 append된 게 없으면 빈리스트 그대로 반환
```

In [15]:

```
reverse_lookup(histogram('parrot'), 1)
```

Out[15]:

```
['p', 'a', 'o', 't']
```

In [16]:

```
reverse_lookup(histogram('parrot'), 2)
```

Out[16]:

```
['r']
```

In [17]:

```
reverse_lookup(histogram('parrot'), 3)
```

Out[17]:

```
[]
```