

Day 9. 참과 거짓 그리고 논리 연산자

앞에서 불 자료형과 비교 연산자에 대해 알아보았습니다.

"사자가 동물이고 고양이가 동물이라면 나는 '맞아!'라고 소리지를꺼야."라는 문장이 있다면, 이 사람은 "맞아"라고 소리를 지를 것입니다. 왜냐면 사자도 동물이고 고양이도 동물이라는것은 모두 참이기 때문입니다. "~이고"라는 표현은 앞과 뒤가 모두 참일때만 참이기 때문입니다. 이런 비교문을 논리 연산자라고 말합니다.

"사자가 동물이거나 수박이 동물이라면 나는 "맞아!"라고 소리칠꺼야." 라는 문장이 있다면 이 사람은 역시 "맞아"라고 소리지를 것입니다. "~이거나"라는 표현은 앞이나 뒤 표현중에 하나라도 참이라면 참이기 때문입니다.

이렇게 프로그래밍에서도 ~이고 , ~이거나 와 같은 표현을 나타낼 수 있는 방법이 있습니다. 이를 논리 연산자라고 하고 논리연산자에는 and, or, not이 있습니다.

예를 들어 True and True라는 식의 결과는 파이썬에선 True가 됩니다. True or False는 파이썬에선 True가 됩니다.

and와 or는 이항 연산자라고 해서, and와 or앞 뒤로 값이 있어야 합니다. 앞 뒤로 값이 어떠냐에 따라서 다음과 같은 결과가 나오게 됩니다.

p1	p2	p1 and p2
True	True	True
True	False	False
False	True	False
False	False	False

and 연산자는 비교하는 두개의 값이 모두 True일때만 True가 나옵니다.

p1	p2	p1 or p2
True	True	True
True	False	True
False	True	True
False	False	False

or 연산자는 비교하는 두개의 값 중에 하나라도 True가 있으면 True가 나옵니다.

not은 단항연산자라고 해서 not 뒤에는 값이 하나만 와야 합니다. not뒤에 어떤 값이 나오냐에 따라서 결과는 다음과 같습니다.

p1	not p1
True	False
False	True

not 연산자는 부정연산자로 하여 True를 False로 False를 True로 바꿉니다.

예제를 통해 알아보도록 하겠습니다.

File메뉴 - New File 메뉴를 선택합니다. 새로운 창이 뜨면 다음과 같은 코드를 입력합니다. 소스코드가 기니 한번에 모두 입력해서 실행하지 말고 조금씩 나눠서 입력하고 실행해 나가도록 합시다. 조금씩 나눠서 입력하기 편하라고 "----- 1 -----", "----- 2 -----" 와 같은 값을 출력하여 단락을 나눠놨습니다.

day09.py
print("----- 1 -----")
p9 = True and True

```
p10 = True and False
```

```
print(p9)
```

```
print(p10)
```

```
print("----- 2 -----")
```

```
p11 = True
```

```
p12 = True
```

```
p13 = False
```

```
p14 = False
```

```
p15 = p11 and p12
```

```
p16 = p11 and p13
```

```
p17 = p11 or p13
```

```
p18 = p13 or p14
```

```
p19 = not p11
```

```
p20 = not p13
```

```
print(p15)
```

```
print(p16)
```

```
print(p17)
```

```
print(p18)
```

```
print(p19)
```

```
print(p20)
```

위의 코드를 실행하면 다음과 같은 결과가 나옵니다.

```
----- 1 -----
```

```
True
```

```
False
```

```
----- 2 -----
```

```
True
```

```
False
```

```
True
```

```
False
```

```
False
```

```
False
```

```
p9 = True and True
```

True 와 True를 and연산을 하게 되면 True가 됩니다. 그 결과는 p9에 저장됩니다. and연산의 경우 앞뒤로 True일 경우에만 True가 나온다는거 기억하세요.

```
p10 = True and False
```

True와 False를 and연산을 하게 되면 False가 됩니다. 그 결과는 p10에 저장됩니다.

```
p11 = True
```

p12 = True

p13 = False

p14 = False

p15 = p11 and p12

and의 앞뒤로 True일 경우에만 True가 나온다고 했습니다. p11과 p12는 모두 True값을 가지기 때문에 그 결과는 True가 나오고 p15에 저장됩니다.

p16 = p11 and p13

p13의 값은 False입니다. 즉 p11 and p13의 결과는 False가 되고 그 결과는 p16에 저장됩니다.

p17 = p11 or p13

or의 앞뒤로 True가 하나라도 있다면 True가 됩니다. p11이 True이기 때문에 p13에 어떤 값이 있어도 그 결과는 True가 됩니다. p17에는 True가 저장됩니다.

p18 = p13 or p14

p13과 p14모두 False기 때문에 18에는 False가 저장됩니다.

p19 = not p11

not은 부정연산자라고 해서 p11의 값을 반대로 바꿉니다. p11은 True니깐 반대로 바뀌어 False가 되고 그 값은 p19에 저장됩니다.

p20 = not p13

p13의 값은 False니깐 not에 의해 반대로 바뀌어 True가 되고 그 결과는 p20에 저장됩니다.

출력결과와 자신이 예상한 결과가 같은지 확인해 보세요.

숙제)

p200 = true

$p_{201} = \text{false}$

$p_{202} = p_{200} \text{ and } p_{200}$

$p_{203} = p_{201} \text{ or } p_{201}$

$p_{204} = \text{not } p_{203}$

위의 문장을 실행하면 결과가 무엇이 나올지 예상해보고, 실제 프로그램을 작성해서 본인이 생각한 것과 값이 같은지 확인해보세요.