

---

# Python 기초 강의

3 Weeks



Cyber-Physical Systems Security

순천향대학교 정보보호학과

김준원([ruskonert@gmail.com](mailto:ruskonert@gmail.com))

2018.04.09

---

# Agenda

❖ 제어문(if)

❖ 반복문(for, while)

❖ Q & A

# 제어문

If의 기본 구조

비교 연산자

다른 조건문

If문을 사용한 사용 예시

If~elif~else문

## ❖ 제어문: 프로그램의 진행 흐름을 바꾸는 문장

- ✓ “만약 돈이 있다면 기차를 타고 가고, 없으면 그냥 걸어간다” (일상 속에 있는 조건문)
  - 일상 속에서도 조건에 따라 다르게 행동하듯이 제어문에서도 조건에 따라서 다르게 작동

## ❖ 키워드 "If"

- **사용방법**
  - C언어와 다르게 괄호({})가 아닌 들여쓰기로 구분한다는 점에서 주의
  - else 키워드는 안 사용해도 되며 비교문이 거짓이면 else 아래 문장들이 작동함

```
if 비교문:
    작동할 문장1
    작동할 문장2
    ...
else
    작동할 문장1
    작동할 문장2
    ...
```

# 비교 연산자

## ❖ 비교문이 참(1) 또는 거짓(0)인지 판단하는 연산자

- ✓ 프로그램의 흐름을 제어하기 위해 필수적으로 사용함
  - "1과 2는 똑같은가? -> 거짓(false)"
  - "2가 1보다 큰가? -> 참(true)"

비교연산자	설명	
$x < y$	x가 y보다 작은가?	
$x > y$	x가 y보다 큰가?	
$x == y$	x와 y가 같은가?	
$x != y$	x와 y가 같은가?	
$x \geq y$	x가 y보다 크거나 같은가?	
$x \leq y$	x가 y보다 작거나 같은가?	
자료형	참	거짓
숫자	0이 아닌 숫자	0
문자열	"abc"	""
리스트	[1,2,3]	[]
튜플	(1,2,3)	()
딕셔너리	{"a":"b"}	{}

## ❖ 리스트, 튜플, 문자열을 위한 조건문

in	not in
x in 리스트	x not in 리스트
x in 튜플	x not in 튜플
x in 문자열	x not in 문자열

◆ 논리 연산자

연산자 (C언어와 비교한 연산자)	설명
x or y (  )	x와 y 둘 중에 하나만 참이면 참이다
x and y (&&)	x와 y 모두 참이어야 참이다
not x (!)	x가 거짓이면 참이다

# If를 이용한 사용 예시

- ❖ “만약 돈이 10,000원보다 많다면 기차를 타고 아니라면 걸어가겠다”

```
money = 1000000000
if money > 10000:
    train()
else:
    walk()
```

- ❖ “만약 x가 10이라면 2를 곱한 값을 아니라면 2로 나눈 값을 대입하세요”

```
global x
x = 8
if x == 10:
    x = x * 2
else:
    x = x / 2
```

- ❖ 만약 name이 “python”이라면 그냥 아무것도 하지 말고 아니라면 “다른 언어구나”라고 출력하기

```
from operator import eq
name = “python”
if eq(name, “python”):
    pass
else:
    print(“다른 언어구나”)
```

# If~elif~else

## ❖ elif

- ✓ 추가 조건문을 필요로 할 때 사용
- ✓ If 조건문이 거짓이면 다음에 수행됨
- ✓ 1개 또는 여러 개의 조건문을 만들 수 있음

```
if 비교문:
    작동할 문장1
    작동할 문장2
...
elif 비교문:
    작동할 문장1
    작동할 문장2
...
elif 비교문2:
    작동할 문장1
    작동할 문장2
...
else:
    작동할 문장1
    작동할 문장2
...
```



---

# 반복문

For문과 while문



❖ 구구단 출력하기 (출력 방식은 자유입니다)

- ✓ 1단계(중): for문을 사용하여 1단부터 9단까지 구구단을 출력하세요.
- ✓ 2단계(중): while문을 사용하여 1단부터 9단까지 구구단을 출력하세요.
- ✓ 3단계(중상): for와 while을 섞어서 1단부터 9단까지 구구단을 출력하세요.
- ✓ 4단계(상): 기존의 코드에서 제어문을 이용하여 홀수 번째는 출력시키지 않도록 하세요.
  - (예를 들어  $2 \times 1 = 2$ ,  $2 \times 2 = 4$ ,  $2 \times 3 = 6$ , ...에서  $2 \times 1 = 2$ ,  $2 \times 3 = 6$ 을 출력시키면 안됩니다)
- ✓ **5단계(최상):** 4단계 소스 코드를 동아리 github 홈페이지에 있는 python-example 레포토리에서 report 브랜치를 만들어 커밋한 후 master 브랜치와 병합하세요.

1단계부터 5단계를 모두 풀지 않아도 좋으나 1단계는 무조건 풀으셔야 합니다.

5단계를 풀은 사람은 개인적으로 말씀해주십시오.

소스 코드 파일(.py)을 ruskonert@gmail.com 또는 [support@cpss.network](mailto:support@cpss.network) 로 4월 18일까지 보내시기 바랍니다.

개인 이메일	<a href="mailto:ruskonert@gmail.com">ruskonert@gmail.com</a>
동아리 이메일	<a href="mailto:support@cpss.network">support@cpss.network</a>
소스 코드 참고	<a href="https://github.com/CPSSOpenSource">github.com/CPSSOpenSource</a>

강의를 들어 주셔서 감사합니다.