

- 01 파이썬 스타일 코드 `pythonic code` 는 파이썬 스타일의 코드 작성 기법이다.
- 02 파이썬 스타일 코드가 생긴 이유는 바로 파이썬의 철학 때문이다. 파이썬은 기본적으로 '인간의 시간이 컴퓨터의 시간보다 더 중요하다.'라는 개념을 가지고 있다. 그러다 보니 코드상으로 사람이 해야 하는 일을 최대한 줄이면서 같은 목표를 달성할 수 있는 문법 체계를 가지고 있다.
- 03 `split()` 함수는 문자열의 값을 특정 값을 기준으로 분리하여 리스트 형태로 변환하는 방법이다.
- 04 `join()` 함수는 문자열로 구성된 리스트를 합쳐 하나의 문자열로 반환할 때 사용한다. `join()` 함수를 사용하는 방법은 구분자, `join(리스트형)` 형태로 사용할 수 있다.
- 05 리스트 컴프리헨션 `list comprehension` 의 기본 개념은 기존 리스트형을 사용하여 간단하게 새로운 리스트를 만드는 기법이다. 리스트와 `for` 문을 한 줄에 사용할 수 있는 장점이 있다.
- 06 리스트 컴프리헨션은 다양한 용법으로 사용할 수 있는데, 대표적으로 필터링, 중첩 반복문, 이차원 리스트가 있다.
- 07 `enumerate()` 함수는 리스트값을 추출할 때 인덱스를 붙여 함께 출력하는 방법이다.
- 08 `zip()` 함수는 1개 이상의 리스트값이 같은 인덱스에 있을 때 병렬로 묶는 함수이다.

01 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)]

02 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

03 다음 코드의 실행 결과를 쓰시오.

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

04 다음 코드의 실행 결과를 쓰시오.

```
user_dict = {}
user_list = ["students", "superuser", "professor", "employee"]
for value_1, value_2 in enumerate(user_list):
    user_dict[value_2] = value_1
print(user_dict)
```

05 다음 코드의 실행 결과를 쓰시오.

```
>>> result = [i for i in range(10) if i%2 == 0]
>>> print(result)
[ ]
>>> items = 'zero one two three'.split("two")
>>> result = [i for i in range(10)]
>>> print(result)
[ ]
>>> items = 'zero one two three'.split()
>>> print(items)
[ ]
>>> example = 'cs50.gachon.edu'
>>> subdomain, domain, tld = example.split('.')
>>> print(subdomain)
[ ]
```

06 다음 코드의 실행 결과를 쓰시오.

```
animal = ['Fox', 'Dog', 'Cat', 'Monkey', 'Horse', 'Panda', 'Owl']
print([ani for ani in animal if 'o' not in ani])
```

07 다음 코드의 실행 결과를 쓰시오.

```
name = "Hanbit University"
student = ["Hong", "Gil", "Dong"]
split_name = name.split()
join_student = ''.join(student)
print(join_student[-4:] + split_name[1])
```

08 다음 코드의 실행 결과를 쓰시오.

```
kor_score = [49, 79, 20, 100, 80]
math_score = [43, 59, 85, 30, 90]
eng_score = [49, 79, 48, 60, 100]
midterm_score = [kor_score, math_score, eng_score]
print(midterm_score[0][2])
```

09 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
>>> a = [1, 2, 3]
>>> b = [4, 5, ]
>>> c = [7, 8, 9]
>>> print([[sum(k), len(k)] for k in zip(a, b, c)])
```

- ① [[6, 3], [9, 2], [24, 3]]
- ② [[12, 3], [15, 3]]
- ③ [[12, 3], [15, 3], [17, 3]]
- ④ 오류
- ⑤ [[12, 3], [15, 3], [12, 2]]

10 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])
```

11 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])
```

12 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)
```