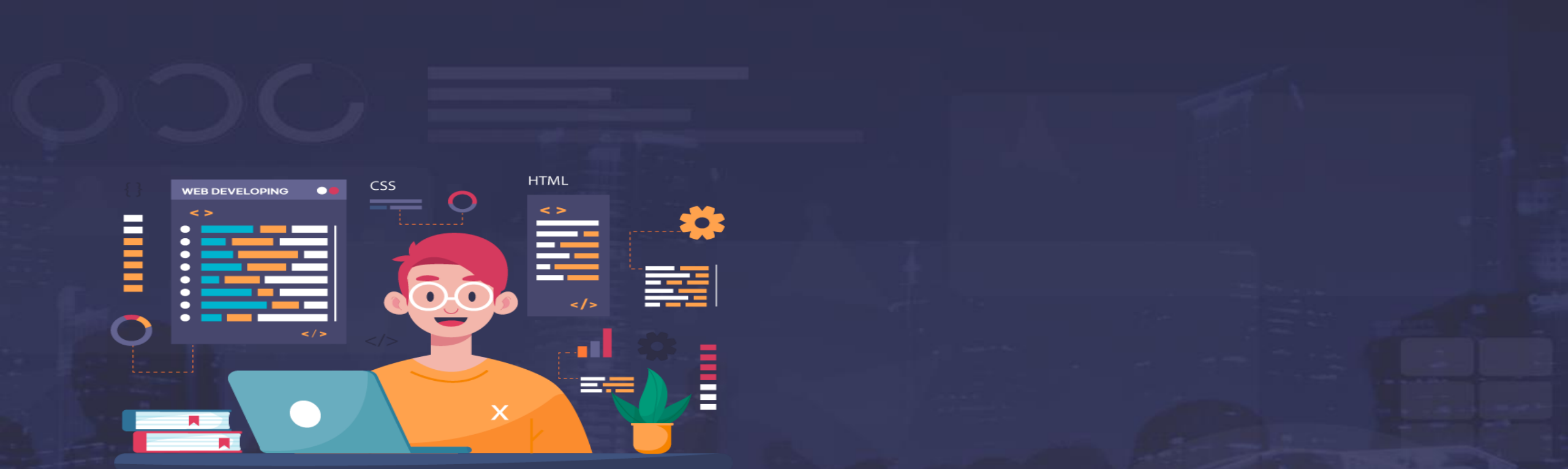


비교과 겨울 방학 특강

파이썬 2일 - 수업자료



양 자영



입출력 명령어

사용자 입력 – input



- 사용자가 입력한 값을 어떤 변수에 대입하고 싶을 때 사용
- input()은 입력되는 모든 것을 문자열로 취급
- 프롬프트 값을 띄워서 사용자 입력받기

```
변수명 = input("입력받기 위해 출력할 내용")
```

```
>>> number = input("숫자를 입력하세요: ")
숫자를 입력하세요: 3
>>> print(number)
3
>>>
```

```
>>> name = input("이름을 입력하시오 :")
이름을 입력하세요 : 홍길동
>>> print(name, "씨, 안녕하세요?")
홍길동 씨, 안녕하세요?
```

input() 연습문제



1. 변수 height에 키(tall)를 실수형(float)으로 묻고 값을 할당하시오.
2. 변수 height를 정수형(int)으로 만들어 소수점 아래를 버리시오.
3. 변수 weight에 상대방의 몸무게를 정수형으로 묻고 값을 할당하시오.
4. 이름 name에 이름을 묻고 값을 할당하시오.

입력받은 후 정수형으로 변환하기

`a = int(input())`으로 받거나 또는 `a = int(a)`

```
print('Hello Python!')
```

Hello Python!

```
print("Hello Python!")
```

Hello Python!

```
print('Hello "Python"')
```

Hello "Python"

```
print("Hello 'Python'")
```

Hello 'Python'

```
print('Hello', 'Python!')
```

Hello Python!

```
print('Hello' + 'Python!')
```

HelloPython!

```
print('I like Python. \nBut I do not like Ruby.')
```

I like Python. But I do not like Ruby.

```
print("""I like Python.
But I don't like C.""")
```

I like Python.
But I don't like C.

```
print('I like Python.
But I don't like C.')
```

I like Python.
But I don't like C.

```
print('Hello', end=' ')
```

Hello Python

```
print('Python')
```

```
print('Hello', end='&&&')
```

Hello&&&Python

```
print('Python')
```

Hello#Python

```
print('Hello', 'Python', sep='#')
```

문자열 포매팅(formatting) %



문자열의 서식(format)을 이용하여 해당 위치에 문자열 불러오기

```
>>> '%s %s %s' %("apple", "banana", "cat")
'apple banana cat'
>>> print("I think %s is the best fruit" %("mango"))
I think mango is the best fruit
>>>
```

문자 = %s

```
>>> '%d %d' %(15, 35)
'15 35'
>>> print("I will diet until I get %dkg" %(75))
I will diet until I get 75kg
>>> print("I will diet until I get %fkg" %(74.5))
I will diet until I get 74.500000kg
>>>
```

숫자 = %d

문자열 포매팅(formatting) %



문장 사이에 여러 변수가 들어갈 때 / 자리에 들어갈 내용 오류 방지

포맷 서식	설명	예시	() 변수명
%s	문자열	"I love %s" %("meet")	
%d	정수	"My thing is %dcm" %(180)	
%c	한글자 문자열	"I get %c grade" %("A")	
%x	16진수로 표현	"%x마일 남았어" %(20)	
%e	부동 소수점 표현	"%e키로만 더 짜워야지" %(12345678)	
%f	실수	"%f키로만 더 빼야지" %(0.005)	

```
i = 123; f = 3.14; s = 'Hello'
```

```
i: 123, f: 3.140000, s: Hello
```

```
print('i: %d, f: %f, s: %s' % (i, f, s))
```

```
i:    123, f:  3.14, s:  Hello
```

```
print('i: %9d, f: %5.2f, s: %7s' % (i, f, s))
```

```
i: 000000123, f: 03.14, s:  Hello
```

```
print('i: %09d, f: %05.2f, s: %7s' % (i, f, s))
```

```
i: 123, f: 3.14, s: Hello
```

```
print('i: {}, f: {}, s: {}'.format(i, f, s))
```

```
f: 3.14, i: 123, s: Hello
```

```
print('f: {1}, i: {0}, s: {2}'.format(i, f, s))
```

```
f: 3.14, i: 123, s: Hello
```

```
print('f: {ff}, i: {ii}, s: {ss}'.format(ii=i, ff=f, ss=s))
```

Print문 사용 f-string



f'문자열 {변수} 문자열'

문자열 맨 앞에 f를 붙이고, 출력할 변수, 값을 중괄호 안에

```
s = 'coffee'
```

```
n = 5
```

```
result1 = f'저는 {s}를 좋아합니다. 하루 {n}잔 마세요.'
```

```
print(result1)
```

f-string 중괄호 출력

```
num = 10
```

```
result = f'my age {{{num}}}, {{num}}'
```

```
print(result)
```



연산자 - 산술연산자

어디에나 있는 수식



- 컴퓨터를 사용하면 우리가 복잡한 계산을 직접 해야 할 필요는 없음.
- 우리는 정확한 계산을 하도록 알맞은 수식을 올바르게 작성하여 컴퓨터에 지시해야 함
- 올바른 수식 작성을 위해 알아야 할 것
 - 연산자 우선순위
 - 연산자 종류 : 산술 연산자, 관계 연산자, 논리연산자
 - 연산의 방향

- 수에 대한 계산

연산자	기호	사용 예	결괏값
더하기	+	$7 + 4$	11
빼기	-	$7 - 4$	3
곱하기	*	$7 * 4$	28
지수(제곱)	**	$7 ** 4$	2401
나누기	/	$7 / 4$	1.75
나누기(몫)	//	$7 // 4$	1
나머지	%	$7 \% 4$	3
대입(할당)	=	$x = 10$	



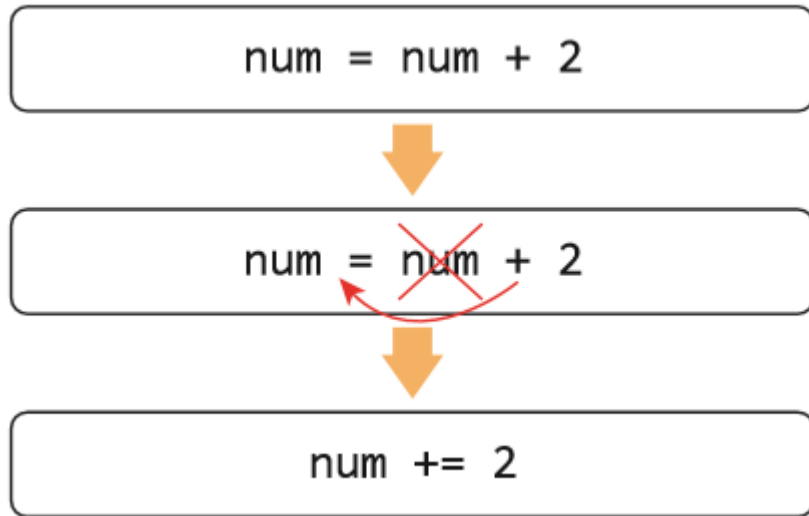
예제



- $x=2$ 일 때, $y = 3x$ 의 값을 계산하여 출력해보자.
- $x=2$ 일 때 2차 함수 $y = 3x^2 + 7x + 9$ 함수의 값을 계산하여 보자.
- $x=2$ 일 때, $y = \frac{3}{x+1}$ 의 값을 계산하여 출력해보자.
- $x=2$ 일 때, $y = 2^{x+1}$ 의 값을 계산하여 출력해보자.
- $x=2$ 일 때, $y = \sqrt{x}$ 의 값을 계산하여 출력해보자.

복합 연산자

- **복합 연산자(compound operator)**란 +=처럼 대입 연산자와 다른 연산자를 합쳐 놓은 연산자이다.



복합 연산자	의미
$x += y$	$x = x + y$
$x -= y$	$x = x - y$
$x *= y$	$x = x * y$
$x /= y$	$x = x / y$
$x \% = y$	$x = x \% y$

x = 10

x = x * 2 + 3

x = 10

x *= 2 + 3

결과의 차이는?

실습 - 삼각형의 면적 구하기



- 밑변과 높이를 사용자에게 입력받아 직각삼각형의 면적을 구해보자

- ① 밑변을 변수 w 에 입력받는다.
- ② 높이를 변수 h 에 입력받는다.
- ③ 삼각형의 넓이($s=w*h/2$)를 계산한다.
- ④ 넓이(s)를 출력한다.

실습 - 총점과 평균 구하기



- 다음과 같은 프로그램을 작성하시오.

이름을 입력하세요 : 파이썬
국어 성적을 입력하세요 : 95
수학 성적을 입력하세요 : 98
영어 성적을 입력하세요 : 84

`round(3.141592, 2)`

총합 277 점, 평균 92.33 점 입니다

실습 - 화씨온도를 섭씨온도로 변환하기

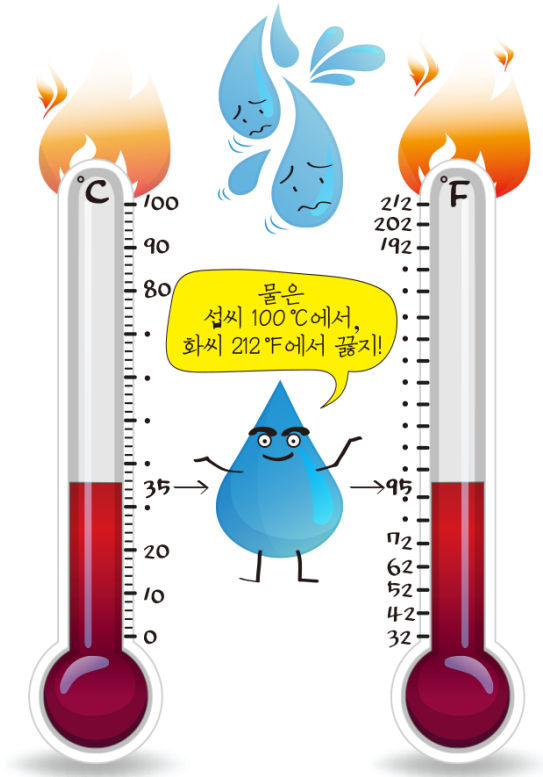
- 화씨온도를 입력 받아서 섭씨온도로 바꾸는 프로그램

$$C = (F - 32) \times \frac{5}{9}$$

실행 결과

화씨온도: 100

섭씨온도: 37.777777777777778



실습 - 직각 삼각형 빗변 길이 구하기



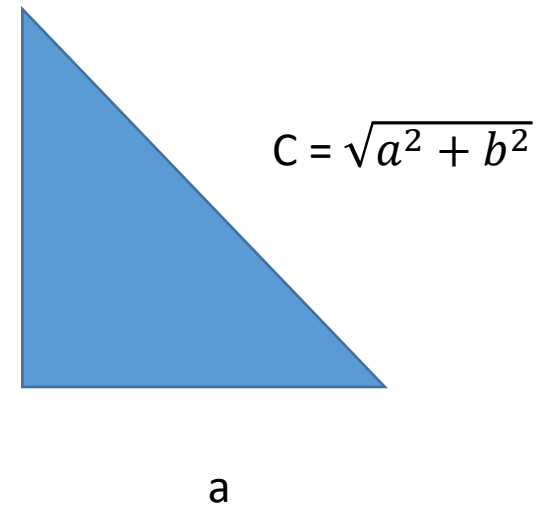
- 직각 삼각형의 빗변을 계산하여 출력하는 프로그램을 작성하시오.

첫번째 변의 길이: 15.3

두번째 변의 길이: 12.1

빗변의 길이는 19.50640920313116 cm입니다.

>>>



실습 - 초를 분,초로 변환하기



- 초를 묻고 몇 분 몇 초로 표현하는 프로그램을 작성하시오.

```
sec =   
min =   
remainder =   
print()
```

초를 입력하시오 **1000**
1000 초 = 16 분 40 초

실습 - 이차방정식의 근 구하기



- 이차방정식의 근을 계산하여 출력하는 프로그램을 작성하시오.

이차방정식 $ax^2 + bx + c$ 해 계산기입니다.

A 값 : 1

B 값 : 5

C 값 : 6

이차방정식 $1x^2 + 5x + 6$ 의 해는
-2.0 -3.0 입니다.

$$\text{근} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



연산자 - 관계연산자, 논리연산자

관계연산자

수학 기호	파이썬 기호	설명	
$a = b$	<code>a == b</code>	참: a와 b가 같으면	거짓: a와 b가 다르면
$a \neq b$	<code>a != b</code>	참: a와 b가 다르면	거짓: a와 b가 같으면
$a < b$	<code>a < b</code>	참: a가 b보다 작으면	거짓: a가 b보다 크거나 같으면
$a \leq b$	<code>a <= b</code>	참: a가 b보다 작거나 같으면	거짓: a가 b보다 크면
$a > b$	<code>a > b</code>	참: a가 b보다 크면	거짓: a가 b보다 작거나 같으면
$a \geq b$	<code>a >= b</code>	참: a가 b보다 크거나 같으면	거짓: a가 b보다 작으면

```
IDLE Shell 3.10.2
File Edit Shell Debug O

>>> x = 100
>>> y = 1
>>> x == y
False
>>> x != y
True
>>> x > y
True
>>> x < y
False
>>> x >= y
True
>>> x <= y
False
```


- '5와 5는 같다.'의 명제를 타이핑하여 보자.
- '8과 5는 같지 않다.'의 명제를 타이핑하여 보자.

논리연산자



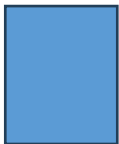
논리연산자	x	y	결과
x and y	False	False	False
	False	True	False
	True	False	False
	True	True	True
x or y	False	False	False
	False	True	True
	True	False	True
	True	True	True
not x	False		True
	True		False

복잡한 논리 조건의 표현 - 논리 연산자



- 논리 연산자

- 여러 개의 조건을 조합하여 참인지 거짓인지를 따질 때 사용
- 예) 나이가 10살 이상이고, 키가 110cm 이상

age \geq 10  height \geq 110



연산	의미
x and y	x와 y가 모두 참이면 참, 하나라도 거짓이면 거짓
x or y	x나 y 중에서 하나만 참이면 참, 모두 거짓이면 거짓
not x	x가 참이면 거짓, x가 거짓이면 참

연산자 우선순위

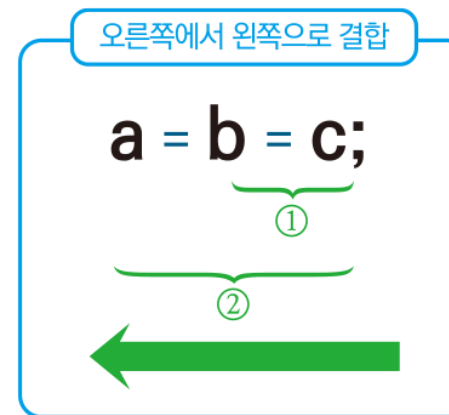


연산자	설명	
**	지수 연산자	 높음
~, +, -	단항 연산자	
*, /, %, //	곱셈, 나눗셈, 나머지 연산자, 나눗셈(몫)	
+, -	덧셈, 뺄셈	
>>, <<	비트 이동 연산자	
&	비트 AND 연산자	 낮음
^,	비트 XOR 연산자, 비트 OR 연산자	
<=, <, >, >=	비교 연산자	
<>, ==, !=	동등 연산자	
=, %=, /=, //=, -=, +=, *=, **=	대입 연산자	
is, is not	아이덴티티 연산자	
in, not in	소속 연산자	
not, or, and	논리 연산자	

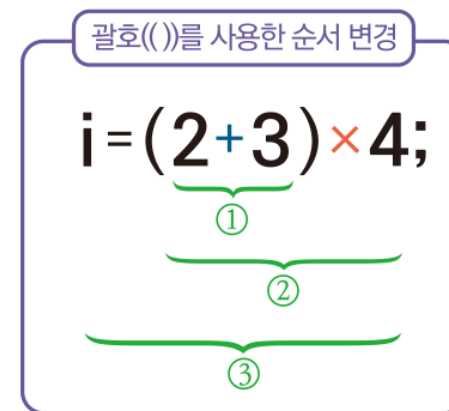
연산자 우선순위



- 같은 우선순위의 경우



- 괄호를 사용한 순서 변경





제어 구조 - 조건문

컴퓨팅 사고 - 알고리즘



- 문제를 해결하기 위한 단계적인 절차
- 어떤 일을 수행할 수 있는 일련의 명령어 또는 규칙의 집합

문제 : 라면 끓이기
알고리즘 : 라면 끓이는 법



1. 물 550ml을 끓인다.



2. 스프와 면을 넣은 후 4분 더 끓인다.

알고리즘을 바탕으로

개인의 취향과 기호에 맞춰 변형 가능

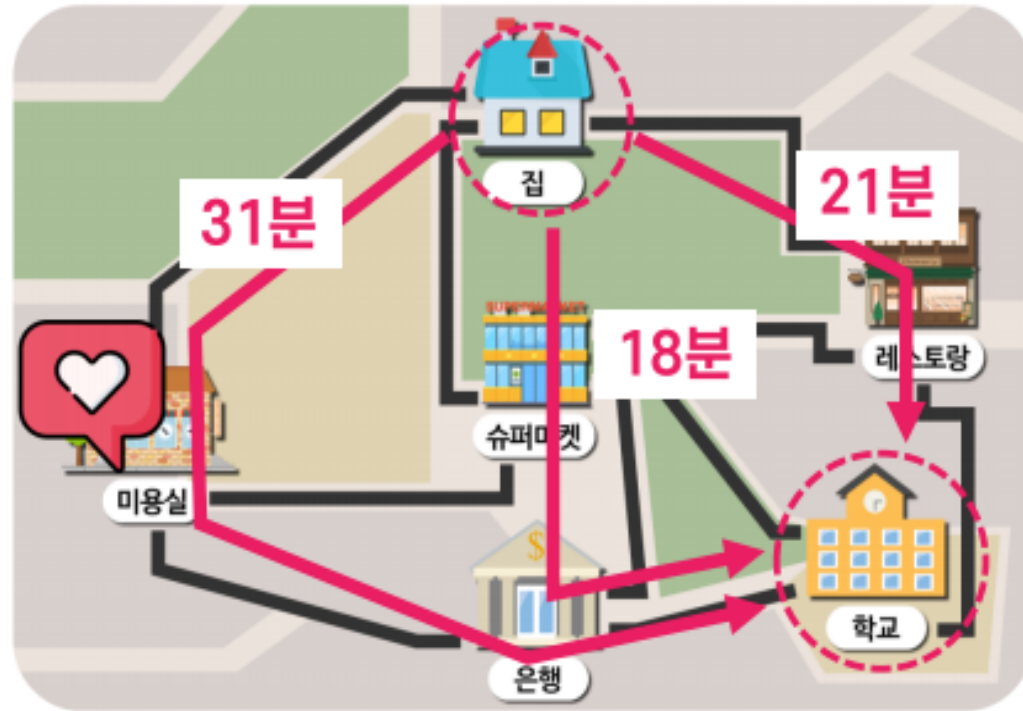


3. 맛있게 먹는다.

컴퓨팅 사고 - 알고리즘

- 알고리즘에 따라 문제 해결의 시간과 비용이 달라짐

집에서 학교까지 가기

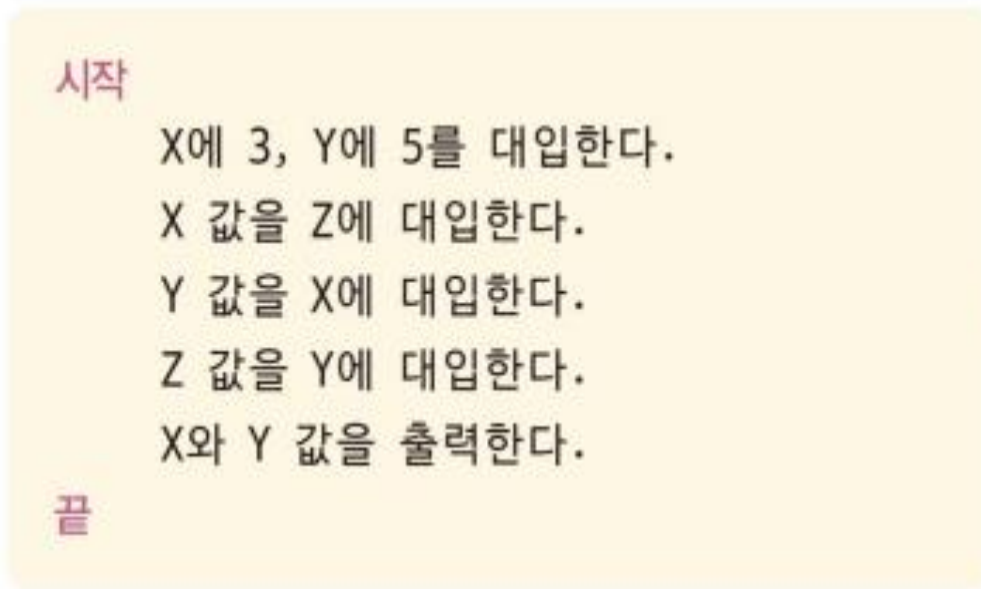


컴퓨팅 사고 - 알고리즘

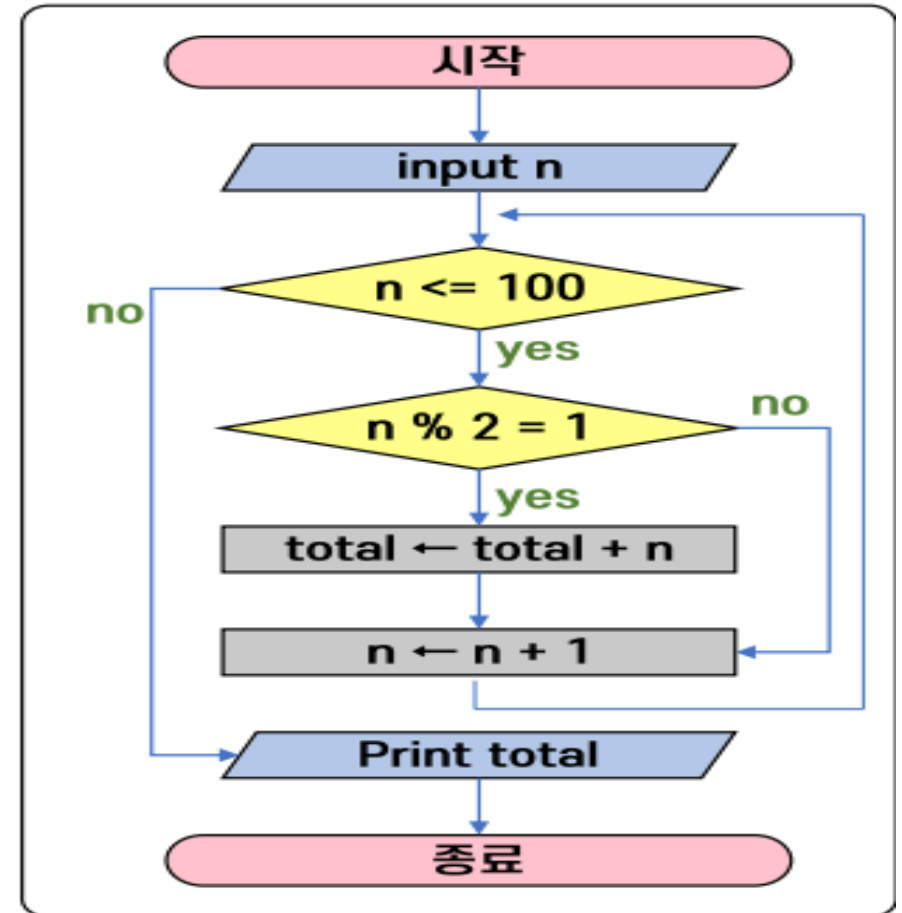
- 알고리즘의 표현

- 의사코드(pseudo code) 또는 순서도로 표현

의사코드



순서도



알고리즘의 표현 (1) - 순서도



- 다양한 기호를 사용하여 알고리즘을 표현한 것
- 순서도 역시 명확하고 실현 가능한 흐름으로 작성해야 함

기호	이름	설명
	흐름선	작업의 흐름을 나타낸다.
	단말	순서도의 시작과 끝을 나타낸다.
	준비	작업 단계 시작 전 준비를 나타낸다.
	처리	처리해야 할 작업을 명시한다.
	판단	조건 연산을 나타낸다. 연산 결과(True, False)에 따라 흐름선이 선택된다.
	입출력	데이터의 입력과 출력을 나타낸다.
	서브루틴	다른 곳에 정의된 서브프로그램을 호출한다.

알고리즘 표현 (2) – 의사코드(pseudo code)



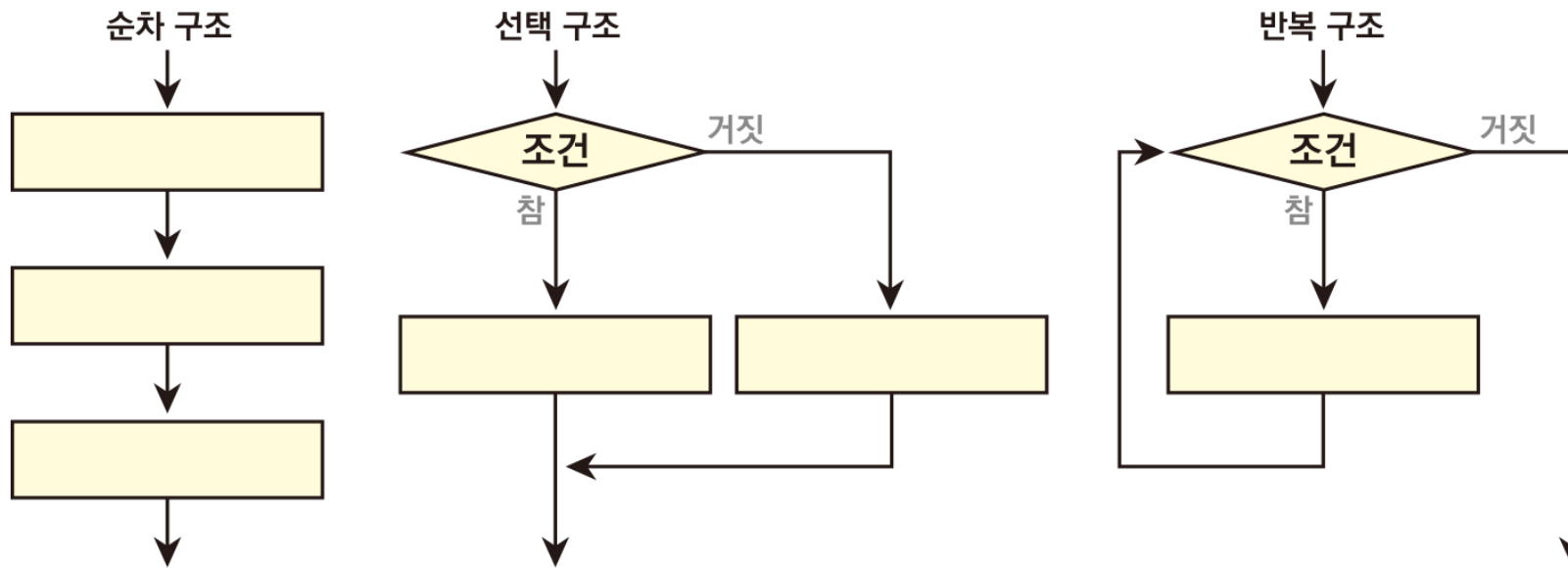
- 키오스크를 이용한 메뉴 주문 알고리즘

- ① 키오스크에서 메뉴를 검색한다.
- ② 단품과 세트 메뉴 중 원하는 메뉴를 선택한다.
- ③ 배가 많이 고프면 사이드 메뉴를 선택하고, 그렇지 않으면 생략한다.
- ④ 결제할 금액을 확인하고 카드 투입구에 카드를 넣는다.
- ⑤ 결제가 완료되면 주문 번호와 영수증을 챙긴다.
- ⑥ 주문 번호가 호출되면 음식을 수령한다.



프로그램에서 명령어 기본 제어 구조

- 순차 구조(sequence) : 명령어들이 순차적으로 실행되는 구조
- 선택 구조(selection) : 둘 중의 하나의 명령을 선택하여 실행되는 구조
- 반복 구조(iteration) : 동일한 명령이 반복되면서 실행되는 구조

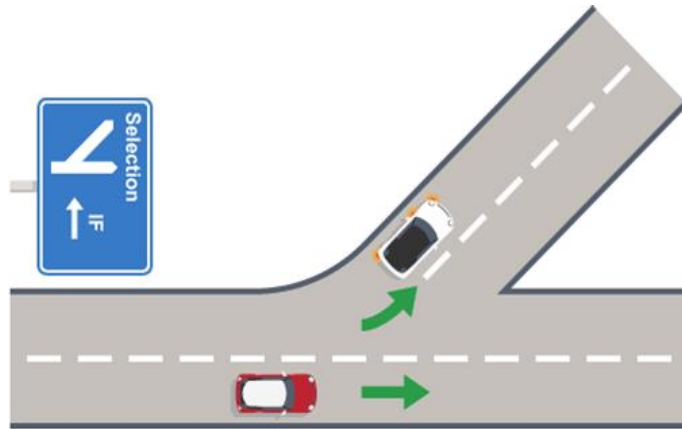


선택 구조가 필요한 이유

- 미성년자가 아니라면 영화 "킹덤"을 시청할 수 있다.
- 게임에서 철수가 정답을 맞혔으면 철수의 점수를 1만큼 증가한다.
- 게임 사용자가 외계인 우주선을 맞추었으면 폭발 사운드를 출력한다.
- 파일이 하드 디스크에 없으면 오류 메시지를 출력한다.
- 11:59분이 지나면 수강신청을 할 수 있다(특정 시간에 따른 실행).



선택에 의해서 가는
길이 달라짐



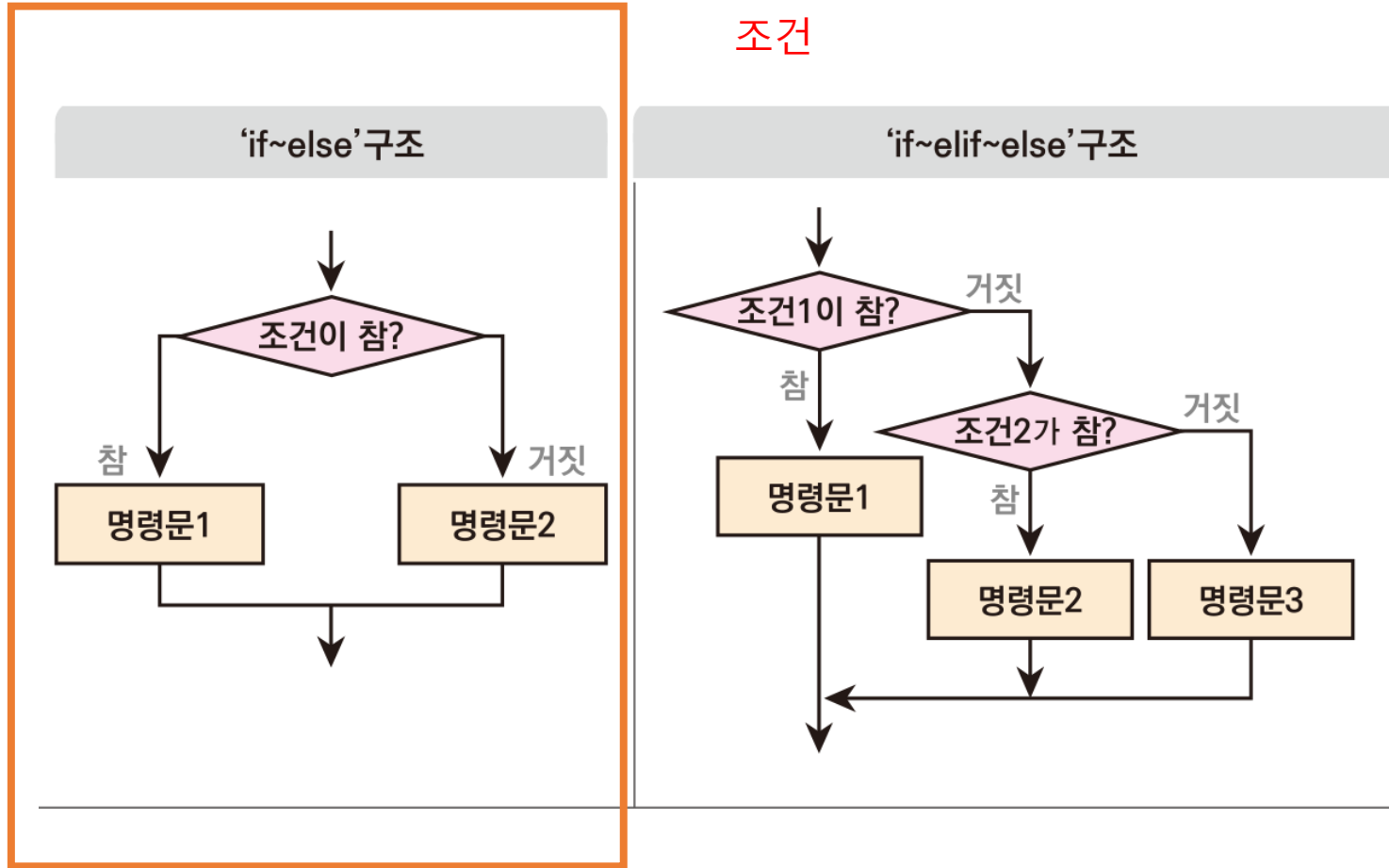
" 만약 ~ 라면 ~ "

↑
조건

↑
동작

"만약 ~ 라면 ~"

↑
조건



조건문 - if문 기본 구조

if와 else를 사용한 조건문의 기본 구조

```
if 조건문:  
    수행할 문장1  
    수행할 문장2  
    ...  
else:  
    수행할 문장A  
    수행할 문장B  
    ...
```

콜론(:) 으로 종속절 시작

들여쓰기로 구분

- 조건문이 참이면 if 블록 수행
- 조건문이 거짓이면 else 블록 수행
- 들여쓰기를 한다. 대부분 4칸
- 들여쓰기가 끝나면 조건문을 종료한다.



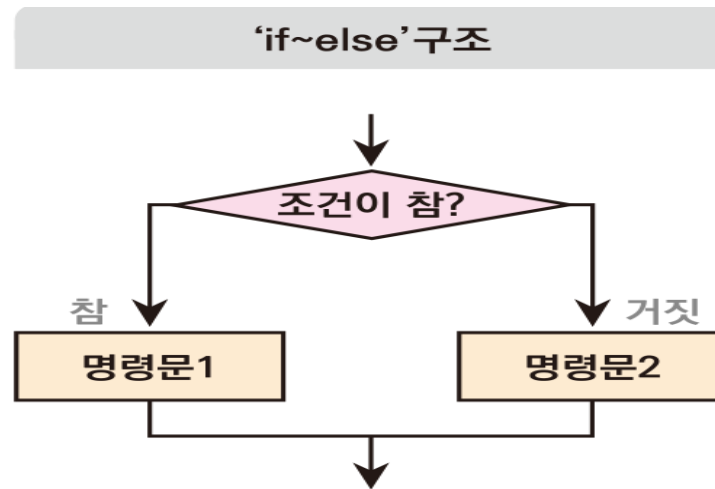
if 조건 :

문장 1	한 번 들여쓰기로 참일 때 실행할 문장임을 나타냄
문장 2	연이어 한 번 들여쓰기로 역시 참일 때 실행할 문장임을 나타냄
문장 3	오류 발생 (두 번의 들여쓰기로 인해 어떤 범주에 속하는지 모호함)

if 조건 :

문장 1	한 번 들여쓰기로 참일 때 실행할 문장임을 나타냄
문장 2	들여쓰기를 하지 않아 참일 때 실행되는 문장이 아님을 나타냄
문장 3	오류 발생 (아무런 의미 없이 들여쓰기를 사용하여 인터프리터가 문장을 해석하는 데 모호성이 발생)

- 변수에 숫자 **문고** 저장하고 **변수값이 3보다 클 경우** "세살보다 크다."를 출력하고 아닌 경우, "세살보다 적다" 을 출력하자
- 변수에 이름을 **문고**, 변수값이 본인의 이름과 **같을 경우** "본인 인증."를 출력하고, 아닌경우 "본인 아님"을 출력하자.



- 숫자를 묻고, 입력받은 정수가 짝수이면 "짝수입니다"를 출력하고 홀수이면 "홀수입니다"를 출력하는 프로그램을 작성하시오.

정수를 입력하세요 : 4
짝수입니다

정수를 입력하세요 : 3
홀수입니다

실습 – 두 수에서 큰값 찾기



- 두 정수를 입력받아 큰값을 구해서 출력하는 프로그램을 작성하자.

```
x,y = map(int, input("두 정수 x y ").split())
```

```
x, y = map(int, input("두 정수 x y ").split())
```

```
if x > y :
```

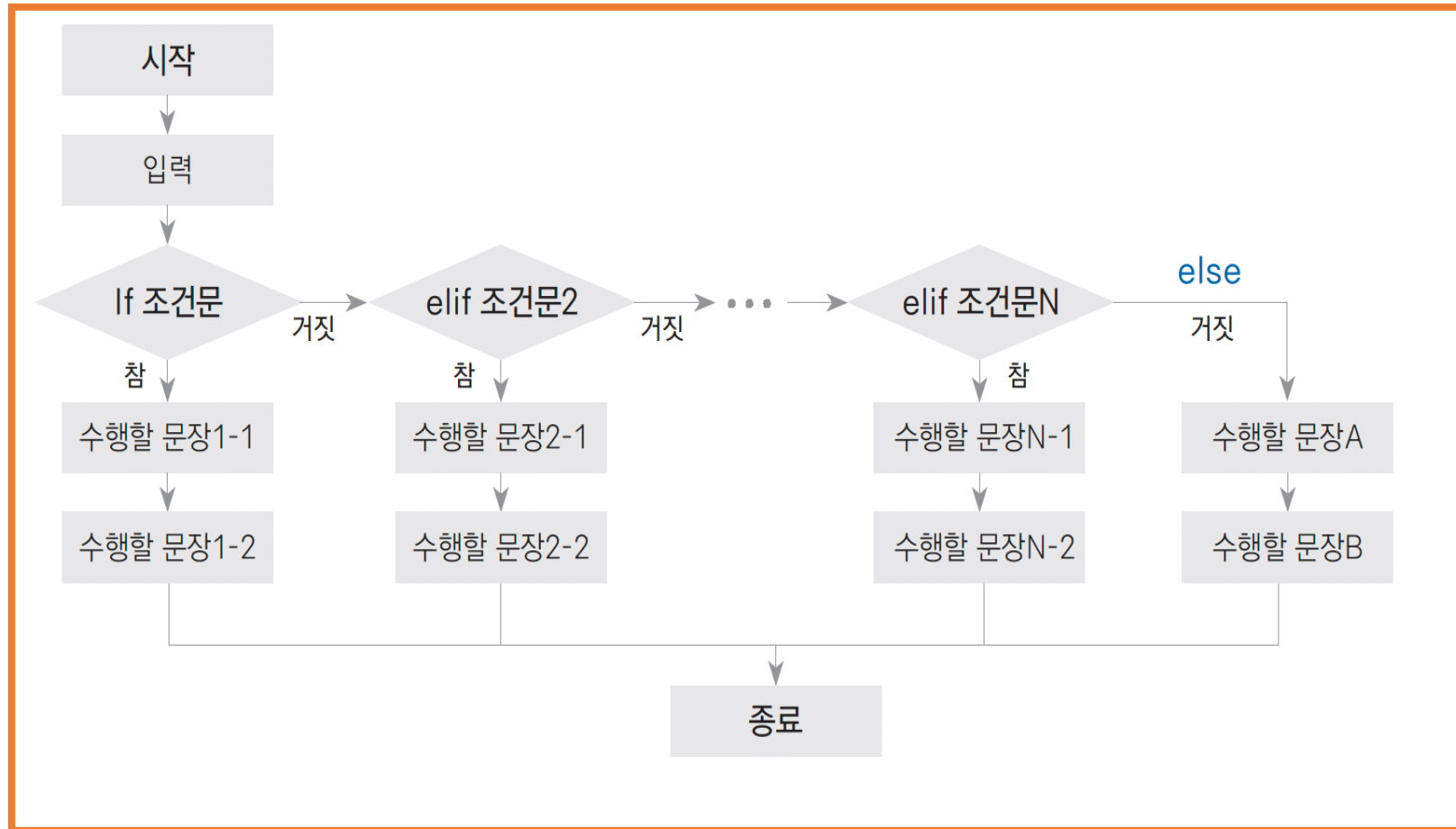
```
    print(x)
```

```
else :
```

```
    print(y)
```

다중 조건문 - elif문

- 여러 가지 조건을 나열해야 할 때 편리하게 사용



조건문 - elif문



if 조건 1 :

조건 1이 참인 경우 문장 1

...

조건 1의 참, 거짓에 따라 실행 결정

➔ 조건 1이 참 이면 실행 O, 거짓인 경우 실행 X

elif 조건 2 :

조건 2가 참인 경우 문장 1

...

조건 2의 참, 거짓에 따라 실행 결정

➔ 조건 1이 거짓이고 조건 2가 참이면 실행 O,
거짓인 경우 실행 X

else:

문장 1

...

➔ 조건 1도 거짓이고 조건 2도 거짓이면 실행

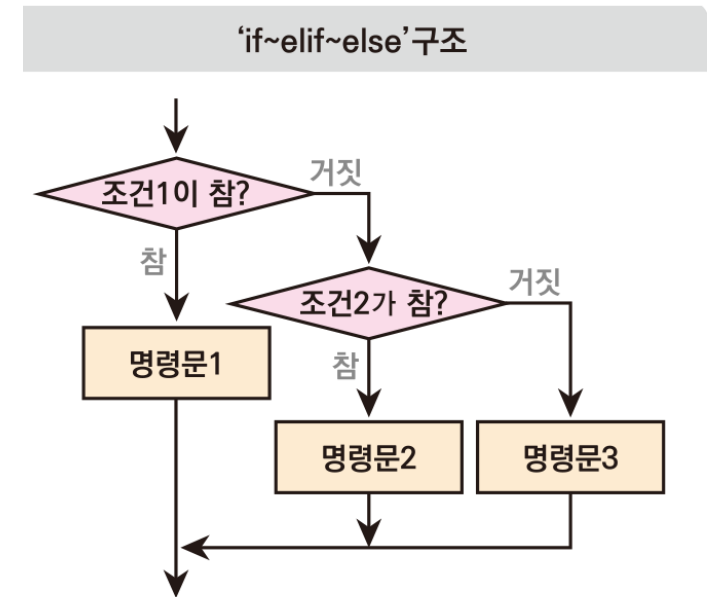
실습 - 합격/재시/불합격 판별하기

- 사용자에게 입력받은 점수가 70 이상이면 "합격입니다", 60 이상이고 70점 미만이면 "재시입니다", 60점 미만이면 "불합격입니다"를 출력하는 프로그램을 작성하시오.

점수를 입력하세요 : 80
합격입니다

점수를 입력하세요 : 67
재시입니다

점수를 입력하세요 : 50
불합격입니다



실습 – A/B/C/D/F 성적 판별하기

- 입력받은 성적이 90점 이상이면 A, 80점 이상이면 B, 70점 이상이면 C, 60점 이상이면 D, 그 외 점수이면 F를 출력하는 프로그램을 작성하시오.

1. score 변수에 성적을 입력받는다.
2. score 값이 90 이상이면
 1. “A” 라고 출력한다.
3. 아니고 Score 값이 80 이상이면
 1. “B” 라고 출력한다
4. 아니고 Score 값이 70 이상이면
 1. “C” 라고 출력한다
5. 아니고 Score 값이 60 이상이면
 1. “D” 라고 출력한다
6. 아니면
 1. “F” 라고 출력한다.

성적을 입력하세요 : 95

A

성적을 입력하세요 : 88

B

성적을 입력하세요 : 70

C

성적을 입력하세요 : 66

D

성적을 입력하세요 : 50

F

복잡한 조건의 표현 - 논리 연산자



- 문제1)

- 국어 점수가 80점 이상 **이고** 수학점수가 80점 이상이면 "PASS", 그렇지 않으면 "FAIL"을 출력하는 프로그램을 작성하시오.

```
kor = int(input("국어 : "))
```

```
mat = int(input("수학 : "))
```

국어 : 80

수학 : 90

PASS

국어 : 70

수학 : 90

FAIL

- 문제2)

- 국어 점수가 80점 이상 **이거나** 수학점수가 80점 이상이면 "PASS", 그렇지 않으면 "FAIL"을 출력하는 프로그램을 작성하시오.

```
kor = int(input("국어 : "))
```

```
mat = int(input("수학 : "))
```

국어 : 70

수학 : 90

PASS

국어 : 50

수학 : 60

FAIL

실습 – BMI 판별하기



이름을 입력하세요 : 파이썬
키(cm)를 입력하세요 : 176
몸무게(kg)를 입력하세요 : 73

파이썬님의 키는 176 cm이고 몸무게는 73 kg 입니다.
BMI 지수는 23.57 입니다. 과체중 입니다.



BMI기준	18.5~22.9	23~24.9	25~29.9	>30
비만분류	정상	과체중	비만	고도비만

실습 - 체력장 합격 판별하기



- 모두 8점 이상을 맞아야 합격 가능성이 높은 것으로 출력

100m 기록(초) : 13.6
1000m 기록(초) : 235
윗몸일으키기 기록(회) : 52
좌우 악력 기록(kg) : 56
팔굽혀펴기 기록(회) : 56

합격 가능성이 매우 높습니다.

	100m	1000m	윗몸일으키기	좌우악력	팔굽혀펴기
8점 커트라인	13.6	237	51	56	46

실습 - 윤년 판별하기



- 연도를 입력받아 윤년을 판단하는 프로그램을 작성하시오.

<윤년의 조건>

- ① 연도가 4로 나누어 떨어지고 100으로 나누어 떨어지지 않으면 윤년이다.
- ② 연도가 400으로 나누어 떨어지면 윤년이다.

1. year 변수에 연도를 입력받는다.
2. year가 4의 배수이면서 100의 배수가 아니거나 400의 배수이면
 1. “윤년” 이라고 출력한다.
3. 아니면
 1. “윤년이 아닙니다” 라고 출력한다.

연도를 입력하세요 : 2023
2023 년은 윤년이 아닙니다.

연도를 입력하세요 : 2024
2024 년은 윤년입니다.

연도를 입력하세요 : 2400
2400 년은 윤년입니다.