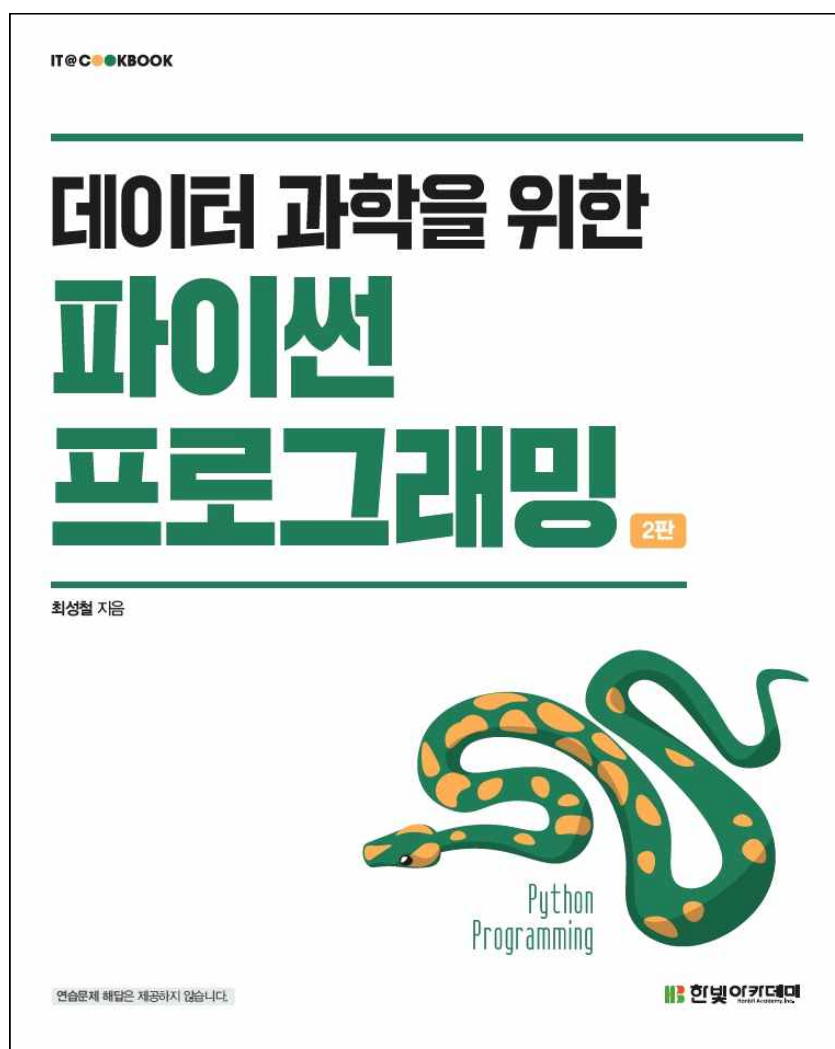


# [데이터 과학을 위한 파이썬 프로그래밍(2판)]

## 추가 연습문제

본 자료의 저작권은 저자 최성철과 한빛아카데미(주)에 있습니다. 이 자료는 강의 보조자료로 제공되는 것으로, 학생들에게 배포되어서는 안 됩니다.



## Chapter 01 프로그래밍 언어와 파이썬

1. 파이썬 인터프리터 언어는 어떤 OS에서든 작동 가능하다.

☐ 예 ☐ 아니오

\* 정답 : 예

2. 파이썬은 C나 자바에 비해 속도가 느리다.

☐ 예 ☐ 아니오

\* 정답 : 예

3. 파이썬 언어의 가장 큰 장점은 기계를 직접 다룰 수 있다는 점이다.

☐ 예 ☐ 아니오

\* 정답 : 아니오

4. 파이썬은 어셈블러와 같은 기계어 변환 과정이 필요 없다.

☐ 예 ☐ 아니오

\* 정답 : 아니오

5. 파이썬은 프로그램 작동 시 별도의 번역 과정이나 소스코드의 해석 없이 CPU에 직접 전달하여 처리 가능한 인터프리터 언어이다.

☐ 예 ☐ 아니오

\* 정답 : 아니오(인터프리터 언어이나 소스코드의 해석이나 CPU에 전달 등의 과정을 거칩니다.)

6. 파이썬의 특징으로 틀린 것은?

- ① 플랫폼에 독립적인 언어이다.
- ② 해당 프로그램이 해결해야 할 문제의 구성요소를 요소별로 정의하고, 각 요소의 기능과 정보를 정의하여 요소들을 결합한 후, 프로그램을 작성하는 방식이다.
- ③ 코드 작성 시 실행 순서를 중심으로 순차적으로 작성한다.
- ④ 실행 시점에서 각 프로그램 변수의 타입을 결정하는 언어이다.
- ⑤ 소스코드 자체가 바로 실행되는 특징이 있는 언어이다.

\* 정답 : ③

7. 파이썬 개발 환경을 결정하는 요인이 아닌 것을 모두 고르면?

- ① 운영체제    ② 웹 브라우저    ③ 파이썬 인터프리터    ④ 코드 편집기    ⑤ 메모장

\* 정답 : ② ⑤

## Chapter 02. 변수와 자료형

1. 다음 코드의 실행 결과를 쓰시오.

```
>>> a = 777
>>> b = 777
>>> print(a == b, a is b)
```

\* 정답 : True False

2. 빈칸에 들어갈 각각의 코드 실행 결과를 쓰시오.

```
>>> a = 3.5
>>> b = int(3.5)
>>> print(a**((a // b) * 2))
(가)
>>> print(((a - b) * a) // b)
(나)
>>> b = (((a - b) * a) % b)
>>> print(b)
(다)
>>> print((a * 4) % (b * 4))
(라)
```

\* 정답 : (가) 12.25    (나) 0.0    (다) 1.75    (라) 0.0

3. 다음 변수 a의 자료형은?

```
a = "True"
```

- |       |       |       |
|-------|-------|-------|
| ① 소수형 | ② 문자형 | ③ 실수형 |
| ④ 불린형 | ⑤ 정수형 |       |

\* 정답 : ②

4. 다음과 같은 코드 작성 시, 실행 결과로 알맞게 짝지어진 것은?

```
>>> a = 10.6
>>> b = 10.5
>>> print(a * b)
[ ]
>>> type(a + b)
[ ]
```

- ① 111.3, <class 'int'>      ② 111.3, <class 'str'>      ③ 111.3, <class 'float'>  
④ 105.0, <class 'int'>      ⑤ 105.0, <class 'float'>

\* 정답 : ③

5. a = "3.5", b = "1.5"일 때, print(a + b)의 실행 결과는?

- ① 5      ② 3.51.5      ③ a + b      ④ ab      ⑤ 2

\* 정답 : ②

6. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
>>> a = '3'
>>> b = float(a)
>>> print(b ** int(a))
```

- ① TypeError      ② '27.0'      ③ 27.0  
④ 27      ⑤ '27'

\* 정답 : ③

7. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
>>> a = '20'
>>> b = '4'
>>> print(type(float(a / b)))
```

- ① <class 'int'>      ② <class 'str'>      ③ <class 'float'>  
④ 4. 3.333333333      ⑤ TypeError

\* 정답 : ⑤

8. 다음 코드의 실행 결과가 나오도록 빈 칸에 코드를 추가하시오.(단, 변수를 삭제하면 안 됨.)

```
>>> a = "Gachon"
>>> b = "CS"
>>> c = 200
```

```
>>> print(a, b, c)
```

```
Gachon CS 50
```

\* 정답 : `c = "50"` 또는 `c = str(50)`

**9. 다음 코드의 예상되는 실행 결과를 쓰시오.**

(가) `print("1.0" * 5)`

(나) `print("1.0" + 2)`

(다) `print("Hanbit" + "Python")`

(라) `print("3.5" + "0.5")`

\* 정답 : (가) 1.01.01.01.01.0    (나) 에러    (다) HanbitPython    (라) 3.50.5

## Chapter 03. 화면 입출력과 리스트 다루기

1. 다음 코드의 실행 결과를 쓰시오.

```
a = [0, 1, 2, 3, 4]
print(a[:3], a[:-3])
```

\* 정답 : [0, 1, 2] [0, 1]

2. 다음 코드의 실행 결과를 쓰시오.

```
a = [0, 1, 2, 3, 4]
print(a[::-1])
```

\* 정답 : [4, 3, 2, 1, 0]

3. 다음 코드의 실행 결과를 쓰시오.

```
num = [1, 2, 3, 4]
print(num * 2)
```

\* 정답 : [1, 2, 3, 4, 1, 2, 3, 4]

4. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
list_a = ['Hankook', 'University', 'is', 'an', 'academic', 'institute', 'located', 'in', 'South Korea']
list_b = [ ]
for i in range(len(list_a)):
    if i % 2 != 1:
        list_b.append(list_a[i])
print(list_b)
```

- ① None
- ② Error
- ③ ['Hankook', 'is', 'academic', 'located', 'South Korea']
- ④ ['University', 'an', 'institute', 'in']
- ⑤ ['Hankook', 'University', 'is', 'an', 'academic', 'institute', 'located', 'in', 'South Korea']

\* 정답 : ③

5. 다음 코드를 실행한 후, 2018과 "2018"을 각각 입력했을 경우 알맞은 실행 결과끼리 묶인 것은?

```
admission_year = input("입학 연도를 입력하세요: ")
print(type(admission_year))
```

- ① <class 'str'>, <class 'float'>
- ② <class 'int'>, <class 'str'>
- ③ <class 'str'>, <class 'str'>
- ④ <class 'int'>, <class 'int'>
- ⑤ <class 'float'>, <class 'int'>

\* 정답 : ③

6. 다음 코드의 실행 결과를 쓰시오.

```
country = ["Korea", "Japan", "China"]
capital = ["Seoul", "Tokyo", "Beijing"]
index = [1, 2, 3]
country.append(capital)
country[3][1] = index[1:]
print(country)
```

\* 정답 : ['Korea', 'Japan', 'China', ['Seoul', [2, 3], 'Beijing']]

7. 다음 코드의 실행 결과가 나오도록 빈칸에 알맞은 코드를 쓰시오.

```
>>> week1 = ["Mon", "Tue", "Wed"]
>>> week2 = ["Thu", "Fri", "Sat", "Sun"]
>>> week3 = 
>>> print(week2[:len(week3) + 1])
['Thu', 'Fri']
```

\* 정답 : [week1 + week2]

8. 다음 코드를 실행하면 다음과 같은 결과가 나온다. 그 이유에 대해 서술하시오.

```
>>> a = 1
>>> b = 1
>>> a is b
True
>>> a = 300
>>> b = 300
>>> a is b
False
```

\* 정답 : 'is'는 주소를 비교하기 위한 명령어로, 파이썬은 처음 실행할 때 -5부터 256까지를 메모리 주소에 넣어서 관리하고, 그 주소를 참조하여 데이터를 비교하기 때문이다.

9. 다음과 같이 코드를 작성했을 때 예측되는 실행 결과를 쓰고, 이러한 결과가 나오는 이유에 대해 서술하시오.

```
>>> a = [5, 4, 3, 2, 1]
>>> b = a
>>> c = [5, 4, 3, 2, 1]
>>> a is b
```

```
>>> a is c
```

\* 정답

- 실행 결과: True, False

- 이유: 'is'는 주소를 비교하기 위한 명령어로 리스트 간에는 'b = a'라고 작성하는 순간 두 변수는 같은 리스트에 대해 주소를 참조한다. 즉 a, b, c는 모두 포함된 값은 동일하지만, a, b는 같은 주소를 참조하고, c는 새로운 주소를 참조한다.

10. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
admission_year = input("당신의 학년을 입력하시오 : ")
print(type(admission_year))
```

<실행 결과>

```
>>> 당신의 학년을 입력하시오 : 3
```

① <class 'int'>

② <class 'str'>

③ <class 'float'>

④ <class 'bool'>

⑤ None

\* 정답 : ②



## Chapter 04. 조건문과 반복문

1. 주어진 자연수  $N$ 에 대해  $N$ 이 짝수이면  $N!$ 을, 홀수이면  $\Sigma N$ 을 구하는 코드를 작성하려 한다. 주어진 코드에서 잘못된 부분을 수정하시오.

```
def calculator(N):
    if n % 2 == 0:
        result = 1
        for i in range(1, N + 1):
            result = result * i
    else:
        result = 0
        for i in range(1, N + 1):
            result = result + i
    return result
```

\* 정답

- 4행:  $N \rightarrow N + 1$
- 8행:  $N \rightarrow N + 1$

2. 다음 코드의 실행 결과를 쓰시오.

```
fruit = 'apple'
if fruit == 'Apple':
    fruit = 'Apple'
elif fruit == 'fruit':
    fruit = 'fruit'
else:
    fruit = fruit
print(fruit)
```

\* 정답 : apple

3. 다음 코드의 실행 결과를 쓰시오.

```
num = ['12', '34', '56']
for i in num:
    i = int(i)
print(num)
```

\* 정답 : ['12', '34', '56']

4. 다음 코드의 실행 결과를 쓰시오.

```
number = ["1", 2, 3, float(4), str(5)]
if number[4] == 5:
    print(type(number[0]))
elif number[3] == 4:
    print(number[2:-1])
```

\* 정답 : [3, 4.0]

5. 다음 코드의 실행 결과를 쓰시오.

```
num = 0
i = 1

while i < 8:
    if i % 3 == 0:
        break
    i += 1
    num += i
print(num)
```

\* 정답 : 5

6. 다음 코드의 실행 결과를 쓰시오.

```
result = 0
for i in range(5, -5, -2):
    if i < -3:
        result += 1
    else:
        result -= 1

print(result)
```

\* 정답 : -5

7. 다음 코드의 실행 결과를 쓰시오.

```
fruit = 'apple'
if fruit == 'Apple':
    fruit = 'Apple'
elif fruit == 'fruit':
    fruit = 'fruit'
else:
```

```
fruit = fruit
print(fruit)
```

\* 정답 : apple

8. 다음 코드의 실행 결과를 쓰시오.

```
first_value = 0
second_value = 0
for i in range(1, 10):
    if i is 5:
        continue
    first_value = i
    if i is 10:
        break
    second_value = i

print(first_value + second_value)
```

\* 정답 : 0

9. 다음 코드의 실행 결과를 쓰시오.

```
num = ""
for i in range(10):
    if i <= 5 and (i % 2)==0:
        continue
    elif i is 7 or i is 10:
        continue
    else:
        num = str(i) + num
print(num)
```

\* 정답 : 986531

10. 다음 함수는 작업의 상태를 나타내는 함수이다. 코드의 실행 결과를 쓰시오.

```
def work_status(task,worker,day):
    rest_task = task
    for k in range(day):
        if rest_task > 0:
            rest_task = rest_task-worker
        elif rest_task <= 0:
            print("Task end")
    if rest_task > 0:
```

```
print("Hire more workers")
```

```
work_status(100, 11, 10)  
work_status(100, 1, 10)  
work_status(100, 9, 10)  
work_status(100, 10, 10)
```

\* 정답 :

Hire more workers

Hire more workers

11. 다음 코드의 실행 결과를 쓰시오.

```
score_list = [5, 10, 15, 20, 25, 30]  
  
sum_of_score = 0  
i = 0  
while i < len(score_list) :  
    if i % 2 == 0 :  
        sum_of_score += score_list[i]  
    i += 1  
  
print(sum_of_score)
```

\* 정답 : 45

12. 다음 코드의 실행 결과를 쓰시오.

```
coupon = 0  
money = 200000  
coffee = 3500  
while money > coffee:  
    if coupon < 4:  
        money = money - coffee  
        coupon += 1  
    else:  
        money += 2800  
        coupon = 0  
print(money)
```

\* 정답 : 2600

13. 다음 코드는 변수 a와 b의 숫자를 비교하여 숫자와 자릿수가 같으면 'strike', 숫자는 서로 포함하나 자릿수가 다르면 'ball'로 카운트하는 프로그램이다. 다음 프로그램의 실행 결과를 쓰시오. 또한, 결과값이 맞지 않는 이유를 쓰고, 올바른 결과값이 나오도록 코드를 수정하시오.

```
a = "369"
b = "693"

strike = 0
ball = 0

for number in a:
    if b.count(number) > 1:
        if b.find(number) != a.find(number):
            strike += 1
    else:
        ball += 1

print("Strike:", strike, "Ball:", ball)
```

\* 정답

- 결과 : strike: 0 Ball: 0
- 오류 수정(7~12행)

```
for number in a:
    if b.count(number) == 1:
        if b.find(number) == a.find(number):
            strike += 1
    else:
        ball += 1
```

14. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
list_data_a = [1, 2]
list_data_b = [3, 4]

for i in list_data_a:
    for j in list_data_b:
        result = i + j
print(result)
```

- ① 20                      ② 6                      ③ [13, 14, 23, 24]                      ④ [4, 5, 5, 6]                      ⑤ Error

\* 정답 : ②

15. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
list_1 = [[1, 2], [3], [4, 5, 6]]  
a,b,c = list_1  
list_2 = a + b + c  
  
print(list_2)
```

- ① [1, 2, 3, 4, 5, 6]      ② [[1, 2], [3], [4, 5, 6]]      ③ 21  
④ Error      ⑤ [[1, 2], [3, 4, 5, 6]]

\* 정답 : ①

## Chapter 05. 함수

### 1. 다음 코드의 실행 결과를 쓰시오.

```
def test(t):  
    t = 20  
    print ("In Function:", t)  
  
x = 10  
print ("Before:", x)  
test(x)  
print ("After:", x)
```

\* 정답 :

Before: 10

In Function: 20

After: 10

### 2. 다음 코드의 실행 결과를 쓰시오.

```
number = "100"  
def midterm(number):  
    result = ""  
    if number.isdigit() is True:  
        if number is 100:  
            if number/10 == 1:  
                result = True  
    else:  
        result = False  
  
    return result
```

\* 정답 : 아무것도 출력되지 않음

### 3. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def is_yes(your_answer):  
    if your_answer.upper() == "YES" or your_answer.upper() == "Y":  
        result = your_answer.lower()
```

```
print(is_yes("Yes"))
```

- ① Error                      ② 'Yes'                      ③ None  
④ 'yes'                      ⑤ 'YES'

\* 정답 : ③

4. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def add_and_mul(a, b, c):  
    return b + a * c + b  
  
print(add_and_mul(3, 4, 5) == 63)
```

- ① 63                      ② 2.39                      ③ True  
④ False                      ⑤ 5.23

\* 정답 : ④

5. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def args_test_3(one, two, *args, three):  
    print(one + two + sum(args))  
    print(args)  
  
args_test_3(3, 4, 5, 6, 7)
```

- ① 25 (5, 6, 7)                      ② 20 (6, 7)                      ③ TypeError  
④ 25 (6, 7)                      ⑤ 20 (5, 6, 7)

\* 정답 : ③

6. 다음 코드의 실행 결과를 쓰시오.

```
def rain(colors):  
    colors.append("purple")  
    colors = ["green", "blue"]  
    return colors  
  
rainbow = ["red", "orange"]  
print(rain(rainbow))
```

\* 정답 : ['green', 'blue']



7. 다음 코드의 실행 결과를 쓰시오.

```
def function(value):  
    print(value ** 3)  
  
print(function(2))
```

\* 정답 :

8

None

8. 다음 코드의 실행 결과를 쓰시오.

```
def get_apple(fruit):  
    fruit = list(fruit)  
    fruit.append("e")  
    fruit = ["apple"]  
    return fruit  
  
fruit = "appl"  
get_apple(fruit)  
print(fruit)
```

\* 정답 : appl

9. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def return_sentence(sentence, n):  
    sentence += str(n)  
    n -= 1  
    if n < 0:  
        return sentence  
    else:  
        return(return_sentence(sentence, n))  
  
sentence = "I Love You"  
print(return_sentence(sentence, 5))
```

① None

② I Love You

③ I Love You543210

④ I Love You54321

⑤ I Love You5

\* 정답 : ③

10. 다음 코드의 실행 결과를 쓰시오.

```
def test(x, y):
    tmp = x
    x = y
    y = tmp
    return y.append(x)

x = ["y"]
y = ["x"]
test(x, y)
print(y)
```

\* 정답 : ['x']

11. 다음 코드의 실행 결과를 쓰시오.

```
def countdown(n):
    if n % 2 == 0:
        print ("Even")
    else:
        print ("Odd")
        countdown(n-1)
countdown(3)
```

\* 정답 :

Odd

Even

12. 다음 코드를 실행하면 결과값으로 120이 나온다. 빈칸에 들어갈 알맞은 코드를 작성하시오.

```
def factorial_calculator(n):
    if n in (0, 1):
        return 1
    else:
        return n * 
print(factorial_calculator(5))
```

\* 정답 : factorial\_calculator(n - 1)

13. 다음 코드의 실행 결과를 쓰시오.

```
def calculate_rectangle_area(rectangle_x,rectangle_y):  
    rectangle_x = 3  
    rectangle_y = 5  
    result = rectangle_x * rectangle_y  
    return result  
  
rectangle_x = 2  
rectangle_y = 4
```

\* 정답 : 아무 것도 출력되지 않음

## Chapter 06. 문자열

### 1. 다음 코드의 실행 결과로 알맞은 것은?

```
a = 11
b = 9

print('a' + 'b')
```

- ① 'a' + 'b'                      ② 20                      ③ ab  
④ a + b                      ⑤ error

\* 정답 : ③

### 2. 다음 코드의 실행 결과로 알맞은 것은?

```
fact = "Python is funny"
print(str(fact.count('n') + fact.find('n') + fact.rfind('n')))
```

- ① 9                      ② 21                      ③ 3513  
④ 18                      ⑤ 333

\* 정답 : ②

### 3. 다음 코드의 실행 결과로 알맞은 것은?

```
text = 'Gachon CS50 - programming with python'
text2 = " Human cs50 knowledge belongs to the world "

text.lower()

print(text[:5] + text[-1] + text[6] + text2.split()[0])
```

- ① Gachon Human                      ② gachon Human                      ③ gachon H  
④ gachon                      ⑤ gachon cs50

\* 정답 : ①

### 4. 다음 코드의 실행 결과로 알맞은 것은? ②

```
class_name = 'introduction programming with python'

for i in class_name:
    if i == 'python':
        i = i.upper()
```

```
print(class_name)
```

- ① introduction programming with PYTHON
- ② introduction programming with python
- ③ INTRODUCTION PROGRAMMING WITH PYTHON
- ④ Introduction programmiNg wIth python
- ⑤ Introduction Programming With Python

\* 정답 : ②

5. 다음 코드의 실행 결과를 쓰시오.

```
a = '10'  
b = '5-2'.split('-')[1]  
  
print(a * 3 + b)
```

\* 정답 : 1010102

6. 다음 코드의 실행 결과를 쓰시오.

```
a = "abcd e f g"  
b = a.split()  
c = (a[:3][0])  
d = (b[:3][0][0])  
  
print(c + d)
```

\* 정답 : aa

7. 다음 코드의 실행 결과가 다음과 같이 나올 수 있도록 빈칸에 코드를 작성하시오.

```
number = 10  
day = 3  
print("I eat %d oranges every day." % number)  
  
number = 10  
day = "three"
```

```
I eat 10 oranges every day.  
I eat 10 oranges every three days.
```

\* 정답 : print("I eat %d oranges every %s days." % (number, day))

8. 다음 코드의 실행 결과를 쓰시오.

```
result = "CODE2018"  
print("{0},{1}".format(result[-1], result[-2]))
```

\* 정답 : 8, 1

9. 다음 코드의 실행 결과가 다음과 같이 나올 수 있도록 빈칸에 코드를 작성하시오.

```
str_a = "this is"  
str_b = "PythoN"  
  
print(str_a. (가) + " " + str_b. (나))
```

This Is PYTHON.

\* 정답 : (가) title(), (나) upper()

10. 다음과 같은 결과가 나올 수 있도록 서식 지정(formatting)을 사용하여 코드를 작성하시오.

```
*  
***  
*****  
*****  
*****  
*****  
*****
```

\* 정답 :

```
for i in range(1, 12, 2):  
    print("{:^11}".format("*"*i))
```

11. 다음 코드의 의미를 설명하시오.

```
>>> print("{0:>10s}".format("Apple"))
```

\* 정답 : "Apple"이라는 텍스트에 총 10개의 텍스트가 들어갈 수 있는 공간으로 좌측 정렬을 하시오.

12. 서식 지정 기능에서 패딩(padding)과 네이밍(naming)의 차이를 서술하시오

\* 정답 : 패딩은 글자를 출력할 때 일정 공간을 남기는 기법을 말하며, 네이밍은 변수명을 사용하여 서식 지정을 할 수 있게 하는 것을 의미한다.

## Chapter 07 자료구조

1. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
animal = ['cat', 'snake', 'monkey', 'ant', 'spider']
legs= [4, 0, 2, 4, 8]
animal_legs_dict = {}
for i in range(len(animal)):
    animal_legs_dict[legs[i]] = animal[i]
animal_legs_dict['ant'] = 6
print(animal_legs_dict)
```

- ① {0: 'snake', 8: 'spider', 2: 'monkey', 6: 'ant', 4: 'cat'}
- ② {0: 'snake', 8: 'spider', 2: 'monkey', 4: 'ant', 4: 'cat', 'ant': 6}
- ③ {0: 'snake', 8: 'spider', 2: 'monkey', 4: 'ant', 'ant': 6}
- ④ {4: 'ant', 0: 'snake', 2: 'monkey', 8: 'spider', 'ant': 6}
- ⑤ {0: 'snake', 8: 'spider', 2: 'monkey', 6: 'ant'}

\* 정답 : ④

2. 다음 코드의 실행 결과를 쓰시오.

```
t = (1, 2, 3)
print(t + t)
print(t * 2)
print(t, t)
```

\* 정답 :

(1, 2, 3, 1, 2, 3)

(1, 2, 3, 1, 2, 3)

(1, 2, 3) (1, 2, 3)

3. 다음 코드의 실행 결과를 쓰시오.

```
Mydict = {'1' : 1, '2' : 2}
Copy = Mydict
Mydict['1'] = 5
result= Mydict['1'] + Copy['1']
print(result)
```

\* 정답 : 10

4. 다음 코드의 실행 결과를 쓰시오.

```
sentence = list('You Love Me?')
result = ''
for i in range(len(sentence)):
    if i % 3 == 0:
        result += sentence.pop()
    else:
        result += sentence.pop(0)
print(result)
```

\* 정답 : ?Yoeu MLo ve

5. 다음 코드의 실행 결과를 쓰시오.

```
number = [5, 6, 7, 8, 9, 1, 2, 3, 4]
result = [ ]
result.append(number.pop(0))
result.append(number.pop())
result.append(number.pop(1))
result.append(number.pop())
result.append(number.pop(0))
print(number[0]+result[-1])
```

\* 정답 : 14

6. 다음 코드의 실행 결과를 쓰시오.

```
a = list(range(10))
a.append(a[3])
a.pop(a[3])
a.insert(3, a[-1])
a.pop( )
print(a)
```

\* 정답 : [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

7. 다음 코드의 실행 결과를 쓰시오.

```
data_1 = {'one' : (1,2,3,4,5,6), 'two' : [1,2,3,4,5,6], 'three' : {'four' : 4, 'five' : 5}}

for k in ['one', 'two', 'three']:
```



```

try:
    print(data_1[k][:1])
except TypeError:
    print("error")

for k in ['one', 'two', 'three']:
    try:
        data_1[k][-1] = "a"
        print(data_1[k][-1])
    except TypeError :
        print("error")

```

\* 정답 :

(1,)

[1]

error

error

a

a

8. 다음 코드의 실행 결과를 쓰시오.

```

class_category = ["A", "B", "C", "D"]
student_category = ["Sam", "Sarah", "Jane", "John"]

class_student_cate = { }
for i in range(len(class_category)):
    class_student_cate[class_category[i]] = student_category[i]
print(class_student_cate)

```

\* 정답 : {'A': 'Sam', 'B': 'Sarah', 'C': 'Jane', 'D': 'John'}

## Chapter 08 파이썬 스타일 코드 I

1. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
mylist = ['apple', 'banana', 'grape']
result = list(enumerate(mylist))
print(result)
```

- ① [('apple', 1), ('banana', 2), ('grape', 3)]
- ② [(1, 'apple'), (2, 'banana'), (3, 'grape')]
- ③ [(0, 'apple'), (1, 'banana'), (2, 'grape')]
- ④ [('apple', 0), ('banana', 1), ('grape', 2)]
- ⑤ [('grape', 0), ('banana', 1), ('apple', 2)]

\* 정답 : ③

2. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
cat_song = "my cat has blue eyes, my cat is cute"
print({i:j for j,i in enumerate(cat_song.split())})
```

- ① {0: 'my', 1: 'cat', 2: 'has', 3: 'blue', 4: 'eyes,', 5: 'my', 6: 'cat', 7: 'is', 8: 'cute'}
- ② {'my': 0, 'cat': 1, 'has': 2, 'blue': 3, 'eyes,': 4, 'my': 5, 'cat': 6, 'is': 7, 'cute': 8}
- ③ {0: 'my', 1: 'cat', 2: 'has', 3: 'blue', 4: 'eyes,', 5: 'is', 6: 'cute'}
- ④ {'my': 5, 'cat': 6, 'has': 2, 'blue': 3, 'eyes,': 4, 'is': 7, 'cute': 8}
- ⑤ 오류

\* 정답 : ④

3. 다음과 같이 코드를 작성했을 때, 예측되는 실행 결과를 쓰시오.

```
colors = ['orange', 'pink', 'brown', 'black', 'white']
result = '&'.join(colors)
print(result)
```

\* 정답 : orange&pink&brown&black&white

4. 다음 코드의 실행 결과를 쓰시오.

```
week = ['mon', 'tue', 'wed', 'thu', 'fri', 'sat', 'sun']
rainbow = ['red', 'orange', 'yellow', 'green', 'blue', 'navy', 'purple']
list_data = [week, rainbow]
print(list_data[1][2])
```

\* 정답 : yellow

5. 다음 코드의 실행 결과를 쓰시오.

```
kor_score = [30, 79, 20, 100, 80]
math_score = [43, 59, 0, 30, 90]
eng_score = [49, 72, 48, 67, 15]
midterm_score = [kor_score, math_score, eng_score]
print ("score:",midterm_score[2][1])
```

\* 정답 : score: 72

6. 다음 코드의 실행 결과를 쓰시오.

```
alist = ["a", "b", "c"]
blist = ["1", "2", "3"]
abcd= []

for a, b in enumerate(zip(alist, blist)):
    try:
        abcd.append(b[a])
    except IndexError:
        abcd.append("error")

print(abcd)
```

\* 정답 : ['a', '2', 'error']

7. 다음과 같은 결과값을 출력하기에 적당한 함수를 빈칸에 쓰시오.

```
alist = ['a1', 'a2', 'a3']
blist = ['b1', 'b2', 'b3']

for a, b in  (alist, blist):
    print(a, b)
```

```
a1 b1
a2 b2
a3 b3
```

\* 정답 : zip

8. 다음 코드의 실행 결과로 'Gachon cs50'을 출력하기 위해 빈칸에 들어가야 할 값을 순서대로 쓰시오.

```
url = "cs50.gachon.ac.kr"
university_name = "Gachon University"
gachon_info = [(i + " " + j) for i, j in  (university_name.split(" "), url.  
 ("."))]
blank = 
print(gachon_info[blank])
```

\* 정답 : (가) zip      (나) split      (다) 0

## Chapter 09. 파이썬 스타일 코드 II

1. 다음 코드를 람다 함수 형태로 수정할 때, 알맞은 코드를 작성하시오.

```
def f(x, y):  
    return x ** y
```

\* 정답 : `f = lambda x, y : x ** y`

2. 다음과 같이 리스트 컴프리헨션으로 되어 있는 코드를 람다(lambda) 함수와 `map()` 함수를 사용하여 표현하시오.

```
>>> ex = [1, 2, 3, 4, 5]  
>>> [value **2 for value in ex]  
[1, 4, 9, 16, 25]
```

\* 정답 :

```
>>> f = lambda x : x ** 2  
>>> list(map(f, ex))
```

3. 팩토리얼(Factorial)을 계산하는 코드를 람다 함수와 `reduce()` 함수를 사용하여 표현하시오.

\* 정답 :

```
from functools import reduce  
x = 10  
reduce(lambda x, y: x * y, list(range(x, 0, -1)))
```

4. 다음 코드를 각각 실행하면 서로 다른 결과가 나온다. 이런 결과가 나오는 이유에 대해 서술하시오.

```
>>> a = [1, 2, 3]  
>>> print(*a)  
1 2 3  
>>> print(a)  
[1, 2, 3]
```

\* 정답 : 별표(asterisk, \*)는 함수의 인자(argument)로 사용될 때, 시퀀스형 자료형의 각각의 값(element)을 하나의 값(element)으로 넣어주게 된다. 그러므로 'print(\*a)'의 의미는 사실 'print(a[0], a[1], a[2])'으로 나타나게 된다.

5. 다음과 같이 2개 이상의 행렬을 더하는 코드를 작성하시오.

```
>>> matrix_y = [[2, 5], [2, 1]]
>>> matrix_z = [[2, 4], [5, 3]]
>>> matrix_addition(matrix_y, matrix_z)
[[4, 9], [7, 4]]
```

\* 정답 :

```
>>> def matrix_addition(*matrix_variables):
...     return [[sum(row) for row in zip(*t)] for t in zip(*matrix_variables)]
...
```

6. 다음과 같이 2개 이상의 벡터를 빼는 코드를 작성하시오.

```
>>> print(vector_subtraction([1, 3], [2, 4]))
[-1, -1]
>>> print(vector_subtraction([1, 5], [10, 4], [4, 7]))
[-13, -6]
```

\* 정답 :

```
>>> def vector_subtraction(*vector_variables):
...     return [value[0] - sum(value[1:]) for value in zip(*vector_variables)]
...
```

## Chapter 10 객체 지향 프로그래밍

1. 다음과 같은 2개의 파일이 있다. 'main.py'를 실행하였을 때 나오는 결과로 알맞은 것은? ①  
<factorial\_calculator.py>

```
def factorial(n):  
    if n == 0:  
        return 1  
    else:  
        return ( n * factorial(n-1))
```

<main.py>

```
from factorial_calculator import factorial  
print(factorial_calculator.factorial(6))
```

- ① NameError                      ② None                      ③ 120  
④ 720                              ⑤ TypeError

\* 정답 : ①

2. 다음 코드의 실행 결과를 쓰시오.

```
class SoccerPlayer(object):  
    def __init__(self, name, position, back_number):  
        self.name = name  
        self.position = position  
        self.back_number = back_number  
  
    def change_back_number(self, back_number):  
        self.back_number = back_number  
  
jinhyun = SoccerPlayer("jinhyun", "MF", 10)  
print("현재 선수의 등번호는:", jinhyun.back_number)  
jinhyun.change_back_number(5)  
print("현재 선수의 등번호는:", jinhyun.back_number)
```

\* 정답 :

현재 선수의 등번호는: 10

현재 선수의 등번호는: 5

3. 다음과 같은 코드는 객체 지향 프로그램의 어떤 특징을 보여 주는지 모두 고르시오.

```
class Class(object):
    def __init__(self, name, score):
        self.name = name
        self.score = score

class Math(Class):
    def say( ):
        print("힘내")
```

- ① 다형성                      ② 가시성                      ③ 상속
- ④ 인스턴스                  ⑤ 속성

\* 정답 : ①, ③

4. 다음 (가)와 (나)에 들어갈 알맞은 단어를 고르시오.



\* 정답 : (가) 클래스, (나) 인스턴스

5. 다음 코드의 실행 결과를 쓰시오.

```
class Score:
    def __init__(self, student):
        tmp = student.split(",")
        self.name = tmp[0]
        self.midterm = int(tmp[1])
        self.final = int(tmp[2])
        self.assignment = int(tmp[3])
        self.score = None
        self.grade = None
```



```

def total_score(self):
    test_score = ((self.midterm + self.final)/2)*0.8

    if self.assignment>=3:
        assign_score = 20
    elif self.assignment>=2:
        assign_score = 10
    elif self.assignment>=1:
        assign_score = 5
    else:
        assign_score = 0

    self.score = test_score + assign_score

def total_grade(self):
    if self.assignment==0:
        grade = "F"
    elif self.score >=90:
        grade = "A"
    elif self.score >=70:
        grade = "B"
    elif self.score >=60:
        grade = "C"

    else:
        grade = "F"

    self.grade = grade
    return grade

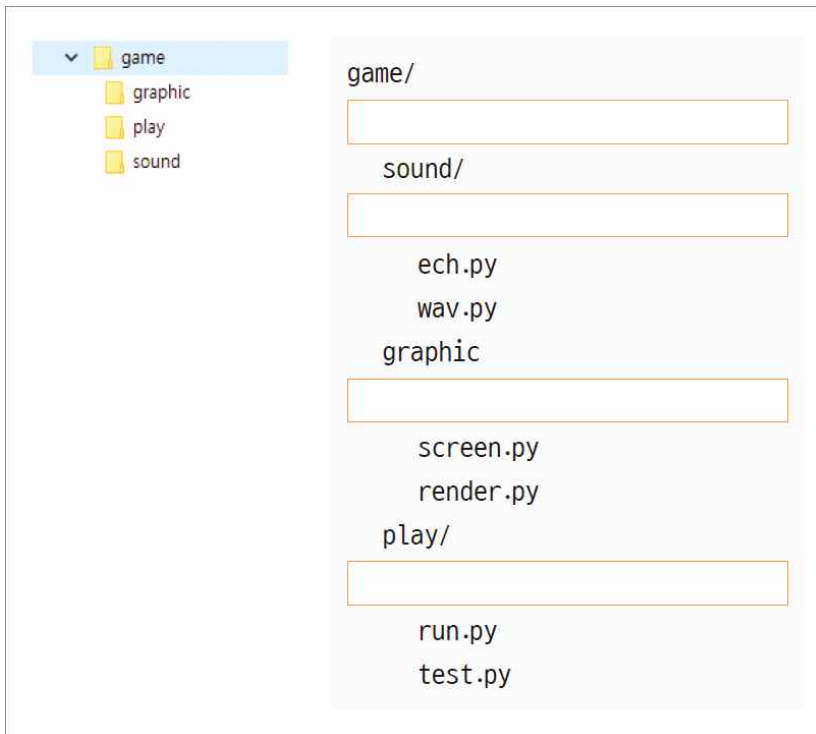
student_john = Score("john,90,90,0")
aa = student_john.total_score()
bb = student_john.total_grade()
print(aa,bb,student_john.score,student_john.grade)

```

\* 정답 : None F 72.0 F

## Chapter 11. 모듈과 패키지

1. 'game'이라는 패키지를 만들고 싶다고 가정하자. 패키지를 만들기 위해 디렉터리별로 필요한 모듈을 구현하고자 한다. 다음 그림에서 빈칸에 들어가야 할 파일은?



- ① `__main__`                      ② `import game`                      ③ `__init__.py`  
④ `__main__.py`                      ⑤ `__init__`

\* 정답 : ③

2. 두 코드 파일인 'fah\_converter.py'와 'module\_ex.py'는 같은 디렉터리에 있다. 다음과 같은 결과 값을 얻기 위해 빈칸에 들어갈 적합한 코드를 쓰시오.

fah\_converter.py

```
def covert_c_to_f(celsius_value):
    return celsius_value * 9.0 / 5 + 32

test_value = 0
```

module\_ex.py

```

print ("Enter a celsius value: ")
```

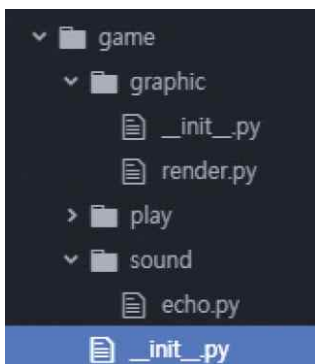
```
celsius = float(input())
fahrenheit = fah.covert_c_to_f(celsius)
print ("That's ", fahrenheit, " degrees Fahrenheit")
```

#### 실행 결과

```
Enter a celsius value:
100
That's 212.0 degrees Fahrenheit
```

\* 정답 : `import fah_converter as fah`

3. 다음과 같이 `game` 패키지를 만들었다. 패키지 내에서 다른 디렉터리의 모듈을 부를 때 부모 디렉터리 기준으로 호출하는 방법은?



\* 정답 : `from game.graphic.render import render_test( )`

4. 'quizpy'와 같이 코드를 작성한 후 저장하고, 파이썬 셸 코드를 실행했을 때의 결과값을 쓰시오. [quiz.py]

```
def quiz():
    user_input = input()
    if int(user_input) > 3:
        print(user_input)
    else:
        print("user_input")
```

#### [파이썬 셸]

```
>>> import quiz as q
>>> q.quiz()
2
```

\* 정답 : `user_input`

## Chapter 12. 예외 처리와 파일

1. 다음과 같이 'hello\_python.txt' 파일을 수정하고자 한다. 빈칸에 들어가야 하는 코드를 쓰시오.

[수정 전 'hello\_python.txt']

```
hello, world!
```

→

[수정 후 'hello\_python.txt']

```
hello, world!  
hello, python!
```

[코드]

```
f = open("hello_python.txt", ' (가) ')  
f. (나) ('hello, python!')
```

\* 정답 : (가) a (나) write

2. 파이썬은 파일 처리를 위해 open() 함수를 사용한다. “파일 열기 모드”에 따라 들어갈 값을 쓰시오.

```
f = open("<파일 이름>", "파일 열기 모드")  
f.close()
```

(가) 쓰기 모드

(나) 추가 모드

(다) 읽기 모드

\* 정답 : (가) w (나) a (다) r

3. 다음 코드의 실행 결과를 쓰시오.

```
def get_half_value(number):  
    assert isinstance(number, int), 'input value must be integer.'  
    result = number / 2  
    assert isinstance(result, int), 'output value must be transformed integer.'  
    return result  
  
print(get_half_value(50))
```

\* 정답 : AssertionError: output value must be transformed integer.

4. 다음 코드의 실행 결과를 쓰시오.

```
for i in list_data_a:  
    for j in list_data_b:
```

```
        result = i+j
    return result

a = [1, 2]
b = [3, 4]
print(sum_data(a, b))
```

\* 정답 : 6

5. 다음 코드는 파이썬에서 'i\_have\_a\_dream.txt' 파일을 읽어오는 코드이다. 같은 기능을 하는 코드를 with 구문과 함께 사용하여 작성하시오.

```
f = open("i_have_a_dream.txt", "r")
contents = f.read()
print(contents)
f.close()
```

\* 정답 :

```
with open("i_have_a_dream.txt","r") as my_file:
    contents = my_file.read()
    print(contents)
```

## Chapter 13. CSV와 로그 처리

1. 다음은 csv 객체를 이용하여 csv 파일을 tsv 파일로 변환하는 코드이다. 빈칸에 들어갈 코드를 차례대로 쓰시오.

```
import csv
# my_first.csv 파일은 이미 작업 디렉토리에 있다고 가정
# tsv로 변환되는 파일명은 transfer.tsv

read_file = open('./my_first.csv', 'r')
reader = csv.reader(read_file, delimiter= (가) )
with open( (나) , 'w') as writer_file:
    writer = csv.writer(writer_file, delimiter= (다) )
    for row in (라) :
        writer.(마) (row)
```

\* 정답

(가) “,”      (나) “transfer.tsv”      (다) “\t”      (라) reader      (마) writerow

2. 다음은 ‘techusa.csv’ 파일을 분석하여 특정 데이터를 출력하는 프로그램이다. 빈칸에 들어갈 코드를 차례대로 쓰시오.

[techusa.csv 데이터]

```
permalink,company,numEmps,category,city,state,fundedDate,raisedAmt,raisedCurrency,round
lifelock,LifeLock,,web,Tempe,AZ,1-May-07,6850000,USD,b
lifelock,LifeLock,,web,Tempe,AZ,1-Oct-06,6000000,USD,a
lifelock,LifeLock,,web,Tempe,AZ,1-Jan-08,25000000,USD,c
mycityfaces,MyCityFaces,7,web,Scottsdale,AZ,1-Jan-08,50000,USD,seed
flypaper,Flypaper,,web,Phoenix,AZ,1-Feb-08,3000000,USD,a
infusionsoft,Infusionsoft,105,software,Gilbert,AZ,1-Oct-07,9000000,USD,a
gauto,gAuto,4,web,Scottsdale,AZ,1-Jan-08,250000,USD,seed
chosenlist-com,ChosenList.com,5,web,Scottsdale,AZ,1-Oct-06,140000,USD,seed
```

[실행 코드]

```
line_counter = 0
```

```

data_header = []
techusa_list = []

with open (  (가) ) as techusa:
    while 1:
        data =  (나)

        if not data:
            break

        if line_counter==0:
            data_header = data.split(",")
        else:
            techusa_list.append(data.split("  (다) "))
            line_counter += 1

print (techusa_list[1][0])

```

[실행 결과]

(라)

\* 정답

(가) `techusa.csv`    (나) techusa.readline()    (다) ,    (라) lifelock

3. 'cs50\_click\_stream.log' 파일에 사용자와 사용자가 방문한 시간을 저장하고자 한다. 모든 사용자의 정보와 방문 시간이 한 줄씩 저장될 수 있도록 빈칸에 알맞은 코드를 쓰시오(방문 시간은 visit\_time 변수에 저장됨).

```

import datetime

visitor_info =["Thanos", "Groot", "Dr.strange", "Captain America", "Spider Man"]

with open( (가) ,  (나) , encoding =  (다) ) as f:
    for i in visitor_info:
        visit_time = datetime.datetime.now()

        f.write("userName : %s \t%i + " visitedTime : %s  (라) "%str(visit_time))

```

\* 정답

(가) "cs50\_click\_stream.log"    (나) "a"    (다) "utf8"    (라) \n

#### 4. 프로그래밍 언어에서 로깅을 사용하는 목적에 대해 서술하시오

\* 정답 : 프로그램의 개발 단계에서 중간 중간 나오는 산출물에 대해 자세히 이력(history)을 남김으로써 사용자 입장에서 이후에 디버깅이나 시스템 개선에 필요한 유용한 정보를 획득할 수 있다.

#### 5 다음과 같은 각각의 조건에 대해 실시해야 하는 로깅 타입을 쓰시오.

로깅 타입: debug, info, warning, error, critical
--

(가) 함수에 인자(argument)로 이차원 리스트를 기대했지만, 일차원 리스트가 들어옴, 이차원으로 변환 후 처리함.

(나) 서버가 시작됨, 사용자 A가 프로그램에 접속함.

(다) 잘못된 접근으로 해당 파일이 삭제됨

(라) 파일에 기록을 해야 하는데 파일이 없음 → 예외 처리 후, 사용자에게 알림.

\* 정답

(가) WARNING      (나) INFO      (다) CRITICAL      (라) ERROR



## Chapter 14 웹 스크래핑

### 1. 정규 표현식에 대한 설명으로 옳은 것은?

- ① .은 줄 바꿈인 '\n'과 '\t'를 제외한 모든 문자와 매치된다.
- ② {0, }는 +와 동일하다
- ③ yester \*day로 yesterday, yesteday, yesterrrrday가 모두 반환될 수 있다.
- ④ [a-zA-z]는 모든 알파벳과 특수문자 '-'과 매치된다.
- ⑤ {1,3}는 1이 3번 반복된다는 뜻이다.

\* 정답 : ③

### 2. 다음 정규 표현식의 결과에 해당하는 것을 모두 고르면?

```
regex = (https?: \ / \ /)?([0-9a-zA-Z]+\.) {2,5}([0-9a-zA-Z]+)
```

- ① https://gachon.ac.kr
- ② http://cyber.gachon.ac.kr
- ③ wind.gachon.ac.kr
- ④ theteamlab.io
- ⑤ http://google.com

\* 정답 : ①, ②, ③

### 3. 다음 설명에 적합한 정규 표현식 메타문자를 순서대로 쓰시오.

- (가) 반복 횟수 1회
- (나) or
- (다) 앞에 있는 글자를 최소 1회 이상 반복
- (라) 줄 바꿈 문자인 \n을 제외한 모든 문자와 매치
- (마) not

\* 정답

(가) ? (나) | (다) + (라) , (마) ^

### 4. 정규 표현식이 다음과 같을 때, 검색 가능한 값을 쓰시오.

- (가) (?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)
- (나) ([a-zA-Z0-9\_-\.\~]+)@[a-z0-9-]+\.[a-z0-9-~]\*\.[a-z]{2,3}
- (다) (https?:\ / \ /)?([da-z\.\~]+)\.[a-z]{2,6}([ \w \.\~]\*)\*\ /?

\* 정답

(가) 203.223.232.223      (나) teamlab.gachon@gmail.com      (다) https://www.google.com

**5. 인터넷을 사용하기 위해 알아야 하는 다음 키워드들의 의미를 서술하시오.**

IP 주소, URL, HTML, HTTP
------------------------

**\* 정답**

IP 주소 : 인터넷 프로토콜 주소(Internet Protocol Address), 컴퓨터가 사용하는 물리적인 주소 정보

URL : Uniform Resource Locator, 웹상의 특정 정보가 위치해 있는 주소

HTML : Hyper Text Markup Language, 웹상에서 구조적으로 정보를 표현하기 위한 웹 표현 언어

HTTP : Hypertext Transfer Protocol, 인터넷에서 서로 통신하기 위한 표준 규약

## Chapter 15. XML과 JSON

1. 'questions\_and\_answers.json' 파일을 이용해 다음과 같은 결과값을 내고자 한다. 빈칸에 알맞은 코드를 순서대로 쓰시오.

[questions\_and\_answers.json]

```
{
  "quiz": {
    "sport": {
      "q1": {
        "question": "Which one is correct team name in NBA?",
        "options": [
          "New York Bulls",
          "Los Angeles Kings",
          "Golden State Warriros",
          "Huston Rocket"
        ],
        "answer": "Huston Rocket"
      }
    }
  }
}
```

[코드 입력]

```
import json
with open("questions_and_answers.json", "r", encoding = "utf8") as f:
    contents = f.read()
    json_ex = json.loads(contents)

print(json_ex [가] + "\n")

options = json_ex [나]
for i in range(len(options)):
    print(str(i) + ". " + options[i])

print("\n answer " + json_ex [다] )
```

[실행 결과]

Which one is correct team name in NBA?

- 0. New York Bulls
- 1. Los Angeles Kings
- 2. Golden State Warriors
- 3. Huston Rocket

answer Huston Rocket

\* 정답

(가) : ["quiz"]["sport"]["q1"]["question"]

(나) : ["quiz"]["sport"]["q1"]["options"]

(다) : ["quiz"]["sport"]["q1"]["answer"]

2. 'researcher.xml' 파일을 이용해 코드로 작성한 결과값을 쓰시오.

[researcher.xml]

```
<researchers>
  <researcher researcherID = "0001">
    <na>Tedd</na>
    <position>Professor</position>
    <degree>Doctor of Engineering</degree>
    <salary format="dollar">70000</salary>
    <day_off>Y</day_off>
  </researcher>
  <researcher researcherID = "0002">
    <na>Henry</na>
    <position>Senior.researcher</position>
    <degree>Master of Engineering</degree>
    <salary format="dollar">7000</salary>
    <day_off>N</day_off>
  </researcher>
  <researcher researcherID = "0003">
    <na>John</na>s
    <position>Junior.researcher</position>
    <degree>Bachelor of Engineering</degree>
    <salary format="dollar">700</salary>
    <day_off>Y</day_off>
  </researcher>
```

```
</researchers>
```

[코드 입력]

```
import urllib.request
from bs4 import BeautifulSoup

with open("researcher.xml", "r", encoding = "utf-8") as researcher_xml:
    xml = researcher_xml.read()

soup = BeautifulSoup(xml,"html.parser")

for i,element in enumerate(soup.findAll('researcher')):
    if i%2==1:
        print(element['researcherid'])
        print(element.name.get_text())
        print(element.salary)
    else:
        pass
```

\* 정답

0002

Henry

<salary format="dollar">7000</salary>

### 3. XML과 JSON 파일 형식의 차이에 대해 서술하시오.

- XML : eXtensible Markup Language의 줄임말로, HTML과 동일한 구조인 XML을 사용하여 데이터를 표현한다. 태그를 사용하기 때문에 데이터의 표현이 매우 길어진다. 전통적으로 2000년대 레거시(legacy) 시스템에서 매우 많이 사용되는 방식이다.
- JSON : JavaScript Object Notation의 줄임말로, 자바스크립트에서 객체를 표현하는 언어의 기법을 빌려와 데이터를 표현한다. 키:값(key:value) 쌍을 쓰기 때문에 기존 파이썬의 딕셔너리 자료형과 유사한 형식으로 데이터를 처리할 수 있다.

### 4. 이 책에서는 BeautifulSoup을 사용하여 XML을 파싱하였다. 해당 모듈을 사용할 때 사용자 입장에서 얻게 되는 장점을 쓰시오.

- \* 정답 : 마크(Markup) 언어를 손쉽게 스크래핑하게 해주어 사용자의 코드의 양이 줄어들고, 기존 정규 표현식 대비 간단히 태그(tag) 정보를 추출할 수 있음

5. 다음 코드의 빈 칸에 들어갈 함수를 쓰고, 그 함수의 역할을 서술하시오.

```
from bs4 import BeautifulSoup

with open("books.xml", "r", encoding="utf8") as books_file:
    books_xml = books_file.read()
soup = BeautifulSoup(books_xml, "lxml")

for book_info in soup. (가) ("author"):
    print (book_info)

    print (book_info. (나) )
```

\* 정답

(가) find\_all( ) : 필요한 태그(tag)를 검색해서 여러 개의 태그를 반환하는 함수, 여기서는 author 이름의 태그를 검색하여 반환함

(나) get\_text( ) : 해당 태그 사이에 있는 값을 반환함

6. 다음 코드의 목적을 쓰시오.

```
import json

dict_data = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
with open("data.json", "w") as f:
    json.dump(dict_data, f)
```

\* 정답 : 기존 dict\_data에 있는 정보를 'data.json' 파일에 작성하여 새로운 파일을 생성한다.

7. 특정 시스템에 있는 정보를 추출하기 위해 최근 API를 많이 사용한다. 이때 사용되는 API 문서란 무엇인지 서술하시오.

\* 정답 : 특정 시스템을 호출해서 사용할 때 인수(argument)와 반환되는(return) 값(value)에 대한 자세한 사용 설명을 작성해 놓은 문서이다.

8. 다음과 같은 JSON 데이터가 있을 때, 'species'의 정보를 가져올 수 있는 파이썬 코드를 작성하시오.

[data\_file.json]

```
{
  "president": {
    "name": "Zaphod Beeblebrox",
    "species": "Betelgeusian"
  }
}
```

\* 정답

```
import json
with open("data_file.json", "r") as read_file:
    data = json.load(read_file)

print(data["president"]["species"])
```