

# **Fastcampus Data Science Extension SCHOOL**

## **Python**

# Index

- Dictionary
- Set
- Iterations
  - for
  - while
- Dictionary with Iterations
- Iterations with Conditional Statements

## Do it yourself!

사용자가 입력한 요일('월'~'일' or '월요일'~'일요일')을

- 평일일 경우, "평일이네요~~"
- 주말일 경우, "주말입니다^^"
- 잘못 입력한 경우, "요일을 입력하세요"

을 출력하도록 Conditional Statements 와 Conditional Expressions로 각각 구현하세요

# Answer

```
weekday = ["월", "화", "수", "목", "금"]
weekend = ["토", "일"]

user_day = input()
# "월화수"와 같은 오입력 방지
if len(user_day) > 1 and user_day[-2:] != "요일":
    print("한글자만 입력하거나 요일 외의 다른 글자를 쓰지 마라! ")
else:
    user_day = user_day[0]

print(user_day)

if user_day in weekday:
    print("평일입니다. ")
else:
    if user_day in weekend:
        print("주말입니다. 데헷")
    else:
        print("값을 다시 입력해라(Type again)")
```

# **Dictionary, Set**

# Dictionary

- 매핑 탑입의 묶음자료형
- hash table을 이용하여 구현 -> 일정한 속도 -> 키의 수가 많아도 일정한 탐색속도를 유지



->



- hash:
  - 하나의 문자열을 보다 빨리 찾을 수 있도록 주소에 직접 접근할 수 있는 짧은 길이의 값이나 키로 변환

## List, Tuple과의 차이

- index가 존재하지 않는다.
- key와 value로 이루어져있고, key로 value를 얻을 수 있다.

## dictionary의 선언

```
dict1 = {}
```

```
print(dict1)
```

```
dict2 = {'ja':0, 'chuk':1, 'in':2}
```

```
dict3 = dict(ja=0, chuk=1, in=2)
```

```
dict4 = dict([('ja',0), ('chuk',1),('in',2)])
```

**dictionary는 key와 value로 이루어져 있으며, 추가하는 법은 다음과 같습니다.**

```
dict1 = {'name': 'foo bar'}  
print(dict1)
```

```
dict1 = {'korean': 95, 'math': 100, 'science': [80, 70, 90,  
60]}  
print(dict1)
```

```
dict1['english'] = "pass"  
print(dict1)
```

**요소 삭제는 del을 활용합니다.**

```
del dict1['math']  
print(dict1)
```

**key**를 활용해 **value**를 출력하는 법을 알아봅시다.

```
print(dict1['korean'])
```

**key**만 출력하는 법을 알아봅시다.

```
print(dict1.keys())
```

**value**만 출력할땐 이렇게 합니다.

```
print(dict1.values())
```

**key**와 **value**를 함께 출력합니다.

```
print(dict1.items())
```

## **pop을 할 수도 있습니다**

```
dict1.pop('english')
```

## **key, value 모두 pop 할 경우엔**

```
dict1.popitem('science')
```

# 값을 얻으려 했지만..

```
>>> dict1['english']
```

```
KeyError: 'english'
```

## get()

```
>>> dict1.get('english', 0)
```

```
>>> dict1.get('english')
```

## setdefault()

```
>>> dict1.setdefault('english', [])
```

```
>>> dict1.get('english')
```

```
>>> dict1['english']
```

**dictionary의 길이를 구할땐**

```
len(dict1)
```

**dictionary를 업데이트할 땐**

```
dict1.update({'korean':40, 'math':50, 'ethics':60})
```

# hash table

```
>>> hash_dict={}
```

index	key	value
0		
1		
2		
3		
4		
5		
6		
7		

# hash table

```
>>> hash_dict['ja']=0
>>> hash('ja')
287026653433178398
>>> hash('ja')%8
6
```

index	key	value
0		
1		
2		
3		
4		
5		
6	'ja'	0
7		

# hash table

```
>>> hash_dict['chuk']=1
>>> hash('chuk')
-2672060011921522239
>>> hash('chuk')%8
1
```

index	key	value
0		
1	'chuk'	1
2		
3		
4		
5		
6	'ja'	0
7		

# hash table

```
>>> hashed_dict['in']=2
>>> hash('in')
4800002792290664974
>>> hash('in')%8
6 ????????
```

index	key	value
0		
1	'chuk'	1
2		
3		
4		
5		
6	'ja'	0
7		

# hash table

index	key&value
0	
1	[('chuk',1)]
2	
3	
4	
5	
6	[('ja',0),('in',2)]
7	

## dictionary with string format

```
contacts = {'name':'Guido', 'country':'Netherland'}  
'{name} is born in {country}'.format(**contacts)
```

# Small Quiz

A = 'fastcampus'

B = 'python'

A ∪ B

A ∩ B

A - B

A Δ B

# Set

- 수학 집합 연산을 쉽게 하기 위해 만든 자료형
- 순서없음
- 중복없음

# Set

## Set 선언

```
ppap = {'pen', 'apple', 'pineapple', 'pen'}  
print(ppap)
```

```
'apple' in ppap  
'applepen' in ppap
```

```
pineapple = set('pineapple')  
pineapple
```

# Set

```
A = set('fastcampus')
```

```
B = set('python')
```

```
A ∪ B == A | B
```

```
A ∩ B == A & B
```

```
A - B == A - B
```

```
A Δ B == A ^ B
```

# For, while

```
for 변수 in (리스트 or 문자열):  
    실행문1  
    ...
```

```
for i in ["python", "java", "golang"]:  
    print(i)
```

# For, while

```
sum = 0
for i in range(1,11):
    sum += i
    sum = sum + i
    print(sum)
```

# For, while

```
while 조건:  
    실행문1  
    ...
```

```
while name != "foo bar":  
    name = input("What's your name? ")  
    print("Hi, " + name + "So, where is foo bar?")
```

```
while 1:  
    print("Hello world!")
```

# Dictionary with Iterations

## get word list

```
a_words = []
with open('./a_word_list.txt') as f:
    textlines = f.readlines()
    for item in textlines:
        a_words.append(item.replace('\n', ''))
```

## 길이별 단어 수 구하기

```
len_count = {}
for i in map(len, a_words):
    if i in len_count:
        len_count[i] += 1
    else:
        len_count[i] = 1
```

## 길이별 단어 수 구하기 with get

```
len_count = {}
for i in map(len, a_words):
    len_count[i] = len_count.get(i, 0) + 1
```

## 길이별 단어 분류하기

```
a_word_by_len = {}
for word in a_words:
    a_word_by_len.setdefault(len(word), []).append(word)
```

# Iterations with Conditional Statements

# Fizzbuzz

1부터 100까지 반복하면서,

3의 배수 = "Fizz"

5의 배수 = "Buzz"

15의 배수 = "FizzBuzz"

나머지 = 그 숫자

# Fizzbuzz

```
num = eval(input("type the number: "))

for i in range(1, num + 1):
    if i % 15 == 0:
        print("fizzbuzz")
    elif i % 3 == 0:
        print("fizz")
    elif i % 5 == 0:
        print("buzz")
    else:
        print(i)
```

# Refactoring numguess

```
import random

answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")

while True:
    guess = eval(input("Hi "+ username + ", guess the number"))

    if guess == answer:
        print("Correct! The answer was ", str(answer))
        break
    else:
        print("That's not what I wanted!! Try again!!")
```

# give a hint!!

```
import random

answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")

while True:
    guess = eval(input("Hi, "+ username + "guess the number: "))

    if guess == answer:
        print("Correct! The answer was ", str(answer))
        break
    elif guess > answer:
        print("Too high!! Try again!!")
    elif guess < answer:
        print("Too Low!! Try again!!")
```

# limit trial

```
import random

answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")
trial = 5
while trial:
    guess = eval(input("Hi, "+ username + ". guess the number: "))

    if guess == answer:
        print("Correct! The answer was ", str(answer))
        break
    elif guess > answer:
        trial -= 1
        print("Too high!! Try again!!(%d times left)" % (trial))
    elif guess < answer:
        trial -= 1
        print("Too Low!! Try again!!(%d times left)" % (trial))

if trial == 0:
    print("You are Wrong! The answer was ", str(answer))
```

# Monty Hall Problem

# MONTY HALL PROBLEM

