

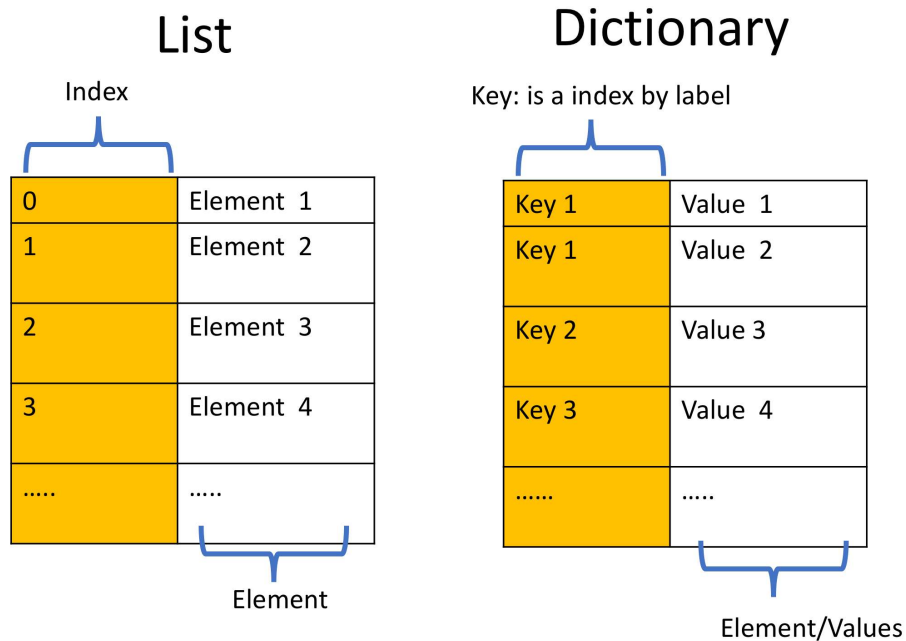
9. 딕셔너리 (Dictionary) 자료형

dictionary 는 key-value 쌍으로 이루어진 data type 이다.

dictionary 는 { } 내에 key: value 로 표시한다.

list, tuple 의 element 들은 indexing 에 의해 접근하지만 , dictionary 는 key 값으로 접근한다.

dictionary = {key: value, key: value...}



9.1 Dictionary basics

dictionary 생성

In [1]:

```
1 dict = {'a': 1, 'b': 2, 'c': 3}
```

In [2]:

```
1 type(dict)
```

Out[2]:

dict

dictionary 값(value) 접근

In [3]:

```
1 dict['b'] + dict['c']
```

Out[3]:

5

In [4]:

```
1 my_info = {'name': 'Kim', 'age': 25, 'height': 175, 'home': 'Seoul'}
2
3 if my_info['age'] < 30 and my_info['height'] > 170:
4     print("My name is " + my_info['name'] + ' and I live in ' + my_info['home'] + '.')
```

My name is Kim and I live in Seoul.

Dictionary 값 update

In [5]:

```
1 my_info['name'] = 'Jane'
2 my_info['home'] = 'New York'
```

In [6]:

```
1 if my_info['age'] < 30 and my_info['height'] > 170:
2     print("My name is " + my_info['name'] + ' and I live in ' + my_info['home'] + '.')
```

My name is Jane and I live in New York.

dictionary 와 for-loop

In [7]:

```
1 for info in my_info:
2     print(info)
```

name
age
height
home

In [8]:

```
1 for key in my_info.keys():
2     print(key)
```

name
age
height
home

In [9]:

```
1 for value in my_info.values():
2     print(value)
```

Jane
25
175
New York

In [10]:

```
1 for key, value in my_info.items():
2     print(key, value)
```

name Jane
age 25
height 175
home New York

In [11]:

```
1 for key in my_info.keys():
2     print(my_info[key])
```

Jane
25
175
New York

In [12]:

```
1 for key in my_info.keys():
2     print(key, my_info[key])
```

name Jane
age 25
height 175
home New York

In [13]:

```
1 for key, value in my_info.items():
2     print(key, '\t', value)
```

name Jane
age 25
height 175
home New York

dictionary 정렬

In [14]:

```
1 dict = {'c': 1, 'a': 2, 'b': 3}
2 sorted(dict)
```

Out[14]:

['a', 'b', 'c']

In [15]:

```
1 for key, value in dict.items():
2     print(key, '\t', value)
```

```
c    1
a    2
b    3
```

In [16]:

```
1 for key in sorted(dict):
2     print(key, '\t', dict[key])
```

```
a    2
b    3
c    1
```

dictionary 에 key 존재 유무 check

In [17]:

```
1 dict = {'a': 1, 'b': 2, 'c': 3}
2 'a' in dict
```

Out[17]:

True

In [18]:

```
1 'd' in dict
```

Out[18]:

False

dictionary 를 이용한 list 의 중복 element count

In [19]:

```
1 def word_count_dict(word_list):
2     word_count = {}
3     for word in word_list:
4         word = word.lower()
5         if not word in word_count:
6             word_count[word] = 1
7         else:
8             word_count[word] += 1
9     return word_count
```

In [20]:

```
1 word_list = ['test', 'Test', 'apple', 'Apple', 'orange', 'dog', 'caT', 'CAT']
2
3 words_cnt = word_count_dict(word_list)
```

In [21]:

```
1 words_cnt
```

Out[21]:

```
{'test': 2, 'apple': 2, 'orange': 1, 'dog': 1, 'cat': 2}
```

In [22]:

```
1 for k, v in words_cnt.items():
2     print(k, v)
```

```
test 2
apple 2
orange 1
dog 1
cat 2
```

계층적 자료구조 구성

In [23]:

```
1 stock = {
2     'name': '삼성전자',
3     'market': 'KOSPI',
4     'close': [
5         ('2010-08-01', 45000),
6         ('2010-08-02', 47000),
7         ('2010-08-03', 42000),
8     ],
9     'market-cap': 2919097,
10    'PER': 15.17
11 }
```

In [24]:

```
1 stock['market-cap']
```

Out[24]:

2919097

In [25]:

```
1 for c in stock['close']:
2     print(c[0], c[1])
```

2010-08-01 45000

2010-08-02 47000

2010-08-03 42000

In [26]:

```
1 stock['phone'] = '02-3455-2020'
```

In [27]:

```
1 stock
```

Out[27]:

```
{'name': '삼성전자',
 'market': 'KOSPI',
 'close': [('2010-08-01', 45000),
 ('2010-08-02', 47000),
 ('2010-08-03', 42000)],
 'market-cap': 2919097,
 'PER': 15.17,
 'phone': '02-3455-2020'}
```

In [28]:

```
1 stock.keys()
```

Out[28]:

```
dict_keys(['name', 'market', 'close', 'market-cap', 'PER', 'phone'])
```

연습문제

1) 다음의 출력 결과는 ?

```
d = {'a': 2, 'b': 4, 'c': 9}
for x in sorted(d):
    print(d[x], end="")
```

2) 다음의 출력 결과는 ?

```
d = {'a': 2, 'b': 4, 'c': 9}
for x in sorted(d.values()):
    print(x, end="")
```

3) 다음의 출력 결과는 ?

```
d = {'a': 21, 'b': 4, 'c': 9}
for x in sorted(d.items()):
    print(x, end="")
```

4) dictionary 를 이용하여 자신의 정보를 구조화 하여 작성

```
ex) {'name': '오영제',
     '성별': '남성',
     '거주지': {'도시': '서울', '동': '충무로', '주소': '123-45'},
     '전화번호': 1012345678,
     '신장': 175.17}
```

5) 두개의 주사위를 던져서 두 주사위의 합이 같은 것끼리 출력하라.

- pseudo-code

```
d = {}
```

Loop with 주사위1 from 1 to 6

Loop with 주사위2 from 1 to 6

newTuple = (주사위1, 주사위2)

added = 주사위1 + 주사위2

if added 가 d 에 없으면 empty list 를 d 에 추가

d 의 기존 list 에 append