

1. 딕셔너리 만들기

사전형
(dictionary)

```
dic = { 키:값, 키:값, 키:값 }
```

- 딕셔너리의 기본 형태
- 리스트, 튜플, 세트와 달리 키(key)와 값으로 이루어짐

1.1 딕셔너리 만들기

```
dic={}
```

: 중괄호를 사용하여
빈 딕셔너리를 선언

- 중괄호를 사용하거나 dict() 함수를 사용해서 빈 딕셔너리를 만들 수 있습니다.

코드 1-1 빈 딕셔너리 만들기

```
dic1 = {}
dic2 = dict()

print(type(dic1), dic1)
print(type(dic2), dic2)
```

```
<class 'dict'> {}
<class 'dict'> {}
```

- kor은 키, 80은 값과 같이 키:값의 형태로 만들 수 있습니다.

코드 1-1-1 딕셔너리 만들기

```
dict = {'kor':80, 'eng':90, 'mat':100}
```

```
print(dict)
```

```
{'kor': 80, 'eng': 90, 'mat': 100}
```

2. 딕셔너리 특징

2.1 값 조회하기

<키를 이용해서 값 조회하기>

```
dic={'kor':80,'eng':90,'mat':77}  
print(dic['kor'])
```

키

80

. index(순서)를 통해 조회하는
다른 구조들과 다름

. 키를 넣으면 값을 얻음

코드 2-1 값 조회하기

```
dict = {'kor':80, 'eng':90, 'mat':77}
```

```
print(dict['kor'])
```

```
80
```

2.2 키와 값을 추가 / 값을 수정

<딕셔너리에 새로운 값 추가>

```
dic={'kor':80,'eng':90,'mat':77}  
dic['tot']=247  
print(dic)
```

→ 새로운 키이면 키와 값을 추가

```
{'tot':247,'kor':80,'eng':90,'mat':77}
```

→ 딕셔너리는 값을 순서대로 저장하지 않는다

<키를 이용해서 값 변경하기>

```
dic={'kor':80,'eng':90,'mat':77}  
print(dic['kor']=100)
```

→ 존재하는 키이면 값을 변경

```
{'kor':100,'eng':90,'mat':77}
```

코드 2-2 새로운 키와 값을 추가하기

```
dict = {'kor':80, 'eng':90, 'mat':77}  
dict['sci']=92 # 새로운 키 sci와 값 92를 추가합니다.  
  
print(dict)
```

```
{'kor': 80, 'eng': 90, 'mat': 77, 'sci': 92}
```

코드 2-2-1 값 수정하기

```
dict = {'kor':80, 'eng':90, 'mat':77}  
dict['kor']=85 # kor은 기존에 있던 키입니다. 따라서 kor에 값을 85로 변경.  
  
print(dict)
```

```
{'kor': 85, 'eng': 90, 'mat': 77}
```

3. 딕셔너리 활용

3.1 딕셔너리 항목 삭제 / 존재 유무 확인하기

<딕셔너리 항목 삭제하기> : 선택해서 삭제할 때는 키 이름 입력, 모든 항목 삭제할 때는 clear

```
dic={'kor':80,'eng':90,'mat':77}
del dic['mat']
print(dic)
```

{'kor':80,'eng':90}

```
dic={'kor':80,'eng':90,'mat':77}
dic.clear()
print(dic)
```

{}

<딕셔너리 항목 존재 유무 확인하기>

```
dic={'kor':80,'eng':90,'mat':77}
print('eng' in dic)
```

True

```
dic={'kor':80,'eng':90,'mat':77}
print('tot' in dic)
```

False

코드 3-1 항목 삭제하기

```
dict = {'kor':80, 'eng':90, 'mat':77}
del dict['mat']

print(dict)
```

{'kor': 80, 'eng': 90}

코드 3-1-1 항목 전체 삭제하기

```
dict = {'kor':80, 'eng':90, 'mat':77}
dict.clear()

print(dict)
```

{}

코드 3-1-2 키의 존재 확인하기

```
dict = {'kor':80, 'eng':90, 'mat':77}

print('kor' in dict)
```

True

3.2 딕셔너리 키, 값 얻기

<모든 키 얻기>

```
dic={'kor':80,'eng':90,'mat':77}
print(dic.keys())

dict_keys['kor', 'eng', 'mat']
```

```
dic={'kor':80,'eng':90,'mat':77}
print(list(dic.keys()))

['kor', 'eng', 'mat']
```

<모든 값 얻기>

```
dic={'kor':80,'eng':90,'mat':77}
print(list(dic.values()))

[80,90,77]
```

<모든 순서쌍 얻기>

```
dic={'kor':80,'eng':90,'mat':77}
print(list(dic.items()))

[('kor',80),('eng',90),('mat',77)]
```

코드 3-2 딕셔너리의 모든 키 얻기

```
dict = {'kor':80, 'eng':90, 'mat':77}

print(dict.keys())
print(list(dict.keys()))          # 가지온 키들을 리스트로 만들어 줍니다.
```

```
dict_keys(['kor', 'eng', 'mat'])
['kor', 'eng', 'mat']
```

코드 3-2-1 딕셔너리의 모든 값 얻기

```
dict = {'kor':80, 'eng':90, 'mat':77}

print(dict.values())
print(list(dict.values()))        # 가지온 값들을 리스트로 만들어 줍니다.
```

```
dict_values([80, 90, 77])
[80, 90, 77]
```

코드 3-2-2 딕셔너리의 모든 순서쌍 얻기

```
dict = {'kor':80, 'eng':90, 'mat':77}

print(dict.items())
print(list(dict.items()))         # 가지온 키와 값의 순서쌍을 리스트로 만들어 줍니다.
```

```
dict_items([('kor', 80), ('eng', 90), ('mat', 77)])
[('kor', 80), ('eng', 90), ('mat', 77)]
```

3.3 딕셔너리를 list / tuple / set으로 변형

- 딕셔너리를 list(), tuple(), set() 함수에 넣으면 키만 list, tuple, set로 변형된다.

코드 3-3 딕셔너리를 list / tuple / set 으로 변형

```
dict = {'kor':80, 'eng':90, 'mat':77}
li = list(dict)
tu = tuple(dict)
s = set(dict)
```

```
print(li)
print(tu)
print(s)
```

```
['kor', 'eng', 'mat']
('kor', 'eng', 'mat')
{'kor', 'eng', 'mat'}
```

3.4 list, tuple 자료형을 딕셔너리로 변형

- 리스트와 튜플의 항목이 짝을 이룰 때 가능합니다.

코드 3-4 리스트, 튜플을 딕셔너리로 변형

```
li = ['ab','cd','ef']
tu = ('ab','cd','ef')
dic1 = dict(li)
dic2 = dict(tu)
```

```
print(dic1)
print(dic2)
```

```
{'a': 'b', 'c': 'd', 'e': 'f'}
{'a': 'b', 'c': 'd', 'e': 'f'}
```

코드 3-4-1 리스트, 튜플을 딕셔너리로 변형

```
li = [['a',1],['b',2],['c',3]]
dic = dict(li)
```

```
print(dic)
```

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