

06. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
my_list = ['i', 'like', 'studying', 'python']
new_list = []
i = len(my_list)
if i == 4:
    new_list.append(my_list[::-2])
print(new_list)
```

- ① [['like', 'studying', 'python']]
- ② ['like', 'studying', 'python']
- ③ None
- ④ ['i', 'like', 'python']
- ⑤ [['i', 'studying']]

14. 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 에러 발생
- ⑤ [[1, 2], [3, 4, 5, 6]]

5. 다음 코드의 실행 결과를 쓰시오.

```
num = 0
i = 1

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

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)
```

01. 함수에서 지역변수(local variable)와 전역 변수(global variable)에 대한 설명이 아닌 것은?

- ① 지역변수는 함수 안에서만 사용되는 변수이다.
- ② 전역변수는 함수 밖의 전체 코드에서 선언된 변수이다.
- ③ 지역변수는 함수 밖에서 참조를 할 수 있다.
- ④ 전역변수는 함수 안에서 참조가 가능하다.
- ⑤ 전역 변수와 지역 변수의 이름이 같을 때 이를 구분해주기 위해 'global' 이라는 특수명령어를 사용한다.

02. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def test(k):  
    print("Input is", k)  
  
k = 100  
test(k)
```

- ① Input is 100
- ② NameError
- ③ IndexError
- ④ ValueError
- ⑤ None

03. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def counter(*args):  
    count = len(args)  
    return count  
  
print(counter(["test", "hello", "oooo"]))
```

- ① 1
- ② 2
- ③ 3
- ④ 4
- ⑤ 0

07. 다음과 같이 코드를 작성했을 때, 실행 결과로 알맞은 것은?

```
def print_hi():  
    print("Hi")
```

- ① Hi
- ② NameError
- ③ hello()
- ④ ValueError
- ⑤ 출력 값 없음