

# 연산자

김지성 강사

# 연산자

✓ 파이썬은 다양한 연산자를 지원해준다. 사용 가능한 연산자로는 다음과 같다.

- 사칙 연산

- 계산을 위한 연산자로 덧셈, 뺄셈 등이 포함

- 관계 연산

- 크다, 작다와 같은 관계를 나타내는 연산자

- 논리 연산

- and, or와 같은 조건과 관련된 연산자

## 연산자 - 산술 연산자

✓ 산술 연산자로는 아래 표와 같이 사용한다.

구분	예제	결과
덧셈	$1 + 2$	3
뺄셈	$1 - 2$	-1
곱셈	$5 * 2$	10
나눗셈	$5 / 2$	2.5
나눗셈 (몫)	$5 // 2$	2
나눗셈 (나머지)	$9 \% 2$	1
제곱	$2 ** 3$	8
괄호 (우선순위)	$(2 + 3) * 2$	10

## 연산자 - 산술 연산자

### ✓ 산술 연산자 실습

```
print(2 + 4)    #6
```

```
print(2 - 4)    #-2
```

```
print(2 * 4)    #8
```

```
print(2 / 4)    #0.5
```

```
print(2 % 4)    #2
```

```
print(2 ** 4)   #16
```

## 연산자 - 산술 연산자

### ✓ 산술 연산자 실습

```
a = 1  
b = 10  
c = 2  
d = 10  
print ( (a + b) * (c + d) )
```

## 연산자 - 산술 연산자

✓ 파이썬은 사칙연산을 할 때 축약해서 사용할 수 있다.

○  $a+=1$ 은  $a = a + 1$ 와 같다.

○  $a-=1$ 은  $a = a - 1$ 와 같다.

○  $a*=2$ 은  $a = a * 2$ 와 같다.

○  $a/=2$ 은  $a = a / 2$ 와 같다.

```
a=10
```

```
a+=1  
print("a+=1 :", a)
```

```
a-=1  
print("a-=1 :", a)
```

```
a*=2  
print("a*=2 :", a)
```

```
a/=2  
print("a/=2 :", a)
```

## 연산자 - 관계 연산자

✓ 관계 연산자로는 아래 표와 같이 사용한다.

연산자	설명
$X == Y$	X와 Y는 같다.
$X != Y$	X와 Y는 같지 않다.
$X > Y$	X가 Y보다 크다.
$X < Y$	X가 Y보다 작다.
$X >= Y$	X가 Y보다 크거나 같다.
$X <= Y$	X가 Y보다 작거나 같다.

## 연산자 - 관계 연산자

- ✓ 관계 연산자의 결과는 boolean type인 True/False로 반환된다.

```
x=2
```

```
y=3
```

```
print(x==y)
```

```
print(x!=y)
```

```
print(x>y)
```

```
print(x<y)
```

```
print(x>=y)
```

```
print(x<=y)
```



## 연산자 - 관계 연산자

- ✓ in을 통해서 해당 값이 포함되어 있는지를 알 수 있다.
  - 결과는 마찬가지로 boolean type으로 True/False로 반환된다.

```
str1 = "hello world"
```

```
print('h' in str1)
```

```
print('hello' in str1)
```

```
print('hi' in str1)
```

## 연산자 - 논리 연산자

- ✓ 논리 연산자는 and, or, not 3가지가 존재한다.
  - and(그리고)는 두 조건이 모두 True일 때만 True를 반환
  - or(또는)은 두 조건 중 하나라도 True이면 True를 반환
  - not(부정)은 조건의 bool type을 반대로 반환
- ✓ 이러한 논리 연산자 특징을 이용하여 Short Circuit을 활용할 수 있다.
  - $i=5$ 일 때  $(i < 2 \text{ or } i != 3)$  /  $(i != 3 \text{ or } i < 2)$ 라는 2개의 조건이 있을 때 뒤의 조건문이 더 효율적

## 연산자 - 논리 연산자

- ✓ 논리 연산자 또한 마찬가지로 boolean type으로 반환한다.

```
a = 4
```

```
#a%2 == 0은 a를 2로 나눴을 때 나머지가 0인 경우이다. 짝수를 의미한다.
```

```
#a가 짝수이고 0이 아닌 경우
```

```
print(a%2 == 0 and a != 0)    #True
```

```
#a가 0이거나 4인 경우
```

```
print(a == 0 or a == 4)    #True
```

```
b = False
```

```
#b의 반대
```

```
print(not b)    #True
```

## 연산자 - 우선순위

- ✓ 파이썬에서의 연산자는 우선순위가 존재하는데 특별하게 주의할 것은 없다.
- ✓ 평상시에 우리가 계산하는 방식대로 계산한다고 생각하면 된다. 표는 참고만 해둘 것

종 류	결합규칙	연산자	우선순위
단항 연산자	←	++ -- + - ~ ! (type)	높음
산술 연산자	→	* / %	
	→	+ -	
	→	<< >>	
비교 연산자	→	< > <= >= isinstance	
	→	== !=	
논리 연산자	→	&	낮음
	→	^	
	→		
	→	&&	
	→		
삼항 연산자	→	?:	낮음
대입 연산자	←	= += -= *= /= %= <<= >>= &= ^=  =	

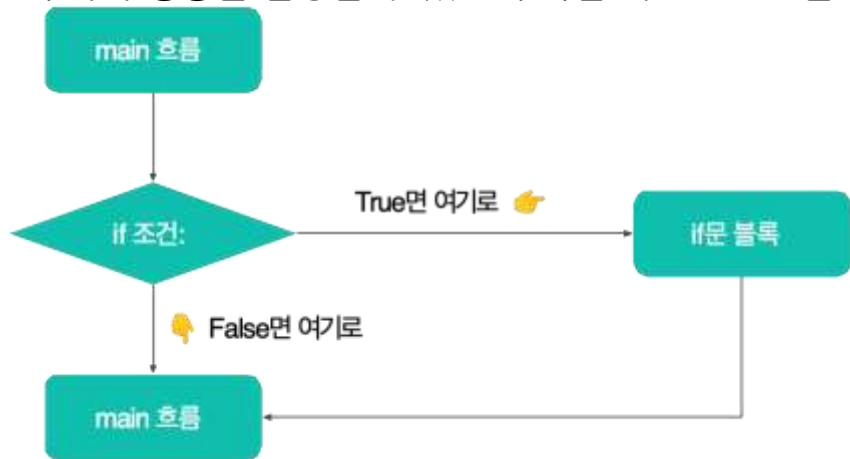
▲ 표 3-4 연산자의 우선순위와 결합규칙

# 조건문

김지성 강사

## 조건문 - if

- ✓ “만약 시간이 늦었다면 택시를 타고 가라”라는 문장이 있다고 가정해보자.
- ✓ 이러한 조건 문장은 특정 조건에 따라 어떤 행동을 해야할지 결정한다.
  - 시간이 늦었다면 -> 택시
- ✓ 파이썬도 특정 조건에 따라 행동을 결정할 수 있는데 이럴 때 if 조건문을 사용한다.



## 조건문 - if

- ✓ 파이썬에서 조건문을 사용하기 위해서는 if 조건문을 사용한다.
- ✓ 이 때 주의할 점은 조건 뒤에는 콜론(:)이 오고 그 아래에 조건에 해당하는 코드 블록은 들여쓰기가 포함되어 있다. 이 들여쓰기를 지키지 않을 경우 코드가 아예 다른 방식으로 실행이 된다.

if 조건:

실행 동작

→ 들여쓰기가 포함되어 있으므로 조건이 참일 때 실행된다.

if 조건:

실행 동작1

실행 동작2

→ 들여쓰기가 없기 때문에 위에서부터 한 줄씩 차례대로 실행되는 코드.

## 조건문 - if

### ✓ 조건문 실습

```
condition = True
if condition:
    print("조건문 내부")
print("조건문 외부")
```

```
condition = False
if condition:
    print("조건문 내부")
print("조건문 외부")
```



## 조건문 - if

### ✓ 조건문 실습

- input() 함수는 사용자로부터 값을 입력받는 함수. 문자열 형태로 받기 때문에 int로 문자열을 정수형으로 변경하여 진행.

```
x = int(input("값을 입력"))  
if x<10:  
    print("x가 10보다 작습니다.")  
  
if x>5 and x<10:  
    print("x는 5보다 크고 10보다 작습니다.")  
  
if x>5 or x<3:  
    print("x는 5보다 크거나 또는 3보다 작습니다.")
```

## 조건문 - if

### ✓ 중첩 조건문 실습

- if문 안에 다시 if문이 있어도 전혀 문제가 없다.

```
x = int(input("값을 입력"))  
if x<10:  
    print("x가 10보다 작습니다.")  
    if x<5:  
        print("x가 5보다 작습니다.")
```

## 조건문 - if

### ✓ if 조건문 정리

- 조건문을 선언할 때 콜론이 없으면 오류 발생

If 조건문  
실행 코드

```
if False == False  
    print("hello")
```

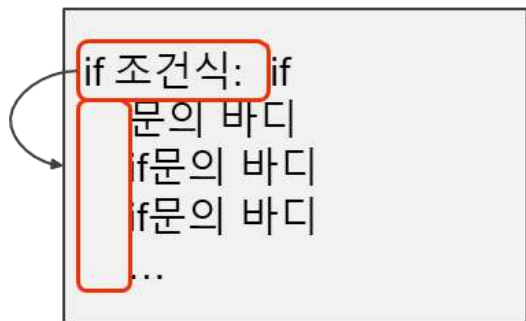
```
File "<ipython-input-7-eedf31ccf33e>", line 1  
    if False == False
```

```
SyntaxError: invalid syntax
```

## 조건문 - if

### ✓ if 조건문 정리

- 조건문을 선언하고 그 바로 뒤에 들여쓰기가 없는 코드가 나오면 오류 발생



들여쓰기는 공백 4칸!

```
if True :  
print("hello")
```

```
File "<ipython-input-2-1a601b2922fd>", line 2  
    print("hello")  
    ^
```

**IndentationError: expected an indented block**

들여쓰기 블록이 예상된다는 에러 메시지를 확인할 수 있습니다.

## 조건문 - if

### ✓ if 조건문 정리

- if 조건문의 실행 범위는 들여쓰기가 끝날 때까지 실행.
- 조건 자체가 False라면 들여쓰기 내의 코드는 실행되지 않음.

if 조건식: if  
 문의 바디  
 if문의 바디  
 if문의 바디  
 들여쓰기 끝

```
if True:
    print("if문의 조건식이", end="")
    print("True일 때만 실행됩니다.")
print("들여쓰기 없는 곳은 늘 실행됩니다.")
```

if문의 조건식이 True일 때만 실행됩니다.  
 들여쓰기 없는 곳은 늘 실행됩니다.

```
if False:
    print("if문의 조건식이", end="")
    print("True일 때만 실행됩니다.")
print("들여쓰기 없는 곳은 늘 실행됩니다.")
```

들여쓰기 없는 곳은 늘 실행됩니다.

## 조건문 - if

### ✓ 조건문 연습 문제

- 사용자로부터 자연수를 입력 받습니다. 그 숫자에서 50을 뺀 값을 출력하세요.
  1. 단 출력 값의 범위는 0 이상입니다.
  2. 50을 뺀 결과가 음수가 될 경우 0을 출력해야 합니다.

실행 결과:

자연수 하나 입력해주세요: 21

0

실행 결과:

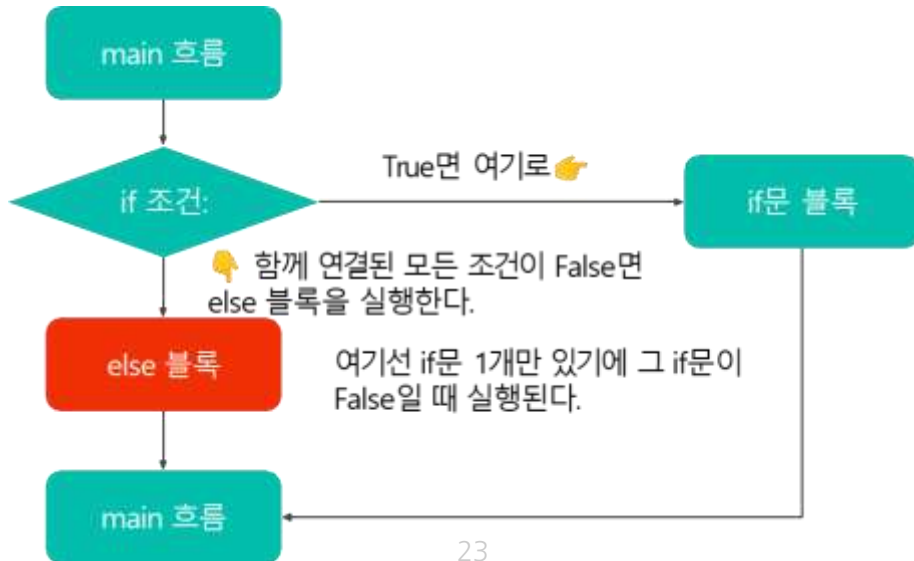
자연수 하나 입력해주세요: 70

20

## 조건문 - if-else

### ✓ if-else 조건문

- if 조건문은 하나의 조건이 참 혹은 거짓일 때의 행동을 제어한다.
- if-else는 조건이 참일 때는 if 내부의 코드가 실행, 거짓일 때는 else 내부의 코드가 실행



## 조건문 - if-else

### ✓ if-else 조건문

- 만약 돈이 없다면 버스를 타고 그렇지 않다면 택시를 타라를 if-else 조건으로 변경 한다면,

```
if 돈이 없다면:  
    버스를 타라  
else :  
    택시를 타라
```

```
if True:  
    if문의 바디  
    if문의 바디  
    if문의 바디  
else:  
    else문의 바디  
    else문의 바디  
    else문의 바디  
    else문의 바디  
들여쓰기 끝
```

```
if False:  
    if문의 바디  
    if문의 바디  
    if문의 바디  
else:  
    else문의 바디  
    else문의 바디  
    else문의 바디  
    else문의 바디  
들여쓰기 끝
```



## 조건문 - if-else

### ✓ if-else 조건문 정리

- else에는 조건식이 필요하지 않음. (if가 거짓일 때 실행하기 때문에 조건이 없다.)
- 즉 if와 else는 항상 반대로 작동이 된다.

True

```
score = 70  
# 60점 이상이면 합격  
if score >= 60:  
    print("합격을 축하합니다.")  
else:  
    print("아쉽게도 불합격입니다.")  
print("안내를 종료합니다.")
```

합격을 축하합니다.  
안내를 종료합니다.

False

```
score = 50  
# 60점 이상이면 합격  
if score >= 60:  
    print("합격을 축하합니다.")  
else:  
    print("아쉽게도 불합격입니다.")  
print("안내를 종료합니다.")
```

아쉽게도 불합격입니다.  
안내를 종료합니다.

## 조건문 - if-else

### ✓ 조건문 연습문제

- 사용자로부터 자연수를 입력 받고 값이 짝수인지 홀수인지 화면에 출력합니다

실행 결과:  
자연수 하나 입력해주세요: 20  
짝수입니다.

실행 결과:  
자연수 하나 입력해주세요: 21  
홀수입니다.

## 조건문 - if-else

### ✓ 조건문 연습문제

- 사용자로부터 소프트웨어 개발, 데이터베이스 구축, 프로그래밍 언어 활용 점수를 입력 받습니다. 과목 평균 점수가 60점 이상일 때 합격입니다. 평균 점수에 따라 '합격', '불합격'을 출력하는 프로그램을 만드세요.
- 점수는 0점부터 100점까지만 유효합니다. 이를 벗어난 점수가 입력되면 "잘못된 점수입니다."를 출력하고 '합격', '불합격'에 대해선 출력않고 프로그램을 종료합니다.
- 과목별 점수는 일단 전부 입력 받습니다.

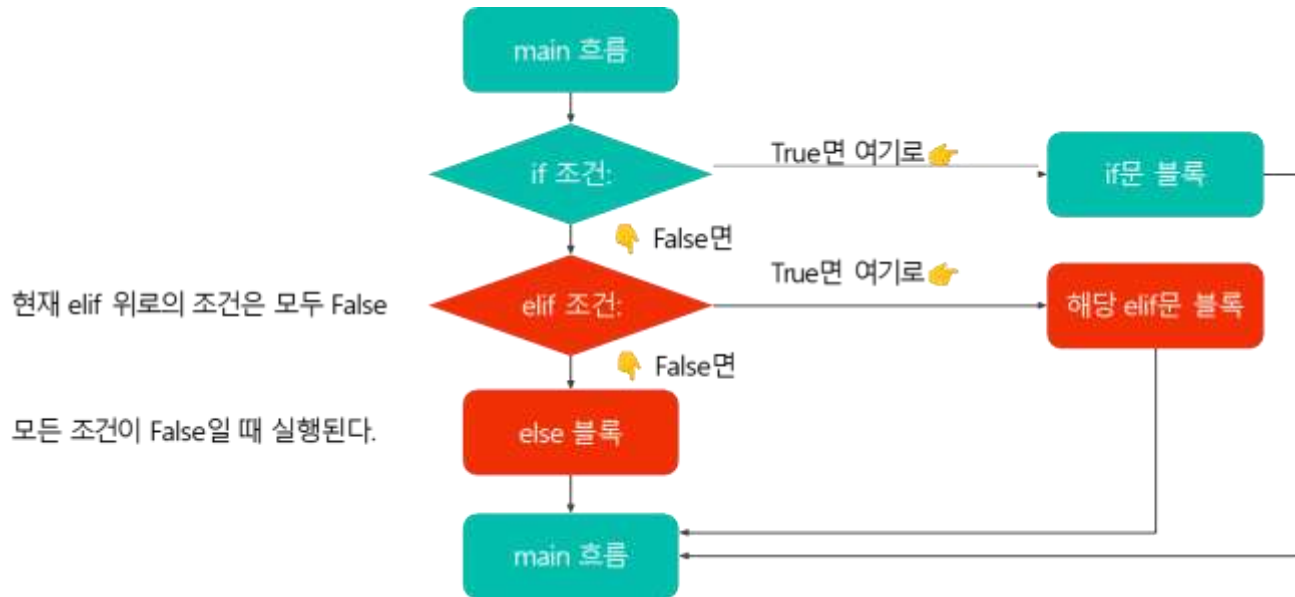
소프트웨어 개발 점수를 입력하세요.. 70  
데이터베이스 구축 점수를 입력하세요.. 80  
프로그래밍 언어 활용 점수를 입력하세요.. 90  
합격

소프트웨어 개발 점수를 입력하세요.. -50  
데이터베이스 구축 점수를 입력하세요.. 30  
프로그래밍 언어 활용 점수를 입력하세요.. 50  
잘못된 점수입니다.

## 조건문 - if-elif-else

### ✓ if-elif-else 조건문

- 여러 개의 조건이 필요하다면 elif를 통해서 조건을 추가할 수 있다.



## 조건문 - if-elif-else

✓ if-elif-else 조건문의 동작 순서

1. if 조건문의 True/False 여부를 판단
2. if 조건문이 True면 if문의 바디를 실행하고 나머지 elif와 else는 실행하지 않고 종료 반대로 if조건문이 False면 그 다음 elif 조건문의 True/False를 확인
3. elif 조건문이 True라면 elif 바디를 실행하고 그 이후에 있는 elif와 else는 실행하지 않고 종료. 반대로 False라면 그 다음 elif의 조건문을 판단. (elif가 있는 동안 반복)
4. 연결된 if와 elif의 조건식이 모두 False라면 else 블록이 실행.

## 조건문 - if-elif-else

### ✓ if-elif-else 조건문

```
score = int(input("점수를 입력해주세요 : "))

print("성적을 알려드립니다.")

if score >= 90:
    print("A입니다.")
elif score >= 80:
    print("B입니다.")
elif score >= 70:
    print("C입니다.")
elif score >= 60:
    print("D입니다.")
else:
    print("F입니다.")

print("안내를 종료합니다.")
```

## 조건문 - if-elif-else

- ✓ 조건문 연습문제
- ✓ 물의 온도에 따라 현재 상태를 출력하는 프로그램을 만들려고 합니다. 사용자로 부터 물의 온도가 몇인지 입력을 받고 100 이상이면 '기체로 예상됩니다.'를 출력합니다. 100보다 작고 0 이상이면 '액체로 예상됩니다.'를 출력합니다. 0 보다 작으면 '고체로 예상됩니다.'를 출력합니다.

실행 결과:

현재 물의 온도를 입력해주세요 -0.12  
고체로 예상됩니다.

## 조건문 - if-elif-else

- ✓ 조건문 연습문제
- ✓ 일반 시내 버스 요금을 받는 프로그램을 만들려고 합니다. 성인(19세 이상)이면 1500원 청소년(12 ~ 18세)이면 1100원 어린이(6~11세)면 800원을 받고 미취학 아동은 무료입니다. 사용자로부터 나이를 입력 받고 나이에 따른 일반 시내 버스 요금을 출력하세요.

실행 결과:  
자신의 나이를 입력하세요:  
19 1500원

실행 결과:  
자신의 나이를 입력하세요  
: 6 800원

실행 결과:  
자신의 나이를 입력하세요  
: 5 무임승차입니다.



## 조건문 - if-elif-else

- ✓ 조건문 연습문제
- ✓ A사는 B사에서 C상품(\$10/개)을 납품합니다. 발주 수량에 따라 수량할인을 고려하여 납품가가 달라집니다. 1000개 미만은 수량할인이 적용되지 않습니다. 1000개부터 1500개 까지 5%, 1500개 초과 2000개까지 10%를 할인합니다. 2000개 초과부터는 20%를 할인합니다. 발주 수량을 사용자로부터 입력 받고 총 발주금액을 출력하는 프로그램을 작성해보세요.(소수점은 할인의 일환으로 버립니다.)

실행 결과:

발주 수량을 입력하세요: 2001

2001개를 주문하셨습니다.

적용 할인율은 20%입니다.

총 발주 금액은 \$16008입니다.

## 조건문 - if-elif-else

- ✓ 조건문 연습문제
- ✓ 주민등록번호 뒷자리 7자리 중 맨 앞 번호가 1, 3인 경우는 남자이고 2, 4인 경우는 여자를 의미합니다. 주민등록번호 뒷자리를 입력 받아 성별을 확인하는 프로그램을 작성하세요.
  1. 자리수가 7자리가 아니면 '자리수를 잘못 입력했습니다.'를 출력합니다.
  2. 맨 앞 번호가 1, 2, 3, 4가 아니면 '잘못된 입력입니다.'를 출력합니다.

실행 결과:

주민번호 뒷자리를 입력하세요: 123456  
자리수를 잘못 입력했습니다.

실행 결과:

주민번호 뒷자리를 입력하세요: 2345678  
여자

실행 결과:

주민번호 뒷자리를 입력하세요: 1234567  
남자

실행 결과:

주민번호 뒷자리를 입력하세요: 5678912  
잘못된 입력입니다.

## 조건문 - if-elif-else

- ✓ 조건문 연습문제
- ✓ 사용자로부터 연도 값을 입력받고 윤년인지 아닌지 결과를 출력하는 프로그램을 작성해보세요.
  1. 연도가 4로 나누어 떨어지는 경우: 윤년, 그 밖의 해: 평년
  2. 예외 규칙으로 100으로 나누어 떨어지되 400으로 나누어 떨어지지 않는 해: 평년

실행 결과:  
확인할 연도를 입력하세요: 1896  
윤년입니다.

실행 결과:  
확인할 연도를 입력하세요: 1900  
평년입니다.

실행 결과:  
확인할 연도를 입력하세요: 2000  
윤년입니다.

실행 결과:  
확인할 연도를 입력하세요: 2023  
평년입니다.

## 조건문 - if-elif-else

### ✓ 조건문 연습문제

```
year = int(input("확인할 연도를 입력하세요:"))  
  
if year % 4 == 0 and (year % 100 != 0 or year % 400 == 0):  
    print("윤년입니다.")  
else:  
    print("평년입니다.")
```