

# dictionary

- 딕셔너리(dictionary)
  - 의미 있는 두 개의 값이 쌍으로 묶인 구조
  - 전화번호부의 성명, 전화번호를 쌍으로 연결한 구조
  - 키(key)와 값(value) 형태로 데이터를 저장하는 자료구조
  - Hash table, Associative Array 구조
- 딕셔너리의 선언
  - 중괄호 {}를 사용하여 키와 값의 쌍으로 구성

딕셔너리 변수 = {키 1:값 1, 키 2:값 2, 키 3:값 3, ...}

Key	Value
키1	값1
키2	값2
키3	값3

# dictionary

- 딕셔너리 생성, 접근, 추가

```
#dictionary 선언 {키:값1, ...}
tel = {'홍길동': '233-1010', '이순신': '111-2323', '황진이': '565-4848'}
print('tel=', tel)

#키로 접근하여 값을 추출
print("tel['홍길동']=" , tel['홍길동'])

#새로운 키로 값을 추가
tel['이몽룡'] = '344-9191'

print('tel=', tel)
```

```
tel= {'홍길동': '233-1010', '이순신': '111-2323', '황진이': '565-4848'}
tel['홍길동']= 233-1010
tel= {'홍길동': '233-1010', '이순신': '111-2323', '황진이': '565-4848', '이몽룡': '344-9191'}
```

Key	Value
홍길동	233-1010
이순신	111-2323
황진이	565-4848

Key	Value
홍길동	233-1010
이순신	111-2323
황진이	565-4848
이몽룡	344-9191

# dictionary

## 딕셔너리 생성, 변경, 추가 예

```
#개인정보 딕셔너리 생성 예
info = {'성명': '홍길동', '전화번호': '233-1010', '학과': '컴공'}
print('info=', info)

#새로운 항목 추가
info['나이'] = 22          #숫자형

#새로운 항목 추가
info['점수'] = [100, 80, 70] #리스트형

#항목 변경
info['학과'] = '데이터과학'

print('info=', info)
```

```
info= {'성명': '홍길동', '전화번호': '233-1010', '학과': '컴공'}
info= {'성명': '홍길동', '전화번호': '233-1010', '학과': '데이터과학', '나이': 22, '점수': [100, 80, 70]}
```

Key	Value
성명	홍길동
전화번호	233-1010
학과	컴공

Key	Value
성명	홍길동
전화번호	233-1010
학과	데이터과학
나이	22
점수	[100,80,70]

# dictionary

- Key , value 추출
  - items( ) : 키-값 쌍을 튜블의 리스트로 반환
  - keys( ) : 키들만 추출하여 리스트로 반환
  - values( ) : 값들만 추출하여 리스트로 반환

```
#key, value 추출
items = info.items()    #키-값 쌍을 튜블의 리스트로 반환
keys = info.keys()      #키들만 추출하여 리스트로 반환
values = info.values()  #값들만 추출하여 리스트로 반환

print('items = ', items)
print('keys = ', keys)
print('values = ', values)

for i in items:    #items리스트를 반복하여 키-값 튜블을 출력
    print(i)
```

```
items = dict_items([('성명', '홍길동'), ('전화번호', '233-1010'), ('학과', '데이터과학'), ('나이', 22), ('점수', [100, 80, 70])])
keys = dict_keys(['성명', '전화번호', '학과', '나이', '점수'])
values = dict_values(['홍길동', '233-1010', '데이터과학', 22, [100, 80, 70]])
('성명', '홍길동')
('전화번호', '233-1010')
('학과', '데이터과학')
('나이', 22)
('점수', [100, 80, 70])
```

# dictionary

- Key , value 추출

```
#key, value 추출
print('* items *')
for k, v in items:    #items의 키와 값을 추출하여 출력
    print(k, v)

print('* keys *')
for k in keys:
    print(k, info[k])    #key로 value를 접근
```

```
* items *
성명 홍길동
전화번호 233-1010
학과 데이터과학
나이 22
점수 [100, 80, 70]

* keys *
성명 홍길동
전화번호 233-1010
학과 데이터과학
나이 22
점수 [100, 80, 70]
```

# QUIZ!

## Report 7. 영한단어장 생성 프로그램

딕셔너리 구조를 이용하여 영한 단어장을 만들고 반복 구조로 단어추가, 단어검색, 단어장출력을 처리하는 프로그램 작성

### <출력 예시>

```
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 1
영어단어 입력 : sun
한글단어 입력 : 태양
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 1
영어단어 입력 : sky
한글단어 입력 : 하늘
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 2
검색할 영어단어 입력 : moon
검색할 영어단어는 등록되어 있지 않습니다.
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 2
검색할 영어단어 입력 : sun
검색된 한글단어 : 태양
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 3
** 영한 단어장 **
영어 한글
-----
sun 태양
sky 하늘
처리할 번호를 입력하세요. (0.종료 1.단어추가 2.단어검색 3.단어장출력) ? 0
종료합니다.
```

- 학습활동 결과는 e-루리에 제출바랍니다.
- 제목 : Report7. 영한단어장 생성 프로그램
- 제출내용 : Report7\_words \_성명. ipynb 파일을 제출
- 제출기한 : 2021년 5월 11일 오후 11:58