



파이선으로 배우는 데이터구조

Data Structures Learning with Python

김영훈

한양대학교 ERICA 인공지능학과

Dictionary

▶ 문법 구조 (생성)

키: 값을 참조하기 위한 ID. 문자열도 가능하고 숫자도 가능하다.

```
{  
  키: 값,  
  키: 값,  
  
  ....  
  
  키: 값,  
}
```

값: Dictionary가 저장하는 데이터. 문자열, 숫자, 리스트 등 어떤 객체도 가능하다. 또 다른 Dictionary도 가능하다.

Functions for Dictionary

Method	Description
<u>len()</u>	사전의 키-값의 쌍의 개수를 반환합니다.
<u>clear()</u>	사전에서 모든 요소를 제거합니다.
<u>copy()</u>	딕셔너리의 복사본을 반환합니다.
<u>fromkeys()</u>	지정된 키와 값을 가진 사전을 반환합니다.
<u>get()</u>	지정된 키의 값을 반환합니다.
<u>items()</u>	각 키-값 쌍에 대한 튜플을 포함하는 리스트를 반환합니다.
<u>keys()</u>	딕셔너리의 키가 포함된 리스트를 반환합니다.
<u>pop()</u>	지정된 키를 가진 요소를 제거합니다.
<u>popitem()</u>	마지막으로 삽입된 키-값 쌍을 제거합니다.
<u>setdefault()</u>	지정된 키의 값을 반환합니다. 키가 존재하지 않는 경우: 지정된 값으로 키를 삽입합니다.
<u>update()</u>	지정된 키-값 쌍으로 사전을 업데이트합니다.
<u>values()</u>	딕셔너리에 있는 모든 값의 목록을 반환합니다.

Dictionary 1.

- ▶ 자신의 이름, 나이, 성별에 대한 key-value 쌍을 dictionary로 생성하는 코드를 작성하시오.

```
My_dict = {  
    "name" : "Kim",  
    "age" : 21,  
    "gender" : "M"  
}
```

Dictionary 2.

- ▶ 주어진 dictionary의 길이를 구하는 코드를 작성하시오.
- ▶ 입력 : {"mango" : 2, "orange" : 6}
- ▶ 출력 : Length: 2

```
fruits = {"mango": 2, "orange": 6}
```

```
print("Length:", len(fruits))
```

Dictionary 3.

- ▶ 주어진 dictionary를 빈 dictionary로 만드는 코드를 작성하시오.
- ▶ 입력 : {"mango" : 2, "orange" : 6}
- ▶ 출력 : {}

```
fruits = {"mango": 2, "orange": 6}
```

```
fruits.clear()
```

```
print(fruits)
```

Dictionary 4.

- ▶ Dictionary에 존재하는 모든 key를 출력하는 코드를 작성하시오.
- ▶ 입력 : {'student_id': 1, 'name': 'Jean Castro'}
- ▶ 출력 : ['student_id', 'name']

```
student = {'student_id': 1, 'name': 'Jean Castro'}  
  
print(student.keys())
```

Dictionary 5.

- ▶ Dictionary에 존재하는 모든 value를 출력하는 코드를 작성하시오.
- ▶ 입력 : {'student_id': 1, 'name': 'Jean Castro'}
- ▶ 출력 : [1, 'Jean Castro']

```
student = {'student_id': 1, 'name': 'Jean Castro'}  
  
print(student.values())
```


Dictionary 6.

- ▶ Dictionary에 존재하는 모든 key-value 쌍을 출력하는 코드를 작성하시오.
- ▶ 입력 : {'student_id': 1, 'name': 'Jean Castro'}
- ▶ 출력 : [('student_id', 1), ('name', 'Jean Castro')]

```
student = {'student_id': 1, 'name': 'Jean Castro'}  
  
print(student.items())
```

Dictionary 7.

- ▶ 주어진 dictionary를 반복문을 사용하여 한줄 씩 출력하는 코드를 작성하시오.
- ▶ 입력 : {"x" : 10, "y" : 20, "z" : 30}
- ▶ 출력 : x => 10
y => 20
z => 30

```
d = {'x': 10, 'y': 20, 'z': 30}
```

```
    ?
```

```
    print(key, '=>', value)
```

Dictionary 8.

- ▶ Dictionary의 key를 index를 통해 출력하는 코드를 작성하시오.
- ▶ 입력 : {'physics' : 80, 'math' : 90, 'chemistry' : 86}
- ▶ 출력 : 'physics', 'math', 'chemistry'

```
num = {'physics': 80, 'math': 90, 'chemistry': 86}
```

```
print( ? )
```

Dictionary 9.

- ▶ 주어진 dictionary에 새로운 key-value 쌍을 추가하는 코드를 작성하시오.
- ▶ 입력 : {0 : 10, 1 : 20}
- ▶ 출력 : {0 : 10, 1: 20, 2: 30}

```
d = {0:10, 1:20}
```

```
?
```

```
print(d)
```

Dictionary 10.

- ▶ 주어진 dictionary에서 key-value 쌍 하나를 제거하는 코드를 작성하시오.
- ▶ 입력 : {'a' : 1, 'b' : 2, 'c' : 3, 'd' : 4}
- ▶ 출력 : {'b' : 2, 'c' : 3, 'd' : 4}

```
myDict = {'a':1,'b':2,'c':3,'d':4}
```

```
?
```

```
print(myDict)
```

Dictionary 11.

- ▶ 주어진 dictionary에 특정 key가 존재하는지 아닌지 판별하는 코드를 작성하시오.
- ▶ 입력 : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

```
d = {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
```

```
def is_key_present(x):
```

```
    ?
```

```
    print('Key is present in the dictionary')
```

```
else:
```

```
    print('Key is not present in the dictionary')
```

```
is_key_present(5)
```

```
is_key_present(9)
```

Dictionary 12.

- ▶ 주어진 dictionary에서 중복된 정보를 제거하는 코드를 작성하시오.
- ▶ 입력 : {'id1': {'name': ['Sara'], 'score': [90]},
'id2': {'name': ['David'], 'score': [70]},
'id3': {'name': ['Sara'], 'score': [90]}}

```
student_data = {  
    'id1': {'name': ['Sara'], 'score': [90]},  
    'id2': {'name': ['David'], 'score': [70]},  
    'id3': {'name': ['Sara'], 'score': [90]}  
}  
result = {}  
for key,value in student_data.items():  
    ?  
    result[key] = value  
print(result)
```

Dictionary 13.

- ▶ 두개의 dictionary 중 하나의 dictionary에 다른 dictionary의 key-value 값을 추가하는 코드를 작성하시오.
- ▶ 입력 : {'a': 100, 'b': 200} , {'x': 300, 'y': 200}
- ▶ 출력 : {'a': 100, 'b': 200, 'x': 300, 'y': 200}

```
d1 = {'a': 100, 'b': 200}
```

```
d2 = {'x': 300, 'y': 200}
```

```
d1. ? (d2)
```

```
print(d1)
```


Dictionary 14.

- ▶ 세개의 dictionary를 반복문을 사용하여 새로운 dictionary로 만드는 코드를 작성하시오.
- ▶ 입력 : {1: 10, 2: 20} , {3: 30, 4: 40}, {3: 30, 4: 40}
- ▶ 출력 : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

```
dic1={1:10, 2:20}  
dic2={3:30, 4:40}  
dic3={5:50,6:60}  
dic4 = {}
```

?

```
print(dic4)
```

Dictionary 15.

- ▶ 주어진 dictionary의 모든 값의 합을 구하는 코드를 작성하시오.
- ▶ 입력 : {'data1':100,'data2':-54,'data3':247}
- ▶ 출력 : 293

```
my_dict = {'data1':100,'data2':-54,'data3':247}
```

```
print ?
```

Dictionary 16.

- ▶ 두개의 dictionary에서 같은 key는 합하고 key가 없다면 추가하는 코드를 작성하시오.
- ▶ 입력 : d1 = {'a': 100, 'b': 200, 'c':300}, d2 = {'a': 300, 'b': 200, 'd':400}
- ▶ 출력 : d1 = {'a': 400, 'b': 400, 'c': 300, 'd': 400}

```
d1 = {'a': 100, 'b': 200, 'c':300}
```

```
d2 = {'a': 300, 'b': 200, 'd':400}
```

```
d1keys = d1.keys()
```



?

Dictionary 17.

- ▶ 주어진 dictionary의 모든 값의 곱을 구하는 코드를 작성하시오.
- ▶ 입력 : {'data1':100,'data2':-54,'data3':247}
- ▶ 출력 : -1333800

```
my_dict = {'data1':100,'data2':-54,'data3':247}
```

```
result=1
```

```
?
```

```
print(result)
```

Dictionary 18.

- ▶ 주어진 dictionary에서 값이 20인 key를 리스트의 형태로 출력하는 코드를 작성하시오.
- ▶ 입력 : {'Thor': 20, 'Hulk': 19, 'Thanos': 20}
- ▶ 출력 : ['Thor', 'Thanos']

```
def test(dict, val):  
    result= list(key for ? in ? () if value == val)  
    return result
```

```
students = {'Thor': 20, 'Hulk': 19, 'Thanos': 20}
```

```
print(test(students, 20))
```

Dictionary 19.

- ▶ 주어진 dictionary의 key와 value의 값을 서로 바꾸는 코드를 작성하시오.
- ▶ 입력 : {'Theodore': 10, 'Mathew': 11, 'Roxanne': 9}
- ▶ 출력 : {10 : 'Theodore', 11 : 'Mathew', 9 : 'Roxanne'}

```
def test(students):  
    return { value: key ? }  
  
students = {  
    'Theodore': 10,  
    'Mathew': 11,  
    'Roxanne': 9,  
}  
print(test(students))
```

Dictionary 20.

- ▶ 주어진 dictionary에서 최대, 최소 값을 출력하는 코드를 작성하시오.
- ▶ 입력 : {"m1":78 , "m2":89 , "m3":64 , "m4":35 , "m5":71}
- ▶ 출력 : "Max" = 89, "Min" = 35

```
marks={"m1":78 , "m2":89 , "m3":64 , "m4":35 , "m5":71}
```

```
v = 
```

```
maxi = max(v)
```

```
mini = min(v)
```

```
print("Max =", maxi, "Min =", mini)
```

Dictionary 21.

- ▶ 주어진 dictionary에서 짝수 값을 필터링하는 코드를 작성하시오.
- ▶ 입력 : {'Mary' : [1, 3, 6, 10], 'John' : [1, 7, 12], 'Mark' : [1, 3, 8]}
- ▶ 출력 : {'Mary': [6, 10], 'John': [12], 'Mark': [8]}

```
def test(dictt):  
    result = {key: ?}  
    return result  
  
students = {'Mary' : [1, 3, 6, 10],  
            'John' : [1, 7, 12],  
            'Mark' : [1, 3, 8]}  
  
print(test(students))
```


Dictionary 22.

- ▶ 주어진 dictionary의 값을 다음 출력과 같이 변경하는 코드를 작성하시오.
- ▶ 입력 : {'Math' : [88, 90], 'Physics' : [94, 89], 'Chemistry' : [90, 87]}
- ▶ 출력 : {'Math': [89, 91], 'Physics': [92, 87], 'Chemistry': [90, 87]}

```
def test(dictt):
```

```
    dictt['Math'] = ?
```

```
    dictt['Physics'] = ?
```

```
    return dictt
```

```
score = {'Math' : [88, 90], 'Physics' : [94, 89], 'Chemistry' : [90, 87]}
```

```
print(test(score))
```

Dictionary 23.

- ▶ 주어진 dictionary 값을 오름차순으로 정렬하는 코드를 작성하시오.
- ▶ 입력 : {'n1': [2, 3, 1], 'n2': [5, 1, 2], 'n3': [3, 2, 4]}
- ▶ 출력 : {'n1': [1, 2, 3], 'n2': [1, 2, 5], 'n3': [2, 3, 4]}

```
num = {'n1': [2, 3, 1], 'n2': [5, 1, 2], 'n3': [3, 2, 4]}
```

```
sorted_dict = 
```

```
print(sorted_dict)
```

Dictionary 24.

- ▶ 주어진 dictionary를 key의 알파벳 순서대로 정렬하시오.
- ▶ 입력 : {'banana': 'yellow', 'kiwi': 'green', 'apple': 'red', 'grape': 'purple'}
- ▶ 출력 : {'apple': 'red', 'banana': 'yellow', 'grape': 'purple', 'kiwi': 'green'}

```
fruits_dict = {'banana': 'yellow',  
               'kiwi': 'green',  
               'apple': 'red',  
               'grape': 'purple'}
```

```
result = {}
```

```
    ?
```

```
print(result)
```

Dictionary 25.

- ▶ 주어진 string의 알파벳을 key로 개수를 value로 하는 dictionary를 만드는 코드를 작성하시오.
- ▶ 입력 : 'Hello Python'
- ▶ 출력 : {'H': 1, 'e': 1, 'l': 2, 'o': 2, ' ': 1, 'P': 1, 'y': 1, 't': 1, 'h': 1, 'n': 1}

```
str1 = 'Hello Python'
```

```
my_dict = {}
```

```
?
```

```
print(my_dict)
```

Dictionary 26.

- ▶ String을 value로 가지는 dictionary의 총 알파벳의 개수를 구하는 코드를 작성하시오.
- ▶ 입력 : {1:'Red', 2:'Blue', 3:'Yellow', 4:'Olive'}
- ▶ 출력 : 18 (3 + 4 + 6 + 5)

```
def test(dictt):  
    result = ?  
    return result  
  
color = {1:'Red', 2:'Blue', 3:'Yellow', 4:'Olive'}  
  
print(test(color))
```

Dictionary 27.

- ▶ 키가 170 이상인 사람을 필터링하여 출력하는 코드를 작성하시오.
- ▶ 입력 : {'Kim': 175, 'Park': 180, 'Lee': 165, 'Sim': 190}
- ▶ 출력 : {'Kim': 175, 'Park': 180, 'Sim': 190}

```
heights = {'Kim': 175, 'Park': 180, 'Lee': 165, 'Sim': 190}
```

```
print("heights greater than 170:")
```

```
result = 
```

?

```
print(result)
```

Dictionary 28.

- ▶ 키가 170이상 그리고 몸무게가 70 미만인 사람을 필터링하여 출력하는 코드를 작성하시오.
- ▶ 입력 : {'Kim': (175, 70), 'Park': (180, 65), 'Lee': (165, 68), 'Sim': (190, 66)}
- ▶ 출력 : {'Park': (180, 65), 'Sim': (190, 66)}

```
def filter_data(students):  
    result = ?  
    return result  
  
students = {'Kim': (175, 70), 'Park': (180, 65), 'Lee': (165, 68), 'Sim': (190, 66)}  
  
print("Height >= 170 and Weight < 70kg:")  
  
print(filter_data(students))
```

Dictionary 29.

- ▶ 주어진 dictionary에서 value가 가장 짧은 key를 출력하는 코드를 작성하시오.
- ▶ 입력 : { 'A': [10, 12], 'B': [10, 20, 30, 40], 'C': [20]}
- ▶ 출력 : ['C']

```
def test(dictt):  
    min_value=1  
    result = ?  
    return result  
  
dictt = { 'A': [10, 12], 'B': [10, 20, 30, 40], 'C': [20]}  
  
print(test(dictt))
```


Dictionary 30.

- ▶ 두개의 리스트 중 l1을 key로 l2를 value로 하는 dictionary를 만드는 코드를 작성하시오.
- ▶ 입력 : l1 = ['a', 'b', 'c', 'd', 'e', 'f'], l2 = [1, 2, 3, 4, 5]
- ▶ 출력 : {'a': 1, 'b': 2, 'c': 3, 'd': 4, 'e': 5}

```
def test(keys, values):
```

```
    return ?
```

```
l1 = ['a', 'b', 'c', 'd', 'e', 'f']
```

```
l2 = [1, 2, 3, 4, 5]
```

```
print(test(l1, l2))
```