

data structure algorithm

made in fastcampus lecture by bools

• operator 모듈

• 모듈이란, 누군가 이미 만들어 놓은 훌륭한 기능(우리는 공짜로 사용할 수 있다.)

수학 연산 관련 모듈

난수 관련 모듈

연산자 관련 모듈

••••

• operator 모듈

산술 연산자 관련 함수

연산자	operator 함수
+	operator.add()
-	operator.sub()
*	operator.mul()
/	operator.truediv()
%	operator.mod()
//	operator.floordiv()
**	operator.pow()

```
num1 = 8
num2 = 3
print('{} + {} : {}'.format(num1, num2, operator.add(num1, num2)))
print('{} - {} : {}'.format(num1, num2, operator.sub(num1, num2)))
print('{} * {} : {}'.format(num1, num2, operator.mul(num1, num2)))
print('{} / {} : {}'.format(num1, num2, operator.truediv(num1, num2)))
print('{} / {} : {}'.format(num1, num2, operator.mod(num1, num2)))
print('{} // {} : {}'.format(num1, num2, operator.floordiv(num1, num2)))
print('{} // {} : {}'.format(num1, num2, operator.pow(num1, num2)))
```

바이쐰 사료구조/알고리즘

• operator 모듈

• operator 모듈

비교 연산자 관련 함수

연산자	operator 함수
==	operator.eq()
!=	operator.ne()
>	operator.gt()
>=	operator.ge()
<	operator.lt()
<=	operator.le()

```
num1 = 8
num2 = 3
print('{} == {} : {}'.format(num1, num2, operator.eq(num1, num2)))
print('{} != {} : {}'.format(num1, num2, operator.ne(num1, num2)))
print('{} > {} : {}'.format(num1, num2, operator.gt(num1, num2)))
print('{} >= {} : {}'.format(num1, num2, operator.ge(num1, num2)))
print('{} < {} : {}'.format(num1, num2, operator.lt(num1, num2)))
print('{} <= {} : {}'.format(num1, num2, operator.le(num1, num2)))</pre>
```

논리 연산자 관련 함수

연산자	operator 함수
and	operator.and_()
or	operator.or_()
not	operator.not_()

```
flag1 = True
flag2 = False
print('{} and {} : {}'.format(flag1, flag2, operator.and_(flag1, flag2)))
print('{} or {} : {}'.format(flag1, flag2, operator.or_(flag1, flag2)))
print('not {} : {}'.format(flag1, operator.not_(flag1)))
```

• 실습

이전 시간에 실습했던 백신 접종 대상자 출력 코드를 operator 모듈을 이용해서 변경해 보자.

[이전 실습]

'백신 접종 대상자는 20세 미만 또는 65세 이상자에 한합니다.'를 논리연산자를 이용해서 코딩해 보자.

```
age = int(input('나이 입력 : '))
vaccine = (age < 20) or (age >= 65)
print('age: {}, result: {}'.format(age, vaccine))

age = int(input('나이 입력 : '))
vaccine = operator.or_(operator.lt(age, 20), operator.ge(age, 65))
print('age: {}, result: {}'.format(age, vaccine))
```

• 실습

random과 operator 모듈을 사용해서 10부터 100사이의 난수 중 십의 자리와 일의 자리가 각각 3의 배수인지 판단하는 코드를 작성해 보자.

```
import random
rInt = random.randint(10, 100)
num10 = operator.floordiv(rInt, 10)
                                                                             난수 : 62
num1 = operator.mod(rInt, 10)
                                                                             십의 자리 : 6
                                                                             일의 자리 : 2
print('난수 : {}'.format(rInt))
                                                                             십의 자리는 3의 배수이다. : True
print('십의 자리 : {}'.format(num10))
                                                                             일의 자리는 3의 배수이다. : False
print('일의 자리 : {}'.format(num1))
print('십의 자리는 3의 배수이다. : {}'.format(operator.mod(num10, 3) == 0))
print('일의 자리는 3의 배수이다. : {}'.format(operator.mod(num1, 3) == 0))
```