

# 딕셔너리의 이해와 활용

II주차\_02\_01

한 동 대 학 교  
김경미 교수

# 학습목표

2

- ▶ 딕셔너리 이해하기
- ▶ 딕셔너리 정의하기
- ▶ 딕셔너리 연산 연습하기

# 사전형(Dictionary)이란?

- ▶ 사전은 리스트와 비슷하지만
  - ▶ 리스트에서는 index 값이 정수여야 함
  - ▶ 사전에서는 모든 종류의 데이터 형을 index로 사용 가능(user-defined indexes)
  - ▶ 정의할 때, '{'와 '}'를 사용

# 사전형 정의하기

## ▶ 정의하기

```
>>> days = {'Sun':'Sunday', 'Mon':'Monday', 'Tue':'Tuesday',  
            'Wed':'Wednesday', 'Thu':'Thursday', 'Fri':'Friday', 'Sat':'Saturday'}
```

## ▶ 첫번째 기술한 아이템이 찾을 때 사용하는 키(index)가 된다

```
>>> days['Sun']
```

```
'Sunday'
```

```
>>> days['Fri']
```

```
'Friday'
```

## ▶ Key : Value => Key로 Value를 찾아낼 수 있음

# 사전형 생성 후 자료 추가

- ▶ 비어있는 사전 생성

```
>>> d = { }
```

- ▶ 함수 dict는 항목(item)없는 새로운 사전을 생성한다

- ▶ 생성 후, 다음과 같이 아이템을 한 개씩 추가 한다

```
# create dictionary
>>> eng2sp = dict()
>>> print(eng2sp)
{}
>>> eng2sp['one'] = 'uno'
>>> print(eng2sp)
{'one': 'uno'}
```

# 사전형 연산

6

operator	Description
len()	사전형 변수에 저장된 아이템의 개수를 알려 준다
k in dictionary_name	k 가 해당 사전형에 존재하면 True, 존재하지 않으면 False
Dictionary_name[k]	k 가 해당 사전형에 존재하면, 해당 아이템 값을 출력한다
Dictionary_name[k] = "item 값"	k 가 해당 사전형에 존재하지 않으면, 해당 아이템 index, 값이 추가되어 저장된다
Dictionary_name.pop(k)	k 가 해당 사전형에 존재하면, 삭제한다

# 사전형 연산, 개수와 아이템 확인 I

7

## ▶ 국가명과 국제전화 코드를 사전형으로 정의

```
>>> country_code = {1:"미국", 20:"이집트", 30:"그리스", 39:"이태리", 81:"일본", 82:"한국"}  
>>> len(country_code)  
6
```

```
>>> country_code[30]  
"그리스"
```

```
>>> country_code[82]  
"한국"
```

# 사전형 연산, 자료 추가와 삭제 I

```
>>> 82 in country_code  
True
```

```
>>> 60 in country_code  
False
```

```
>>> country_code[60] = "말레시아"           # append an item  
>>> 60 in country_code  
True
```

```
>>> country_code.pop(81)                     # delete an item  
'일본'  
>>> 81 in country_code  
False
```



# 사전형 연산, 자료 읽기 I

9

# 사전형 city 값을 읽어내기

```
>>> city = {"New York City":8175133, "Los Angeles": 3792621, "Washington":632323,
"Chicago": 2695598, "Toronto":2615060, "Montreal":11854442, "Ottawa":883391,
"Boston":62600}
```

```
>>> city["Toronto"]
2615060
```

```
>>> city["Boston"]
62600
```

# 사전형 food 아이템 추가하기

```
>>> food = {"ham": "yes", "egg": "yes", "spam": "no" }
```

```
>>> food
{'egg': 'yes', 'ham': 'yes', 'spam': 'no'}
```

```
>>> food["spam"] = "yes"
```

```
>>> food
{'egg': 'yes', 'ham': 'yes', 'spam': 'yes'}
```

# 사전형 연산, 자료 삭제 I

10

```
>>> car = { "brand": "Ford", "model": "Mustang", "year": 1964 }
```

```
>>> car.pop("model")  
Mustang
```

```
>>> car  
{'brand': 'Ford', 'year': 1964}
```

# 사전형 연산, 개수와 아이템 확인 2

11

- ▶ 국가명과 국제전화 코드를 사전형으로 정의

```
>>> sports = {1:"soccer", 2:"baseball", 4:"swimming", 5:"basketball"}
```

```
>>> l = len(sports)
```

```
>>> index = l//2
```

```
>>> sports[index]  
'baseball'
```

# 사전형 연산, 자료 추가와 삭제 2

12

```
>>> 3 in sports  
False
```

```
>>> sports.pop(5)  
'basketball'
```

```
>>> sports[3] = "judo"
```

```
>>> sports  
{1: 'soccer', 2: 'baseball', 4: 'swimming', 3: 'judo'}
```

# 사전형 연산, 자료 읽기 2

13

```
>>> country = {"Korea":1, "USA":2, "Germany":3, "France":4, "Mexico":5}
```

```
>>> country["Korea"]
```

```
1
```

```
>>> country["France"]
```

```
4
```

```
>>> country["Mexico"]=8
```

```
>>> country
```

```
{'Korea': 1, 'USA': 2, 'Germany': 3, 'France': 4, 'Mexico': 8}
```

# 사전형 연산, 자료 삭제 2

14

```
>>> country = {'Korea': 1, 'USA': 2, 'Germany': 3, 'France': 4, 'Mexico': 8}
```

```
>>> country.pop("Germany")
```

```
3
```

```
>>> country.pop("France")
```

```
4
```

```
>>> country
```

```
{'Korea': 1, 'USA': 2, 'Mexico': 8}
```

# 연습문제 I

15

- ▶ 아이템이 없는 사전형 `birthdate`를 정의한다
- ▶ 사용자에게 이름과 생일을 입력 받아서 사전형 `birthdate`에 추가한다
- ▶ 5명의 이름과 생일을 입력 받아서 추가한 후, 추가된 내용을 화면에 출력한다
- ▶ 특정한 한 사람의 이름을 입력 받아서, 생일을 화면에 출력한다

# 연습문제 I 코드

```
birthdate = { }
def add_birth(num):
    for i in range(num):
        input_name = input("이름: ")
        input_birth = int(input("생일: "))
        birthdate[input_name] = input_birth
```

```
add_birth(5)
print(birthdate)
```

```
name=input("생일을 찾고 싶은 사람의 이름을 입력하세요: ")
print(birthdate[name])
```

```
>>>
===== RESTART: E:/1_Works/2017Work/KMooC강의/Exercise Code/9주차_연습.py
이름: kmkim
생일: 0224
이름: sjkang
생일: 0512
이름: kdhong
생일: 1212
이름: ychoi
생일: 0409
이름: hjkim
생일: 1103
{'kmkim': 224, 'sjkang': 512, 'kdhong': 1212, 'ychoi': 409, 'hjkim': 1103}
생일을 찾고 싶은 사람의 이름을 입력하세요: kmkim
224
```



## 연습문제 2

17

- ▶ 연습문제 1에서 생성한 dictionary 사용한다
- ▶ 입력 받은 사람의 자료를 삭제한다
- ▶ 삭제 후 in을 사용하여 삭제 여부를 확인한다

## 연습문제 2 코드

```
birthdate = {}  
def add_birth(num):  
    for i in range(num):  
        input_name = input("이름: ")  
        input_birth = int(input("생일: "))  
        birthdate[input_name] = input_birth
```

```
add_birth(5)  
print(birthdate)
```

```
name=input("생일을 삭제하고 싶은 사람의 이름을 입력하세요: ")  
birthdate.pop(name)  
print(name in birthdate)  
print(birthdate)
```

```
생일을 삭제하고 싶은 사람의 이름을 입력하세요: kmkim  
False  
{'sjkang': 512, 'kdhong': 1212, 'ychoi': 409, 'hjkim': 1103}  
>>>
```

# 강의 요약

19

- ▶ 딕셔너리 이해하기
  - ▶ 중괄호 사용하여 정의
  - ▶ 모든 종류의 데이터형을 index로 사용 가능
- ▶ 딕셔너리 정의하기
- ▶ 딕셔너리 연산 연습하기

# 목표 달성 질문

- ▶ 딕셔너리는 key값을 정의하는가? 리스트와 같이 위치에 따라 값이 정해지는가?
- ▶ 다음 딕셔너리 연산 함수의 기능을 설명하시오
  - ▶ len()
  - ▶ .pop()

# 목표 달성 질문

- ▶ 딕셔너리는 key값을 정의하는가? 리스트와 같이 위치에 따라 값이 정해지는가?
- ▶ 다음 딕셔너리 연산 함수의 기능을 설명하시오
  - ▶ len()
  - ▶ .pop()

# 감사합니다