

되 새 김 문 제

전 코딩을 공부하신 여러분!
포기하지 말고 이제김 문제를 통해
실력을 향토해 보세요!

02장의 정답 및 풀이는 403~406쪽에 있습니다.

01 평균 점수 구하기

홍길동 씨의 과목별 점수는 다음과 같다. 홍길동 씨의 평균 점수를 구해 보자.

과목	점수
국어	80
영어	75
수학	55

02 홀수, 짝수 판별하기

자연수 13이 홀수인지, 짝수인지 판별할 수 있는 방법에 대해 말해 보자.

03 주민등록번호 나누기

홍길동 씨의 주민등록번호는 881120-1068234이다. 홍길동 씨의 주민등록번호를 연월일 (YYYYMMDD) 부분과 그 뒤의 숫자 부분으로 나누어 출력해 보자.

```

pin = "881120-1068234"
yyyyymmdd =
num =
print( ) # 연월일 부분 출력
print( ) # 숫자 부분 출력
    
```

문자열을 나누기
이렇게 풀어?



Q4 주민등록번호 인덱싱

주민등록번호 뒷자리의 맨 첫 번째 숫자는 성별을 나타낸다. 주민등록번호에서 성별을 나타내는 숫자를 출력해 보자.

```
pin = "881120-1068234"  
print( )
```

문자열 인덱싱을
이용해 보자.



Q5 문자열 바꾸기

다음과 같은 문자열 a:b:c:d가 있다. 문자열의 replace 함수를 사용하여 a#b#c#d로 바꿔 출력해 보자.

```
a = "a:b:c:d"  
b =  
print(b) ← 문자열 "a#b#c#d" 출력
```

Q6 리스트 역순 정렬하기

[1, 3, 5, 4, 2] 리스트를 [5, 4, 3, 2, 1]로 만들어 보자.

```
a = [1, 3, 5, 4, 2]  
a.  
a.  
print( )
```

← [1, 2, 3, 4, 5]로 변경

← [5, 4, 3, 2, 1]로 변경

리스트의 내장 함수를
쓰면 쉽게 할 수 있네



Q7 리스트를 문자열로 만들기

['Life', 'is', 'too', 'short'] 리스트를 Life is too short 문자열로 만들어 출력해 보자.

```
a = ['Life', 'is', 'too', 'short']
result =
print(result) ← "Life is too short" 출력
```

문자열이기에 합쳐줄
사유가 어떤 세팅이?



Q8 튜플 더하기

(1, 2, 3) 튜플에 값 4를 추가하여 (1, 2, 3, 4)를 만든 후 출력해 보자.

```
a = (1, 2, 3)
a =
print(a) ← (1, 2, 3, 4) 출력
```

허더가서 덧셈자를 쓰면
값이 뭘 수 있겠어?



Q9 딕셔너리의 키

다음과 같은 딕셔너리 a가 있다.

```
>>> a = dict()
>>> a
{}

```

다음 중 오류가 발생하는 경우를 고르고, 그 이유를 설명해 보자.

```
a['name'] = 'python'
a[('a',)] = 'python'
a[[1]] = 'python'
a[250] = 'python'

```

Q10 딕셔너리 값 추출하기

딕셔너리 a에서 'B'에 해당하는 값을 추출해 보자.

```
a = {'A':90, 'B':80, 'C':70}
result =
print(a) ← {'A':90, 'C':70} 출력
print(result) ← 80 출력
```

딕셔너리 a의 pop 함수를
이용해 보자.



Q11 리스트에서 중복 제거하기

a 리스트에서 중복 숫자를 제거해 보자.

```
a = [1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 5]
aSet =
b =
print(b) ← [1, 2, 3, 4, 5] 출력
```

집합 자료형의 주요 특징은
중복될 수 없다는 특징을
이용해 볼까?



Q12 파이썬 변수

파이썬은 다음처럼 동일한 값에 여러 개의 변수를 선언할 수 있다. 다음과 같이 a, b 변수를 선언한 후 a의 두 번째 요소를 변경하면 b 값은 어떻게 될까? 그리고 이런 결과가 나타나는 이유를 설명해 보자.

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

되/새/김/문/제

간혹으로 공부하신 여러분!
포기하지 말고 되새김 문제를 통해
실력을 정복해 보세요!

■ 03장의 정답 및 풀이는 407~408쪽에 있습니다.

01 조건문의 참과 거짓

다음 코드의 결과값은 무엇일까?

```
a = "Life is too short, you need python"

if "wife" in a: print("wife")
elif "python" in a and "you" not in a: print("python")
elif "shirt" not in a: print("shirt")
elif "need" in a: print("need")
else: print("none")
```

02 3의 배수의 합 구하기

while 문을 사용해 1부터 1000까지의 자연수 중 3의 배수의 합을 구해 보자.

```
result = 0
i = 1
while i <= 1000:
    if           :
        result += i
    i += 1

print(result) ← 166833 출력
```

3으로 나누어떨어지는
수가 3의 배수야



03 별 표시하기

while 문을 사용하여 다음과 같이 별(*)을 표시하는 프로그램을 작성해 보자.

```
i = 0
while True:
    i += 1  <-- while 문을 수행할 때 1씩 증가
    if      :break  <-- i 값이 5를 초과하면 while 문을 벗어난다.
    print(      )  <-- i 값의 개수만큼 '*'을 출력
```

실행 결과

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

04 1부터 100까지 출력하기

for 문을 사용해 1부터 100까지의 숫자를 출력해 보자.

```
>>> for i in
...     print(i)
...
1
2
3
4
5
6
7
8
9
10
(...생략...)
```

05 평균 점수 구하기

A 학급에 총 10명의 학생이 있다. 이 학생들의 중간고사 점수는 다음과 같다.

```
[70, 60, 55, 75, 95, 90, 80, 80, 85, 100]
```

for 문을 사용하여 A 학급의 평균 점수를 구해 보자.

```
A = [70, 60, 55, 75, 95, 90, 80, 80, 85, 100]
total = 0
for score in A:
    total +=
average =
print(average)
```

모든 학생의 점수를 더해
총 학생 수로 나누면 평균
점수를 구할 수 있겠네



06 리스트 컴프리헨션 사용하기

다음 소스 코드는 리스트의 요소 중에서 홀수만 골라 2를 곱한 값을 result 리스트에 담는 예제이다.

```
numbers = [1, 2, 3, 4, 5]
result = []
for n in numbers:
    if n % 2 == 1:
        result.append(n * 2)
```

이 코드를 리스트 컴프리헨션을 사용하여 표현해 보자.

```
numbers = [1, 2, 3, 4, 5]
result =
print(result)
```

되/새/김/문/제

간혹 코딩으로 공부하신 여러분이
모기까지 알고 되새김 문제를 통해
실력을 정복해 보세요!

■ 04장의 정답 및 풀이는 409~412쪽에 있습니다.

01 홀수, 짝수 판별하기

주어진 자연수가 홀수인지, 짝수인지 판별해 주는 함수 `is_odd`를 작성해 보자. `is_odd` 함수는 홀수이면 `True`, 짝수이면 `False`를 리턴해야 한다.

```
def is_odd(number):
    if
        return True
    else:
        return False
```

2로 나누었을 때
나머지가 1이면 홀수야!



02 모든 입력의 평균값 구하기

입력으로 들어오는 모든 수의 평균값을 계산해 주는 함수를 작성해 보자. 단, 입력으로 들어오는 수의 개수는 정해져 있지 않다.

```
def avg_numbers(
):
    result = 0
    for i in args:
        result += i
    return
```

평균값을 구할 때
len 함수를 써 보자!



`avg_numbers(1, 2)` ← 1.5 출력

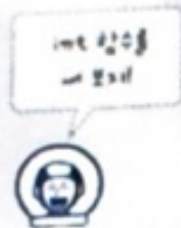
`avg_numbers(1,2,3,4,5)` ← 3.0 출력

03 프로그램 오류 수정하기 1

다음은 2개의 숫자를 입력받아 더한 후에 리턴하는 프로그램이다.

```
input1 = input("첫 번째 숫자를 입력하세요: ")
input2 = input("두 번째 숫자를 입력하세요: ")

total = input1 + input2
print("두 수의 합은 %s입니다" % total)
```



이 프로그램을 실행해 보자.

```
첫 번째 숫자를 입력하세요: 3
두 번째 숫자를 입력하세요: 6
두 수의 합은 36입니다
```

3과 6을 입력했을 때 9가 아닌 36이라는 결괏값이 출력되었다. 이 프로그램의 오류를 수정해 보자.

04 출력 결과가 다른 것은?

다음 중 출력 결과가 다른 하나를 골라 보자.

- ① `print("you" "need" "python")`
- ② `print("you" + "need" + "python")`
- ③ `print("you", "need", "python")`
- ④ `print("".join(["you", "need", "python"]))`

05 프로그램 오류 수정하기 2

다음은 파일(test.txt)에 "Life is too short" 문자열을 저장한 후 다시 그 파일을 읽어 출력하는 프로그램이다.

```
f1 = open("test.txt", 'w')
f1.write("Life is too short")

f2 = open("test.txt", 'r')
print(f2.read())
```

이 프로그램은 우리가 예상한 'Life is too short'라는 문장을 출력하지 않는다. 우리가 예상한 값을 출력할 수 있도록 프로그램을 수정해 보자.

06 사용자 입력 저장하기

사용자의 입력을 파일(test.txt)에 저장하는 프로그램을 작성해 보자. 단, 프로그램을 다시 실행하더라도 기존에 작성한 내용을 유지하고 새로 입력한 내용을 추가해야 한다.

```
user_input = input("저장할 내용을 입력하세요:")
f = open('test.txt', 'a') ← 내용을 추가하기 위해 'a'를 사용
f.write(user_input)
f.write("\n") ← 입력한 내용을 줄 단위로 구분하기 위해 줄 바꿈 문자 삽입
f.close()
```

07 파일의 문자열 바꾸기

다음과 같은 내용을 지닌 test.txt가 있다. 이 파일의 내용 중 "java"라는 문자열을 "python"으로 바꾸어 저장해 보자.

```
Life is too short  
you need java
```

```
f = open('test.txt', 'r')  
body =   
f.close()  
body =   
f = open('test.txt',   
f.write(body)  
f.close()
```

← test.txt의 내용을 body 변수에 저장

← body 문자열에서 "java"를 "python"으로 변경

) ← 파일을 쓰기 모드로 다시 실행

replace 함수를 써 볼까?



08 입력값을 모두 더해 출력하기

다음과 같이 실행할 때 입력값을 모두 더해 출력하는 스크립트(C:\doit\myargv.py)를 작성해 보자.

```
C:\> cd doit  
C:\doit> python myargv.py 1 2 3 4 5 6 7 8 9 10  
55
```

sys.argv를 쓰면
되지 않을까?



되/새/김/문/제

간혹 혼동으로 공부하신 여러분!
보가하지 말고 되새김 문제를 통해
실력을 정교하게 보세요!

■ 05장의 정답 및 풀이는 413~419쪽에 있습니다.

01 클래스 상속받고 메서드 추가하기 1

다음은 Calculator 클래스이다.

```
class Calculator:
    def __init__(self):
        self.value = 0

    def add(self, val):
        self.value += val
```

이 클래스를 상속하는 UpgradeCalculator를 만들고 값을 뺄 수 있는 minus 메서드를 추가해 보자. 즉, 다음과 같이 동작하는 클래스를 만들어야 한다.

```
cal = UpgradeCalculator()
cal.add(10)
cal.minus(7)

print(cal.value) ← 10에서 7을 뺀 3을 출력
```


02 클래스 상속받고 메서드 추가하기 2

객체변수 `value`가 100 이상의 값은 가질 수 없도록 제한하는 `MaxLimitCalculator` 클래스를 만들어 보자. 즉, 다음과 같이 동작해야 한다.

```
cal = MaxLimitCalculator()
cal.add(50) ← 50 더하기
cal.add(60) ← 60 더하기

print(cal.value) ← 100 출력
```

단, 반드시 다음과 같은 `Calculator` 클래스를 상속해서 만들어야 한다.

```
class Calculator:
    def __init__(self):
        self.value = 0

    def add(self, val):
        self.value += val
```

03 참과 거짓 예측하기

다음 결과를 예측해 보자.

```
>>> all([1, 2, abs(-3)-3])
```

```
>>> chr(ord('a')) == 'a'
```

Q4 음수 제거하기

filter와 lambda를 사용하여 리스트 [1, -2, 3, -5, 8, -3]에서 음수를 모두 제거해 보자.

Q5 16진수를 10진수로 변경하기

234라는 10진수의 16진수는 다음과 같이 구할 수 있다.

```
>>> hex(234)
'0xea'
```

내장 함수 hex를 활용
하면 쉬울 거야!



이번에는 반대로 16진수 문자열 '0xea'를 10진수로 변경해 보자.

Q6 리스트 항목마다 3 곱하여 리턴하기

map과 lambda를 사용하여 [1, 2, 3, 4] 리스트의 각 요소값에 3이 곱해진 리스트 [3, 6, 9, 12]를 만들어 보자.

Q7 최댓값과 최솟값의 합

다음 리스트의 최댓값과 최솟값의 합을 구해 보자.

```
[-8, 2, 7, 5, -3, 5, 0, 1]
```

08 소수점 반올림하기

17 / 3의 결과는 다음과 같다.

```
>>> 17 / 3
5.666666666666667
```

위와 같은 결과값 5.666666666666667을 소숫점 4자리까지만 반올림하여 표시해 보자.

09 디렉터리 이동하고 파일 목록 출력하기

os 모듈을 사용하여 다음과 같이 동작하도록 코드를 작성해 보자.

- ① C:\doit 디렉터리로 이동한다.
- ② dir 명령을 실행하고 그 결과를 변수에 담는다.
- ③ dir 명령의 결과를 출력한다.

010 파일 확장자가 .py인 파일만 찾기

glob 모듈을 사용하여 C:\doit 디렉터리의 파일 중 확장자가 .py인 파일만 출력하는 프로그램을 작성해 보자.

011 날짜 표시하기

time 모듈을 사용하여 현재 날짜와 시간을 다음과 같은 형식으로 출력해 보자.

```
2018/04/03 17:20:32
```

Q12 로또 번호 생성하기

random 모듈을 사용하여 로또 번호(1~45 사이의 숫자 6개)를 생성해 보자(단, 중복된 숫자가 있으면 안 됨).

Q13 누나는 영철이보다 며칠 더 먼저 태어났을까?

영철이 누나의 생일은 1995년 11월 20일이고 영철이의 생일은 1998년 10월 6일이다. 영철이 누나는 영철이보다 며칠 더 먼저 태어났을까?

날짜를 비교할 때는
datetime.date
함수를 사용하면



Q14 기록순으로 정렬하기

다음은 1학년 3반 학생들의 100m 달리기 기록이다.

```
data = [('윤서현', 15.25),  
        ('김예지', 13.31),  
        ('박예원', 15.34),  
        ('송순자', 15.57),  
        ('김시우', 15.48),  
        ('배숙자', 17.9),  
        ('전정웅', 13.39),  
        ('김혜진', 16.63),  
        ('최보람', 17.14),  
        ('한지영', 14.83),  
        ('이성호', 17.7),  
        ('김옥순', 16.71),  
        ('황민지', 17.65),  
        ('김영철', 16.7),  
        ('주병철', 15.67),  
        ('박상현', 14.16),  
        ('김영순', 14.81),  
        ('오지아', 15.13),  
        ('윤지은', 16.93),  
        ('문재호', 16.39)]
```


기록순으로 data를 출력해 보자.

◆ operator.itemgetter를 사용해 보자.

남자는 operator
itemgetter 함수를
써 보자!



Q15 형소 당번 2명 뽑기

다음 4명의 학생 중 형소 당번 2명을 뽑을 수 있는 경우의 수를 모두 나열하시오.

['나지혜', '정성민', '윤지현', '김정숙']

Q16 문자열 나열하기

"abcd" 문자열을 나열하는 경우의 수를 다음과 같이 모두 출력하시오.

abcd, abdc, adcb, (...생략...)

경우의 수를 구할 때는
itertools의 함수 중
하나를 쓰면 되겠지요?



Q17 5명에게 할 일 부여하기

다음 5명이 있다.

['김승현', '김진호', '강훈자', '이예준', '김현주']

그리고 해야 할 일은 다음처럼 3가지가 있다.

['청소', '빨래', '설거지']

5명을 무작위로 섞어 앞의 3명에게 차례로 해야 할 일인 ["청소", "빨래", "설거지"]를 지정하고 나머지 2명에게는 "휴식"을 지정할 수 있는 프로그램을 작성하시오.

Q18 벽에 타일 붙이기

가로와 세로의 길이는 200cm이고 세로의 길이는 80cm인 벽이 있다. 이 벽에 되도록 큰 정사각형 모양의 타일을 붙이려고 한다. 이때 붙이려는 타일 한 선의 길이와 붙이는 데 필요한 타일의 개수를 구하시오.

문제 풀이
힌트를 보세요



01 문자열 바꾸기 ★☆☆

다음과 같은 문자열이 있다.

```
a:b:c:d
```

문자열의 `split`과 `join` 함수를 사용해 문자열을 다음과 같이 고치시오.

```
a#b#c#d
```

02 딕셔너리 값 추출하기 ★☆☆

다음은 딕셔너리 `a`에서 'C'라는 key에 해당하는 value를 출력하는 프로그램이다.

```
>>> a = {'A':90, 'B':80}
>>> a['C']
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 'C'
```

`a` 딕셔너리에는 'C'라는 key가 없으므로 위와 같은 오류가 발생한다. 'C'에 해당하는 key 값이 없을 경우, 오류 대신 70을 얻을 수 있도록 수정하시오.

03 리스트의 더하기와 `extend` 함수 ★★☆☆

다음과 같은 리스트 `a`가 있다.

```
>>> a = [1, 2, 3]
```

리스트 a에 [4, 5]를 +를 사용해 더한 결과는 다음과 같다.

```
>>> a = [1, 2, 3]
>>> a = a + [4, 5]
>>> a
[1, 2, 3, 4, 5]
```

리스트 a에 [4, 5]를 extend를 사용해 더한 결과는 다음과 같다.

```
>>> a = [1, 2, 3]
>>> a.extend([4, 5])
>>> a
[1, 2, 3, 4, 5]
```

+를 사용하여 더한 것과 extend한 것의 차이점이 있을까? 있다면 그 차이점을 설명하시오.

Q4 리스트 총합 구하기 ★★☆☆

다음은 A학급 학생의 점수를 나타내는 리스트이다. 다음 리스트에서 50점 이상 점수의 총합을 구하시오.

```
A = [20, 55, 67, 82, 45, 33, 90, 87, 100, 25]
```

Q5 피보나치 함수 ★★★

첫 번째 항의 값이 0이고 두 번째 항의 값이 1일 때 이후에 이어지는 항은 이전의 두 항을 더한 값으로 이루어지는 수열을 '피보나치 수열'이라고 한다.

```
0, 1, 1, 2, 3, 5, 8, 13, ...
```

입력을 정수 n으로 받았을 때 n항 이하까지의 피보나치 수열을 출력하는 함수를 작성하시오.

Q6 숫자의 총합 구하기 ★☆☆

사용자에게 다음과 같은 숫자를 입력받아 입력받은 숫자의 총합을 구하는 프로그램을 작성하시오. 숫자는 '.'로 구분하여 입력한다).

```
65,45,2,3,45,8
```

Q7 한 줄 구구단 ★☆☆

사용자에게 2~9의 숫자 중 하나를 입력받아 해당 숫자의 구구단을 한 줄로 출력하는 프로그램을 작성하시오.

실행 결과

```
구구단을 출력할 숫자를 입력하세요(2~9): 2
```

```
2 4 6 8 10 12 14 16 18
```

Q8 파일을 읽어 역순으로 저장하기 ★☆☆

다음과 같은 내용의 파일 abc.txt가 있다.

```
AAA
```

```
BBB
```

```
CCC
```

```
DDD
```

```
EEE
```

이 파일의 내용을 다음과 같이 역순으로 바꾸어 저장하시오.

```
EEE
```

```
DDD
```

```
CCC
```

```
BBB
```

```
AAA
```

09 평균값 구하기 ★★★

다음과 같이 총 10줄로 이루어진 sample.txt가 있다. sample.txt의 숫자 값을 모두 읽어 총합과 평균값을 구한 후 평균값을 result.txt에 쓰는 프로그램을 작성하시오.

```
70
60
35
75
95
90
80
80
85
100
```

010 계산기 만들기 ★★☆☆

다음과 같이 동작하는 클래스 Calculator를 작성하시오.

```
>>> cal1 = Calculator([1, 2, 3, 4, 5])
>>> cal1.sum() ← 합계
15
>>> cal1.avg() ← 평균
3.0
>>> cal2 = Calculator([6, 7, 8, 9, 10])
>>> cal2.sum() ← 합계
40
>>> cal2.avg() ← 평균
8.0
```

011 모듈을 사용하는 방법 ★★★

C:\doit 디렉터리에 mymod.py 파이썬 모듈이 있다고 가정해 보자. 명령 프롬프트 창에서 파일을 열어서 이 모듈을 import해서 사용할 수 있는 방법을 모두 기술하시오(즉, 다음과 같이 import mymod 수행할 때 오류가 없어야 한다).

```
>>> import mymod
>>>          ← 오류 없음
```

012 오류와 예외 처리 ★☆☆

다음 코드의 실행 결과를 예측하고 그 이유에 대해 설명하시오.

```
result = 0

try:
    [1, 2, 3][3]
    "a" + 1
    4 / 0
except TypeError:
    result += 1
except ZeroDivisionError:
    result += 2
except IndexError:
    result += 3
finally:
    result += 4

print(result)
```

Q13 DashInsert 함수 ★★☆☆

DashInsert 함수는 숫자로 구성된 문자열을 입력받은 뒤 문자열 안에서 홀수가 연속되면 두 수 사이에 '-'를 추가하고, 짝수가 연속되면 '*'를 추가하는 기능을 가지고 있다. DashInsert 함수를 완성하시오.

입력 예시: 4546793

출력 예시: 454*67-9-3

Q14 문자열 압축하기 ★★☆☆

문자열을 입력받아 같은 문자가 연속적으로 반복되는 경우, 그 반복 횟수를 표시해 문자열을 압축하여 표시하시오.

입력 예시: aaabbcccccca

출력 예시: a3b2c6a1

Q15 Duplicate Numbers 함수 ★★☆☆

0~9의 문자로 된 숫자를 입력받았을 때 이 입력값이 0~9의 모든 숫자를 각각 한 번씩만 사용한 것인지 확인하는 함수를 작성하시오.

입력 예시: 0123456789 01234 01234567890 6789012345 012322456789

출력 예시: True False False True False

016 모스 부호 해독 ★★★

문자열 형식으로 입력받은 모스 부호 .(dot)과 -(dash)를 해독하여 영어 문장으로 출력하는 프로그램을 작성하시오.

글자와 글자 사이는 공백 1개, 단어와 단어 사이는 공백 2개로 구분한다.
예를 들어 다음 모스 부호는 'HE SLEEPS EARLY'로 해석해야 한다.

.... -.-. -.-. -.-. -.-.

모스 부호 규칙 표

문자	부호	문자	부호
A	.-	N	-.
B	-...	O	---
C	-.-.	P	.-.-
D	-. .	Q	--.-
E	.	R	.-.
F	..-.	S	...
G	--.	T	-
H	U	..-
I	..	V	...-
J	.-...	W	.-.
K	-. -	X	-. -.
L	.-. .	Y	-. -.
M	--	Z	--. .

017 정규식 — 기초 메타 문자 ☆☆☆

다음 중 정규식 a.](3,)b와 매치되는 문자열은 무엇일까?

- ① acccb
- ② a....b
- ③ aaab
- ④ a.cccb

Q18 정규식 — 문자열 검색 ★★☆☆

다음 코드의 결과값은 무엇일까?

```
>>> import re
>>> p = re.compile("[a-z]+")
>>> m = p.search("5 python")
>>> m.start() + m.end()
```

Q19 정규식 — 그룹핑 ★★★

다음과 같은 문자열에서 휴대폰 번호 뒷자리인 숫자 4개를 #####로 바꾸는 프로그램을 정규식을 사용하여 작성하시오.

```
===
park 010-9999-9988
kim 010-9909-7789
lee 010-8789-7768
===
```

Q20 정규식 — 전방 탐색 ★★★

다음은 이메일 주소를 나타내는 정규식이다. 이 정규식은 park@naver.com, kim@daum.net, lee@myhome.co.kr 등과 매치된다. 긍정형 전방 탐색 기법을 사용하여 .com, .net이 아닌 이메일 주소는 제외시키는 정규식을 작성하시오.

```
.*[0].*[*].*5
```