

조건문 (if else)

들어가기

스무고개는 하나의 정답을 맞추기 위해 트리 구조를 만든다면, 컴퓨터의 결정 트리는 **Yes** 또는 **No** 각각의 경우에 맞춰 컴퓨터가 동작할 수 있게 프로그래밍 하는 것이다.

이를 흐름제어 **Control Flow** 라고 한다.

학습 목표

조건부 실행 **Conditional Execution** 에 대해 알아보자.

조건부 실행은 우리의 코드가 무언가가 검사를 하거나 결정을 내릴 때 사용한다.

파이썬이 명령문을 실행시키거나 넘어가는 방법을 알아보자.

핵심 키워드

- 조건부 실행
- 들여쓰기
- if
- else

학습하기

if문

if문의 기본적인 형태는 다음과 같다.

```
x = 5
```

```
if x < 10: # if는 예약어이며, 컴퓨터는 if 다음에 나오는 조건문의 True, False 를 판단하게 된다.  
    print("Smaller") # 만약 True인 경우: 콜론(:) 아래로 들여쓰기 된 부분을 실행하게 된다.
```

비교 연산자

조건문의 참 또는 거짓을 판별하기 위해 사용되는 비교 연산자들이 있다. 다음과 같은 비교 연산자들을 사용한다.

연산자	의미	예시
>	x가 y보다 클 때 True, 그 외에는 False	x > y
<	x가 y보다 작을 때 True, 그 외에는 False	x < y
>=	x가 y보다 크거나 같을 때 True, 그 외에는 False	x >= y
<=	x가 y보다 작거나 같을 때 True, 그 외에는 False	x <= y
==	x와 y가 같을 때 True, 그 외에는 False	x == y
!=	x와 y가 다를 때 True, 그 외에는 False	x != y

들여쓰기 (Indentation)

파이썬에서는 들여쓰기를 매우 엄격하게 생각한다.

들여쓰기가 제대로 되어 있지 않다면, 파이썬은 문법 에러를 통해 우리에게 도움을 요청하게 된다.

```

1 File "part2.py", line 4
2   print('Smaller')
3   ^
4 IndentationError: expected an indented block
5

```

통상 들여쓰기는 `tab` 또는 `space 4번` 과 같다.

컴퓨터가 tab을 잘못 인식하는 경우도 있으므로, space 4번을 사용하도록 하자.

단일 if문, if else 문

단일 if문

단일 if문으로 사용하는 경우, 조건문이 참인 경우에만 미리 입력해 놓은 실행코드를 실행하게 된다.

```

x = 5

if x < 10:
    print("Smaller")

```

if else문

1번째 조건문의 조건이 거짓인 경우에 대해 처리하기 위해 우리는 `else` 문을 사용할 수 있다.

즉, 1번째 조건이 거짓 → `else` 문 이하의 실행코드가 실행된다.

```
x = 11

if x < 10:
    print("Smaller")
else:
    print('Bigger')

# 11 < 10 은 False이다.
# 따라서 Bigger이 출력된다.
```



조건문(if, else)를 사용할 때는 주의할 점이 2가지 있다.

1. 조건문 후에 `:` 콜론을 찍어야한다.
2. 조건문이 참일 경우 → 실행할 코드는 `들여쓰기` 를 해야한다.

코드를 실행시켰는데 에러가 난다면, 위 2조건을 제대로 지켰는지 확인해보자.

