

## 예시 문제 1. 이상한 덧셈 - ex01.py

- 리스트 numbers에 담겨있는 숫자들 중, 양의 정수들의 합을 반환하는 `positive_sum` 함수를 작성하시오.

```
def positive_sum(numbers):
    """
    여기에 코드를 작성하시오.
    numbers는 숫자들이 담긴 리스트입니다.
    numbers에 담긴 숫자들 중, 양의 정수들의 합을 반환합니다.
    """
```

```
# 예시 입력 및 출력
print(positive_sum([1, -4, 7, 12]))
print(positive_sum([-1, -2, -3, -4]))
```

20  
0

## 예시 문제 2. 문자열 계산하기 - ex02.py

- 아래와 같이 수식 문자열이 주어졌을 때, 이를 계산을 해주는 `calc` 함수를 작성하시오.
- 이 계산기는 덧셈과 뺄셈만 가능하다.

```
def calc(equation):
    """
    아래에 코드를 작성하십시오.
    equation은 덧셈 뺄셈으로 이루어진 수식으로 된 문자열입니다.
    계산된 결과를 정수로 반환하십시오.
    """
```

[illegible]

1  
1212  
0

## 예시 문제 3. 도형 만들기 - ex03.py

- 파이썬 클래스를 활용하여 꼭지점과 원을 표현하시오.
- `Point` 클래스에 대한 명세는 다음과 같습니다.
- 인스턴스가 생성될 때, 정수 두 개(좌표 값)를 인자로 받아 생성됩니다. 각각 인스턴스 변수명은 `x`, `y` 입니다.

인스턴스 변수	타입
x	int
y	int

메소드명	인자	반환값 (return)	비고
(생성자)	x좌표, y 좌표	없음	인자로 들어온 값을 각각 인스턴스 변수 <code>x</code> , <code>y</code> 에 할당합니다.
<code>__str__</code>	없음	문자열	아래의 예시와 같이 다음의 문자열을 반환합니다. <code>Point:(x,y)</code>

- `__str__` 메소드에서 문자열 반환시 `x`, `y`는 아래와 같이 인스턴스 변수 값으로 출력되어야 합니다.

```
# 예) 좌표 (3, 5)의 점
p1 = Point(3, 5)
print(p1)
#=> Point:(3,5)
```

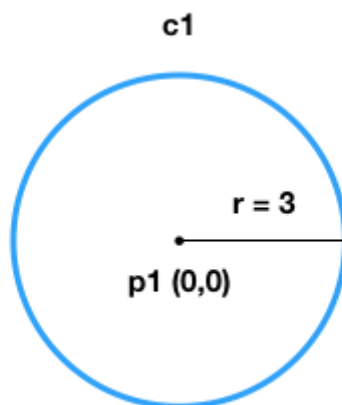
- `Circle` 클래스에 대한 명세는 다음과 같습니다.
- 인스턴스가 생성될 때, 원의 중심과 반지름을 인자로 받아 생성됩니다. 인스턴스 변수명은 `center`와 `r` 입니다.

인스턴스 변수	타입
center	Point 클래스 인스턴스
r	int

메소드명	인자	반환값 (return)	비고
(생성자)	Point 클래스 인스턴스, 반지름	없음	인자로 들어온 값을 인스턴스 변수 <code>center</code> 와 <code>r</code> 에 할당합니다.
<code>get_area</code>	없음	면적	면적을 계산하여 반환합니다. (원주율은 3.14)
<code>get_perimeter</code>	없음	둘레값	둘레를 계산하여 반환합니다. (원주율은 3.14)
<code>get_center</code>	없음	(x, y)	tuple : 원의 중점(cener)의 x 값과 y값
<code>__str__</code>	없음	문자열	아래의 예시와 같이 다음의 문자열을 반환합니다. <code>Circle:(x,y),r:r</code>

- `__str__` 메소드에서 문자열 반환시 x, y, r는 아래와 같이 인스턴스 변수 값으로 출력되어야 합니다.

아래의 예시의 c1은 다음과 같이 구성됩니다.



# 아래의 코드를 실행하였을 때 아래와 같이 출력되어야 합니다.

```
p1 = Point(0, 0)
c1 = Circle(p1, 3)
print(c1.get_area())
print(c1.get_perimeter())
print(c1.get_center())
print(c1)
p2 = Point(4, 5)
c2 = Circle(p2, 1)
print(c2.get_area())
print(c2.get_perimeter())
print(c2.get_center())
print(c2)
```

28.26

18.84

(0, 0)

circle:(0,0),r:3

3.14

6.28

(4, 5)

circle:(4,5),r:1