



프로그래밍 언어

2주차



인하대학교



원의 면적 계산하기

실행결과

반지름 20인 원의 면적= 1256.0

알고리즘

STEP #1. 원의 반지름을 저장한다.

STEP #2. 공식을 적용하여 면적을 계산한다.

$$\text{area} = \text{radius} * \text{radius} * \pi$$

STEP #3. 면적을 화면에 출력한다.

코드

변수 radius에 값을 저장한다.

radius = 20

주석(Comment)

공식을 적용하여 면적을 계산한다

area = 3.14 * radius * radius

면적을 화면에 출력한다.

print("반지름이 ", radius, "인 원의 면적=", area)

- 주석(comment)은 소스 코드에 붙이는 설명글과 같은 것이다.
- 주석은 프로그램이 하는 일을 설명한다.
- #표시가 있으면 주석으로 인식
- 주석은 프로그램의 실행 결과에 영향을 끼치지 않는다.

사용자로부터 화씨온도를 입력받는다.

```
ftemp = int(input("화씨온도: "))
```

```
ctemp = (ftemp-32.0)*5.0/9.0
```

```
print("섭씨온도:", ctemp)
```

화씨온도->섭씨온도

섭씨온도를 화면에 출력한다.

주석으로 컴파일러에게 무시되지만 프로그램에 대한 설명이나 메모를 붙이는 것이다.



프로그램을 이해하기 쉽게 하려면

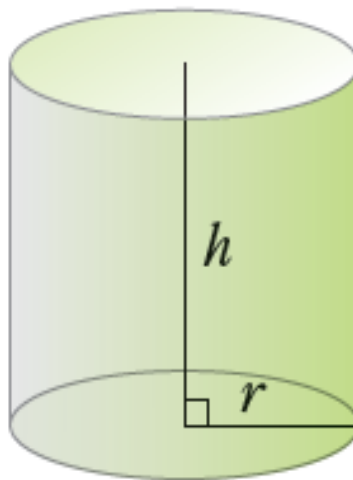
의미있는
변수이름

주석문
활용

실습

원기둥의 부피 계산

반지름 = 5
높이 = 10
원기둥의 부피 = 785.0



$$V = \pi r^2 h$$

원기둥의 부피

```
PI = 3.14
```

```
radius = 5
```

```
height = 10
```

```
volume = PI * radius * radius * height
```

```
print("반지름=", radius, "높이=", height, "원기둥의 부피=", volume)
```

입출력명령(함수)

출력 : print()

표준출력장치 : 모니터

입력 : input()

표준입력장치 : 키보드

출력

```
print("홍길동씨, 안녕하세요?")
```

```
print("파이썬에 오신 것을 환영합니다.")
```



모니터

홍길동 씨, 안녕하세요?

파이썬에 오신 것을 환영합니다.

입력

문자

```
name=input("이름을 입력하시오: ")
```

```
name = input()  
print(name, "씨, 안녕하세요?")  
print("파이썬에 오신 것을 환영합니다.")
```

사용자입력

홍길동

홍길동 씨, 안녕하세요?
파이썬에 오신 것을 환영합니다.

이름을 입력하시오: 강감찬
강감찬 씨, 안녕하세요?
파이썬에 오신 것을 환영합니다.

입력 `input()` 함수

Syntax: `input()` 함수

형식 변수 = `input`(안내메시지)

예 `x = input("이름을 입력하시오: ")`

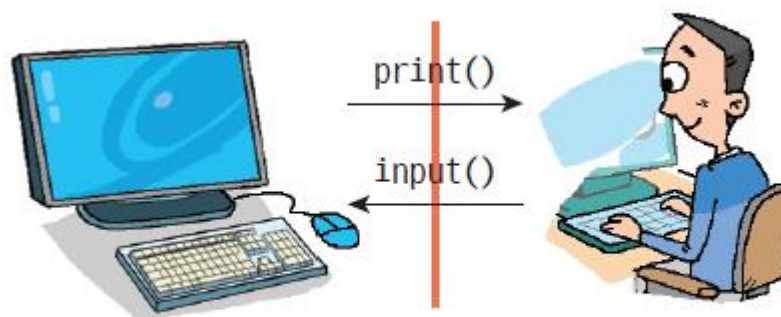
변수

안내 메시지를 출력하고 사용자가 입력한 값을 문자열 형태로 반환한다.

한 개의 인수만 허용

예. `input("이름을", "입력하시오")` X

`Input("이름을"+"입력하시오")` O



사용자 인터페이스

실습

두 수를 사용자로부터 입력받으려면?

```
a=5
```

```
b=10
```

```
sum = a+b
```

```
print(sum)
```

```
a= input()
```

```
b= input()
```

```
sum = a+b
```

```
print(sum)
```

결과

실행결과

```
>>>
=====
100
50
10050
>>>
```

- input() 함수는 모든 종류의 값을 입력받지만, 문자열로 취급하기 때문에 오류 발생
- 자료형 변환 필요



자료형 (Data Type)

실수형



인하대학교

- Floating point number
- 소수점이 있는 데이터

```
>>> a=3.14
>>> b=3.14e5
>>> print(a, b)
3.14 314000.0
>>> type(b)
<class 'float'>
```

문자열

문자열은 반드시 양쪽을 큰따옴표(“)나 작은따옴표(’)로 감싸야 함

```
>>> a=“파이썬”
```

```
>>> print(a)
```

```
파이썬
```

```
>>> type(a)
```

```
<class 'str'>
```


불(Boolean)

- Boolean : 참(true) 또는 거짓(false)의 두 가지 값만을 가질 수 있는 데이터 타입
- Bool형은 단독으로 사용하기보다는 if 조건문 이나 while 반복문 등과 함께 주로 사용

```
>>> a=True
>>> type(a)
<class 'bool'>
```

```
>>> a=100
>>> b=200
>>> print(a>b)
False
>>> print(a<b)
True
```



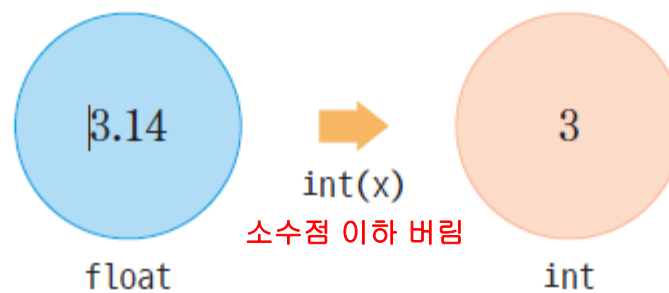
자료형 변환

타입 변환

Syntax: 타입변환

형식 새로운타입(변수)

예 $x = 3.14$
 $y = \text{int}(x)$



자료형변환

- 문자열은 `int()` 함수에 의해서 정수로, `float()` 함수에 의해서 실수로 변경

```
>>> s1='100'
>>> s2='100.123'
>>> s3='999999999999'
>>> print(int(s1)+1)
101
>>> print(float(s2)+1)
101.123
>>> print(int(s3)+1)
1000000000000
```

```
>>> a='10'
>>> print(a)
'10'
>>> print(int(a))
10
>>> print(int("100"))
100
>>> print(int(100.123))
100
```

```
>>> a='123.4'
>>> int(a)
오류
>>> int(float(a))
123
```

자료형 변환

- 숫자를 문자열로 변환하기 위해서는 `str()` 함수를 사용

```
>>> a=100
>>> b=100.123
>>> print(str(a)+'1')
1001
>>> print(str(b)+'1')
101.1231
```

```
>>> movie = "Terminator" + 3
TypeError: can only concatenate str (not "int") to str
```

```
>>> movie = "Terminator" + str(3)
>>> movie
'Terminator3'
```

(주의)문자에 사용가능한 연산자 : +와 *

- 문자와 변수(또는 숫자)를 함께 출력하고 싶을 때

- 콤마(,)로 연결

예. `print("당신의 이름은",name,"입니다.")`

- + 로 연결할 때

- 문자끼리 사용(문자변수가능)

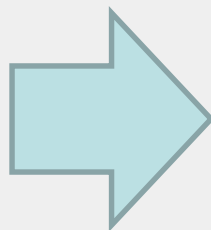
예. `print("당신의 이름은"+name+"입니다.")`

- 숫자가 저장된 변수라면 형변환함수 이용

예. `print("당신의 중간고사 평균점수는"+ave+"점입니다.")` X

→ `print("당신의 중간고사 평균점수는"+str(ave)+"점입니다.")` O

```
a=input()  
b=input()  
sum = a+b  
print(sum)
```



```
a=int(input())  
B=int(input())
```



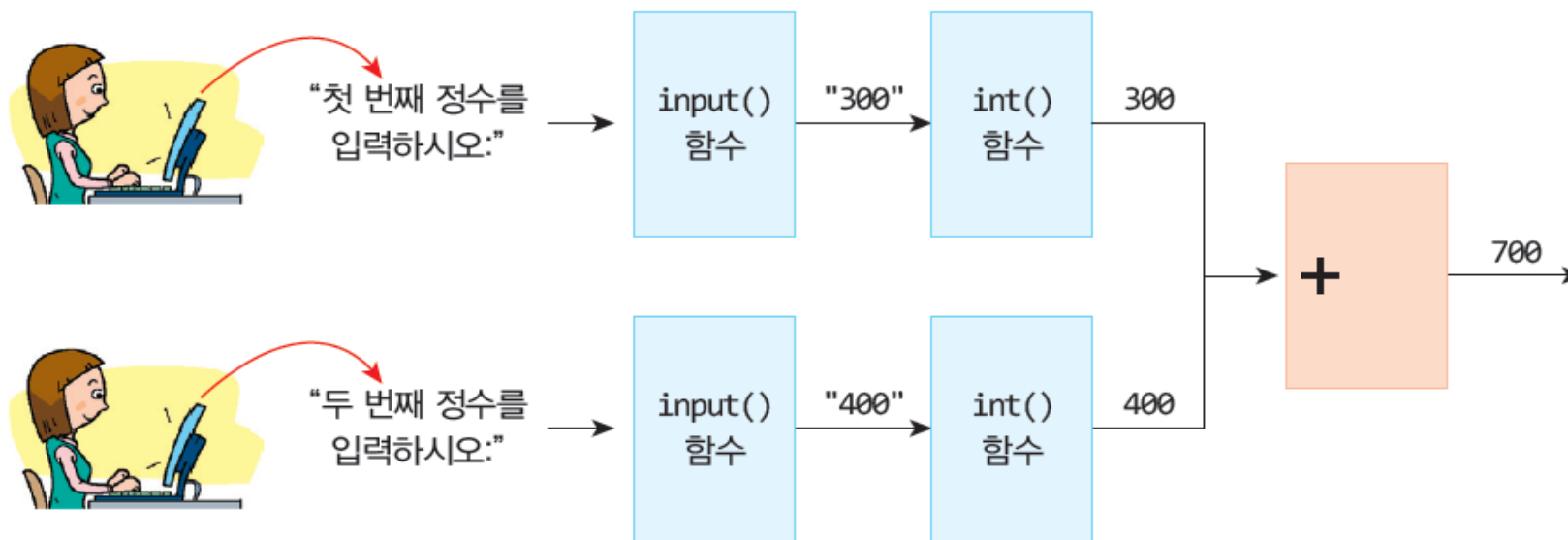
```
a=int(input( “첫번째 수?” ))  
b=int(input( “두번째 수?” ))
```

실습

숫자

```
x = int(input("첫 번째 정수를 입력하시오: "))  
y = int(input("두 번째 정수를 입력하시오: "))  
sum = x + y  
print(x, "과", y, "의 합은", sum, "입니다.")
```

첫 번째 정수를 입력하시오: 300
두 번째 정수를 입력하시오: 400
300 과 400 의 합은 700 입니다.



퀴즈

다음 코드를 실행한 후에 111과 222를 입력하면 어떤 내용이 출력될까요?

```
var1=input()
var2=input()
result=var1+var2
print(var1,"+",var2,"=",result)
```



```
>>>111222
```

```
var1=int(input())
var2=int(input())
result=var1*var2
print(var1,"*",var2,"=",result)
```



```
TypeError: can't multiply sequence  
by non-int of type 'str'
```

(스크립트 모드) 세 과목 점수를 입력받아 평균을 구하는 프로그램

결과화면

수학점수를 입력하시오 : 80

영어점수를 입력하시오 : 70

국어점수를 입력하시오 : 90

당신의 중간고사 평균점수는 80점 입니다.

```
math = int(input("수학점수를 입력하시오 : "))  
eng = int(input("영어점수를 입력하시오 : "))  
kor = int(input("국어점수를 입력하시오 : " ))  
ave = (math+eng+kor)/3  
print("당신의 중간고사 평균점수는 " ,ave, "점입니다.")
```

실습 2/2

첫 번째 정수를 입력하시오: 10

두 번째 정수를 입력하시오: 3

10 의 3 승은 1000 입니다.

```
first = int(input("첫 번째 정수를 입력하시오 : " ))
```

```
second = int(input("두 번째 정수를 입력하시오 : "))
```

```
print(first, '의', second, '은', first**second, '입니다')
```

특수 문자열

- '\n'과 같은 백슬래시를 조합한 코드를 '이스케이프 코드(escape code)'라고 한다.
- 이스케이프 코드는 프로그래밍할 때 사용할 수 있도록 미리 정의해 둔 '문자 조합'이다

특수 문자열	의미
\n	줄바꿈 문자
\t	탭문자
\\	역슬래시 자체
\"	큰따옴표 자체
\'	작은따옴표 자체

```
>>> print("말 한마디로\n천냥빚을 갚는다")
```

말 한마디로

천냥빚을 갚는다

```
>>> print(“말 한마디로  
천냥빚을 갚는다”)
```

말 한마디로

천냥빚을 갚는다



print() 서식

변수와 문자열을 동시에 출력할 때

1

```
x = 100
```

```
y = 200
```

```
print(x, "와 ", y, "의 합=", x+y)
```

```
100 와 200 의 합= 300
```

2

```
print(str(x)+ "와 " +str(y)+ "의 합="+str(x+y))
```

3

```
print(f "{x}와 {y}의 합={x+y}")
```

4

```
print( "%d와 %d 의 합=%d" %(x, y, x+y))
```

f 문자열

```
>>>a = 4.2  
>>>print(f'{a}')
```

4.2

```
x=25  
y=98  
prod=x*y  
print(f'{x}와 {y}의 곱은 {prod}이다')
```




형식화된 출력 %

문자열

```
>>> name = '홍길동'
>>> print(" 저의 이름은 ", name, "입니다.")
저의 이름은 홍길동입니다.
```

- 문자열에 변수의 값을 삽입하여 출력하고 싶으면 %기호 사용
- 서식사용방법 : 앞에 %가 붙음. **%s**는 문자열(String)을 의미.

```
>>> name = '홍길동'
>>> print(" 저의 이름은 %s입니다." % name)
저의 이름은 홍길동입니다.
```

형식화된 출력 %

문자열

- 문자열 형 데이터 서식 지정 : %s

"%s" → P y t h o n

자릿수만큼 출력

"%10s" → 

오른쪽 정렬

열 자리 확보

정수 %d는 정수(Decimal)를 의미.

```
>>> age = 21
>>> print("나이는 ", age, " 살입니다.")
나이는 21살입니다.
```



```
>>> age = 21
>>> print("나이는 %d살입니다" %age)
나이는 21살입니다.
```

```
>>> print(100)
100
>>> print("%d" %100)
100
>>> math=98
>>> print("수학점수는", math, "입니다")
수학점수는 98입니다
>>> print("수학점수는 %d 입니다" %math)
수학점수는 98입니다
```

"%d" → 

숫자의 자릿수만큼 정렬

"%5d" → 

오른쪽에 붙여서 정렬

다섯 자리 확보

"%05d" → 

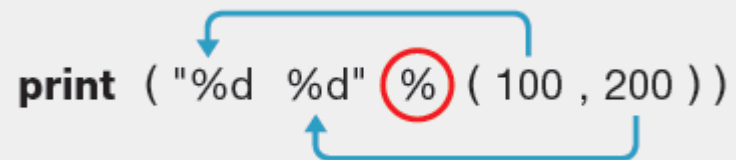
오른쪽에 붙여서 정렬
빈칸을 0으로 채움

다섯 자리 확보

정수

- 둘 이상의 변수를 출력하고 싶을 때

```
print ( "%d %d" % ( 100 , 200 ) )
```



- 서식의 개수와 % 뒤에 나오는 숫자(또는 문자)의 개수가 같아야 함

오류발생



```
print("%d" % (100, 200) )  
print("%d %d" % (100) )
```

- 서식과 숫자형이 일치하지 않을 때 서식에 맞춤(%d는 소수점이하 버림)

```
print("%d / %d = %d" % (100, 200, 0.5))
```

```
100/200=0
```

- %f는 실수(Floating point number)를 의미.

```
>>> width = 8.75
```

```
>>> print("폭은 ",width, "입니다.")
```

폭은 8.75입니다.


```
>>> width = 8.75
```

```
>>> print("폭은 %f입니다" %width)
```

폭은 8.75입니다.


• 실수 형 데이터의 서식 지정

- 두 번째 %7.1f는 소수점 을 포함한 전체 자리인 일곱 자리를 확보하고 소수점 아래는 한 자리만 차지한다는 의미

"%f" →  소수점 아래 여섯 자리까지
무조건 출력

"%7.1f" →  소수점 아래 첫째 자리만 출력
소수점 아래 둘째 자리에서 반올림

일곱 자리 확보

"%7.3f" →  소수점 아래 셋째 자리까지 출력
오른쪽 빈칸은 0으로 채움

일곱 자리 확보

% .3f


```
1 print("%d" % 123)
2 print("%5d" % 123)
3 print("%05d" % 123)
4
5 print("%f" % 123.45)
6 print("%7.1f" % 123.45)
7 print("%7.3f" % 123.45)
8
9 print("%s" % "Python")
10 print("%10s" % "Python")
```

%.3f

```
123
 123
00123
123.450000
 123.5
123.450
Python
  Python
>>> |
```

실습



인하대학교

사용자입력

첫번째 숫자를 입력하시오 30

두번째 숫자를 입력하시오 20

30+20=50 `print("%d+%d=%d" %(a,b,a+b))`

30-20=10

30*20=600

30/20=1.5

```
a= int(input('첫번째 숫자를 입력하시오 '))
```

```
b= int(input('두번째 숫자를 입력하시오 '))
```

```
print('%d+%d=%d' %(a,b,a+b))
```

```
print('%d-%d=%d' %(a,b,a-b))
```

```
print('%d*%d=%d' %(a,b,a*b))
```

```
print('%d/%d=%.1f' %(a,b,a/b))
```

실습

반지름을 입력받아 원의 면적을 구하는 프로그램

원의 면적 = $3.14159 \times \text{반지름} \times \text{반지름}$

결과화면

반지름을 입력하시오 : 10

반지름이 10인 원의 넓이 = 314.16

%.2f

```
r = int(input("반지름을 입력하시오: "))  
area=3.14159*r*r 또는 area=3.14*r**2  
print("반지름이 %d인 원의 넓이=%.2f" %(r,area))
```

```
r = int(input("반지름을 입력하시오: "))  
area=3.14159*r*r 또는 area=3.14159*r**2  
print("반지름이", r, "인 원의 넓이=", round(area,2))
```

- 물건 값의 7.5%가 부가세라고 하자. 물건값이 12345원일 때, 부가세를 소수점 2번째 자리까지 계산하는 프로그램


```
price = 12345  
tax = price * 0.075  
print("%.2f"%tax)
```

```
price = 12345  
tax = price * 0.075  
tax = round(tax, 2)  
print(tax)
```

925.88

f 문자열로 소수점 표현

```
>>>a = 4.2  
>>>print(f'{a}')  
4.2
```



```
>>>a = 4.2  
>>>print(f'{a:5.3f}')  
4.200
```

- 화씨온도를 받아서 섭씨온도로 바꾸는 프로그램.

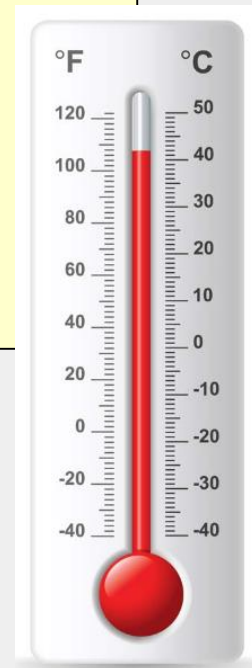


$$C = (F - 32) * \frac{5}{9}$$

사용자입력

화씨온도: 100
섭씨온도: 37.78

```
ftemp = int(input("화씨 온도: "))  
ctemp = (ftemp-32.0)*5.0/9.0  
print("섭씨 온도:%5.2f" %ctemp)
```

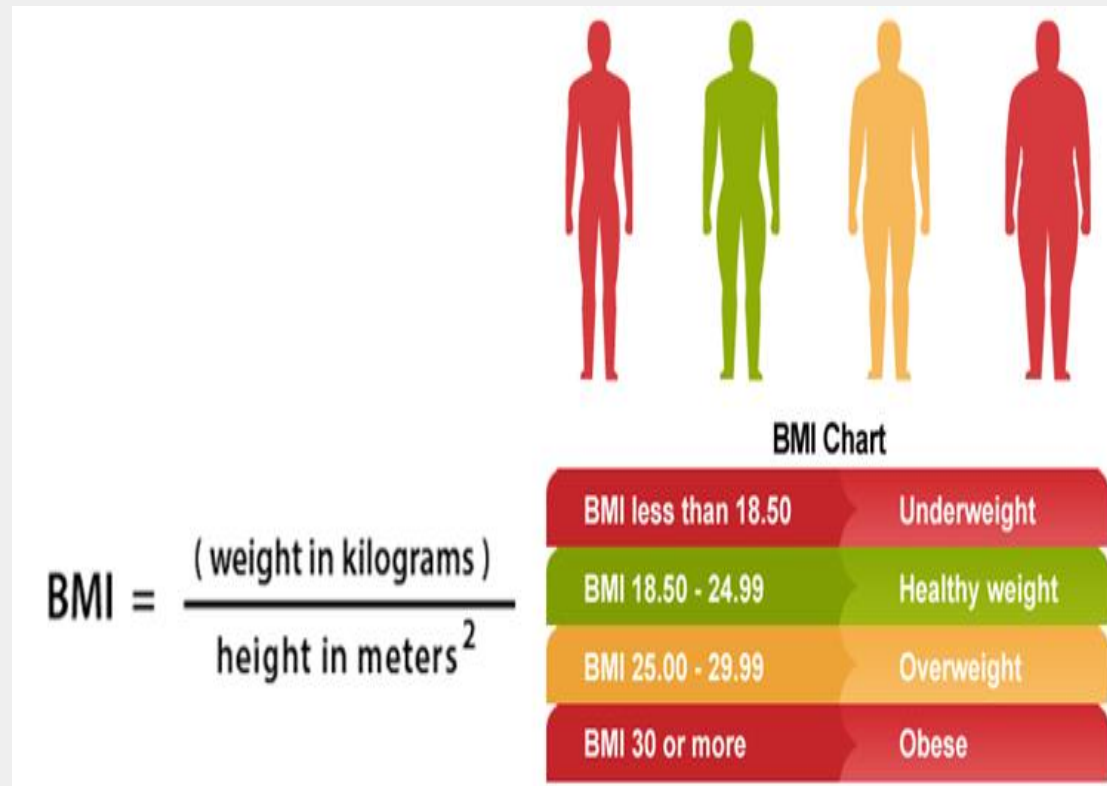


실습 2/2



인하대학교

- 사용자로부터 신장과 체중을 입력받아서 BMI 값을 출력하는 프로그램



몸무게를 kg 단위로 입력하시오: 85.0

키를 m 단위로 입력하시오: 1.83

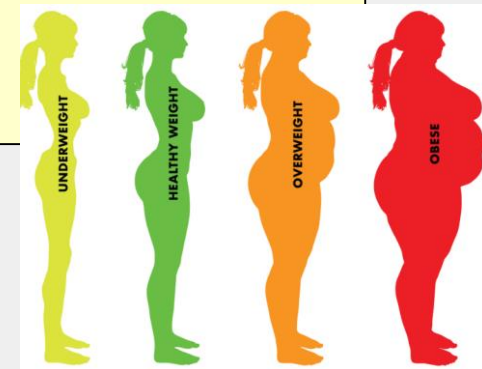
당신의 BMI= 25.38

```
weight = float(input("몸무게를 kg 단위로 입력하시오: "))
```

```
height = float(input("키를 미터 단위로 입력하시오: "))
```

```
bmi = weight / (height**2)
```

```
print("당신의 BMI=%.2f 입니다" %bmi)
```





인하대학교

연산자

연산자

- ❖ 더하기/빼기/곱하기/나누기 등 산술 연산자.
- ❖ 같다/크다/작다 등의 관계 연산자.
- ❖ and/or/not 등의 논리 연산자.



산술 연산

연산자	기호	사용예	결과값
덧셈	+	$7 + 4$	11
뺄셈	-	$7 - 4$	3
곱셈	*	$7 * 4$	28
정수 나눗셈	//	$7 // 4$	1 몫
실수 나눗셈	/	$7 / 4$	1.75
나머지	%	$7 \% 4$	3 나머지

몫과 나머지 연산

```
p = 7  
q = 4  
print("나눗셈의 몫=", p // q)  
print("나눗셈의 나머지=", p % q)
```

나눗셈의 몫= 1
나눗셈의 나머지= 3

응용) 어떤 숫자가 짝수인지 홀수인지 알려면?

응용) 어떤 숫자가 3의 배수인지 알려면?

사용자가 오늘 들어온 빵의 개수를 입력하면

1. 3개씩 나눠주었을 때 최대 몇 명에게 나누어줄 수 있는지,
2. 남은 빵의 개수는 몇 개인지 구하시오

실행결과

빵갯수 : 89

빵을 나누어줄 수 있는 최대인원 : 29

남은 빵 개수 : 2

```
bread = int(input('빵갯수 :'))  
maxCount = bread//3  
rest= bread%3  
print("빵을 나누어줄 수 있는 최대인원", maxCount)  
print("남은 빵 개수 :", rest)
```


할당 연산

$x = y = z = 0$

$x=0; y=0; z=0$

$x, y, z = 10, 20, 30$

한번에 여러 개의 변수값 할당

$x=10; y=20; z=30$

$x, y = y, x$

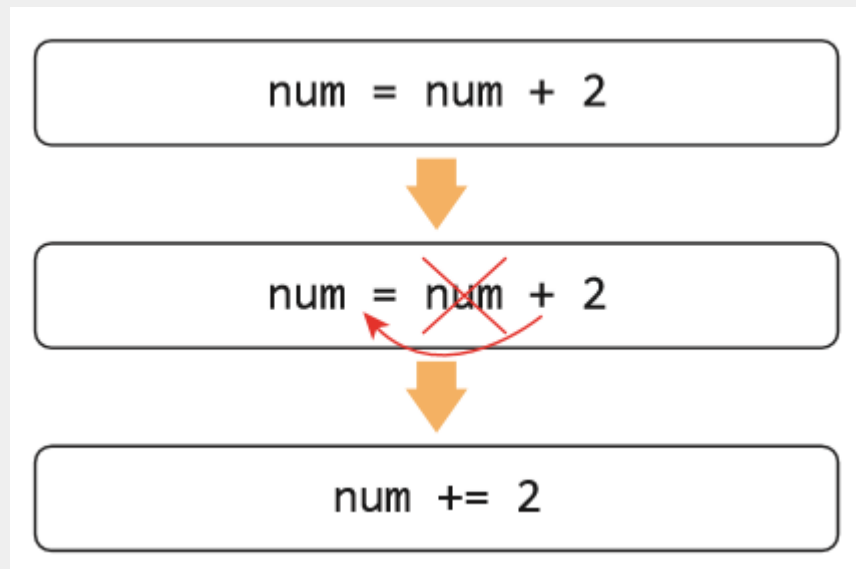
x와 y의 값을 서로 교환한다.

$x=y; y=x$

결과가 다름

복합 연산자 (compound operator)

- $+=$ 처럼 대입 연산자와 다른 연산자를 합쳐 놓은 연산자





복합 연산자

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

복합 연산자

```
x = 1000  
print("초깃값 x=", x)  
x += 2;  
print("x += 2 후의 x=", x)  
x -= 2;  
print("x -= 2 후의 x=", x)
```

```
초깃값 x= 1000  
x += 2 후의 x= 1002  
x -= 2 후의 x= 1000
```

지수 계산

- 지수(power)를 계산하려면 ** 연산자를 사용한다.

```
>>> 2 ** 7  
128
```

- 원리금 계산

```
a = 1000          # 원금  
r = 0.05          # 이자율  
n = 10            # 기간  
result = a*(1+r)**n  # 원리금 합계  
  
print("원리금 합계=", result)
```

```
원리금 합계= 1628.894626777442
```



()
**
*, /
+, -

실습 1/2

복리 계산

- 1626년에 아메리카 인디언들이 뉴욕의 맨하탄섬을 단돈 60길더(약 24달러)에 탐험가 Peter Minuit에게 팔았다고 한다. 382년 정도 경과한 맨하탄 땅값은 약 600억달러라고 한다.
- 하지만 만약 인디언이 24달러를 은행의 정기예금에 입금해두었다면 어떻게 되었을까? 예금 금리는 복리로 8%라고 가정하자. 그리고 382년이 지난 후의 원리금을 계산하여 보자.

공식

$$\text{원금} \times (1 + \text{연이율})^{\text{기간}}$$



```
init_money = 24  
interest = 0.08  
years = 382  
print(init_money*(1+interest)**years)
```

```
140632545501736.62
```

140조달러

- 자동 판매기를 시뮬레이션하는 프로그램을 작성하여 보자.

물건값을 입력하시오: 750

1000원 지폐개수: 1

500원 동전개수: 0

100원 동전개수: 0

500원 = 0 100원 = 2 10원 = 5 1원 = 0



코딩

```
itemPrice = int(input("물건값을 입력하시오: "))
note = int(input("1000원 지폐개수: "))
coin500 = int(input("500원 동전개수: "))
coin100 = int(input("100원 동전개수: "))

change = note*1000 + coin500*500 + coin100*100 - itemPrice

# 거스름돈(500원 동전 개수)을 계산한다.
nCoin500 = change//500
change = change%500

# 거스름돈(100원 동전 개수)을 계산한다.
nCoin100 = change//100
change = change%100

# 거스름돈(10원 동전 개수)을 계산한다.
nCoin10 = change//10
change = change%10

# 거스름돈(1원 동전 개수)을 계산한다.
nCoin1 = change

print("500원=", nCoin500, "100원=", nCoin100, "10원=", nCoin10, "1원=", nCoin1)
```



연산자

- ❖ 더하기/빼기/곱하기/나누기 등 산술 연산자.
- ❖ 같다/크다/작다 등의 관계 연산자.
- ❖ and/or/not 등의 논리 연산자.

관계연산자

- 어떤 것이 큰지, 작은지, 같은지를 비교하는 것, 결과는 참(True)이나 거짓(False)
- 주로 조건문(if)이나 반복문(for, while)에서 사용

$a < b = \begin{cases} \text{참} & : \text{True} \\ \text{거짓} & : \text{False} \end{cases}$

관계 연산자	의미	설명
==	같다	두 값이 동일하면 참
!=	같지 않다	두 값이 다르면 참
>	크다	왼쪽이 크면 참
<	작다	왼쪽이 작으면 참
>=	크거나 같다	왼쪽이 크거나 같으면 참
<=	작거나 같다	왼쪽이 작거나 같으면 참

관계연산자

- 관계 연산자 예

```
a,b = 100,200  
print(a == b, a != b, a > b, a < b, a >= b, a <= b)
```

출력 결과

```
False True False True False True
```

- a와 b를 비교하기 위한 관계 연산자 ==를 사용시 =을 하나만 쓰는 경우 → 오류발생
a =b는 b의 값을 a에 대입하라는 의미이지 관계 연산자가 아님

```
print(a=b)
```

실습



인하대학교

```
score = int(input("점수를 입력하시오"))  
print(score>=80)
```

실행결과

```
점수를 입력하시오 95  
True
```

```
점수를 입력하시오 70  
False
```

논리연산자

논리 연산자	의미	설명	사용 예
and	~이고, 그리고(AND)	둘 다 참이어야 참	$(a > 100) \text{ and } (a < 200)$
or	~이거나, 또는(OR)	둘 중 하나만 참이어도 참	$(a == 100) \text{ or } (a == 200)$
not	~아니다, 부정(NOT)	참이면 거짓, 거짓이면 참	$\text{not}(a < 100)$

입력값		A 그리고 B	A 또는 B	A가 아니다
A	B	(A and B)	(A or B)	(!A)
True	True	True	True	False
True	False	False	True	False
False	True	False	True	True
False	False	False	False	True



논리연산자

`a = 99`

`(a > 100) and (a < 200)`

`(a > 100) or (a < 200)`

`not(a == 100)`

출력 결과

False True True

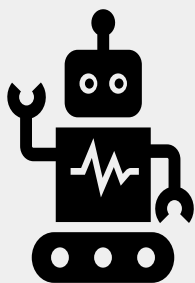
논리연산자

나이가 20세 이상인 인천시민 찾기

```
age = int(input("나이를 입력하시오"))  
city = input("거주지역을 입력하시오"))  
print(age>=20 and city=='인천')
```

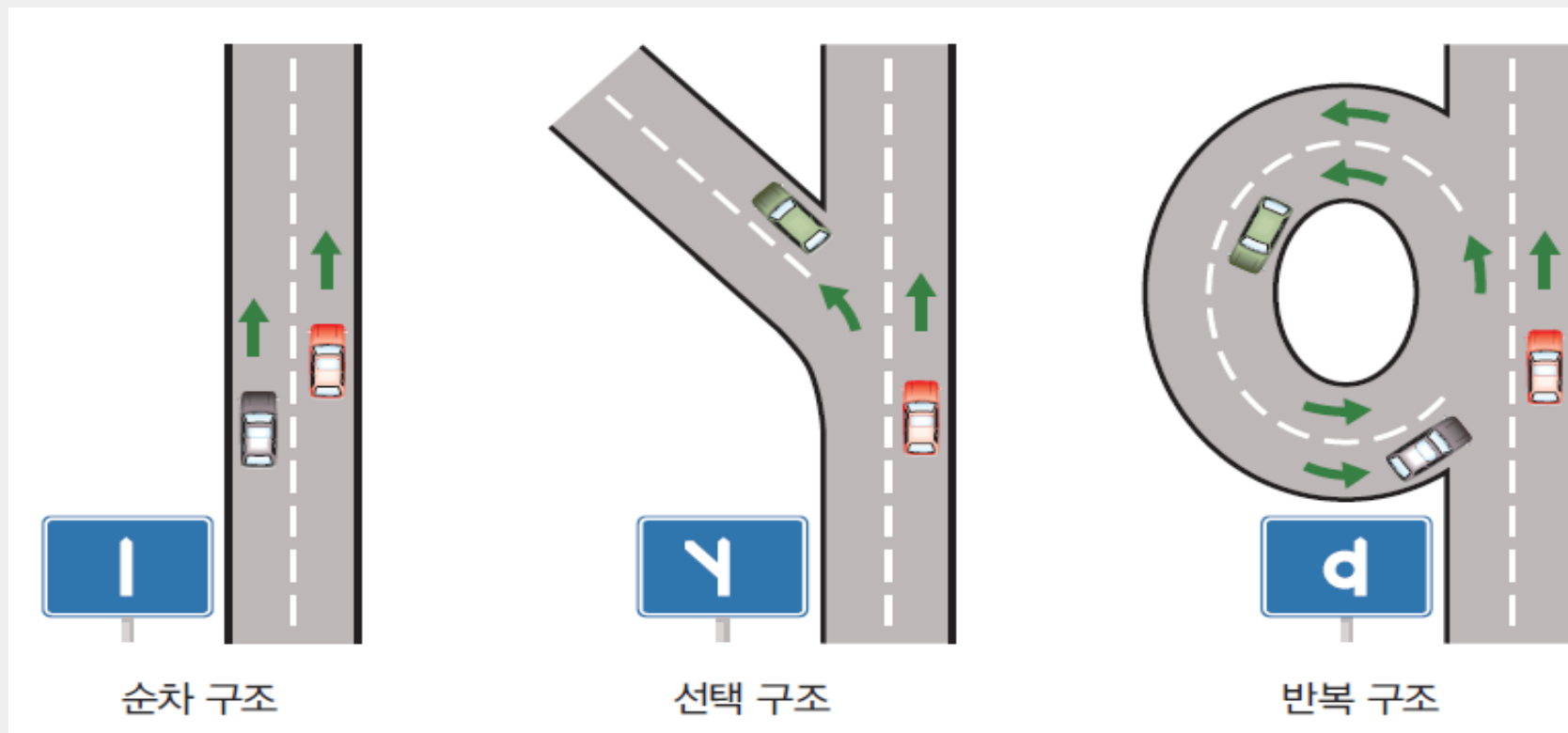
실행결과

```
나이를 입력하시오 25  
거주지역을 입력하시오 서울  
False
```

제어문

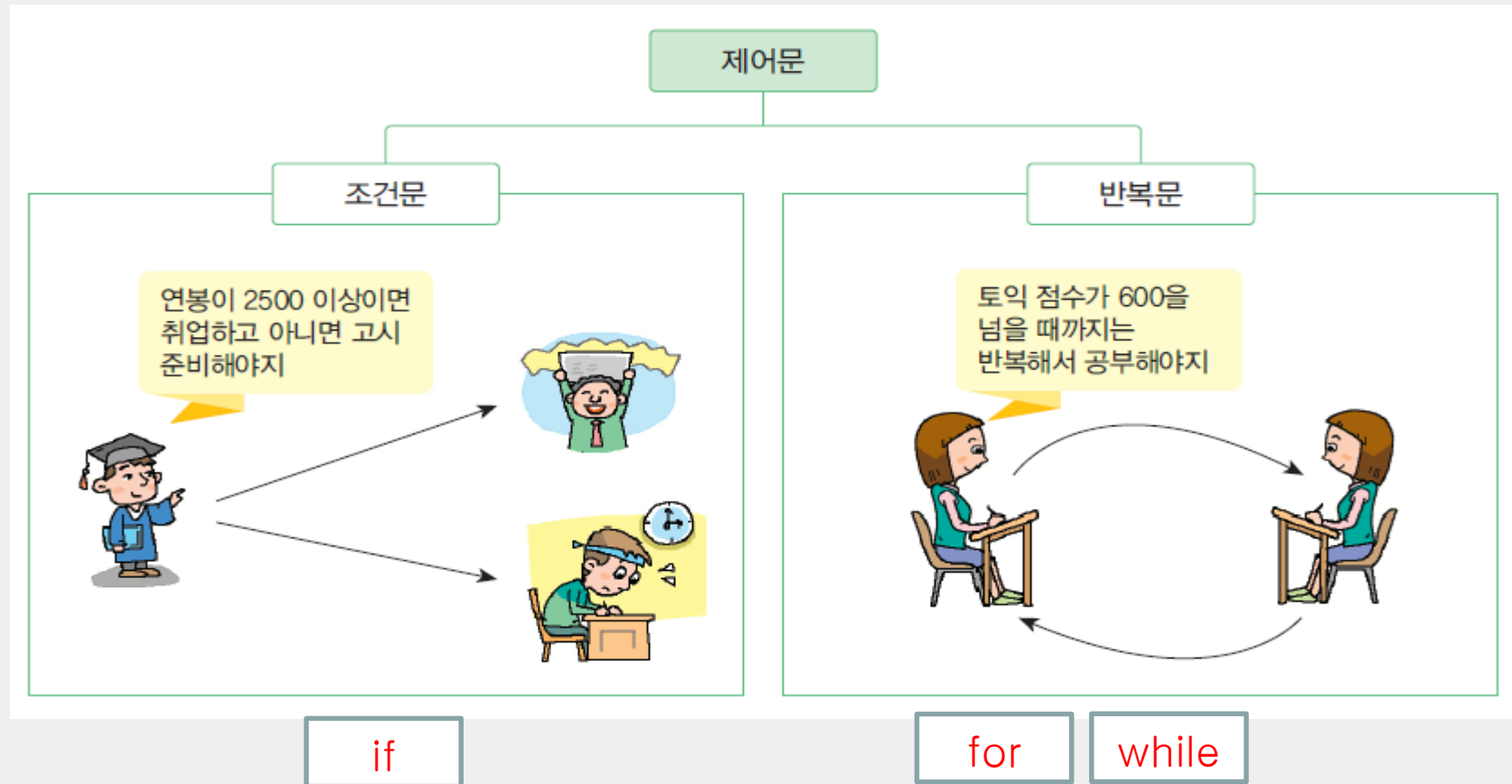
기본 제어구조



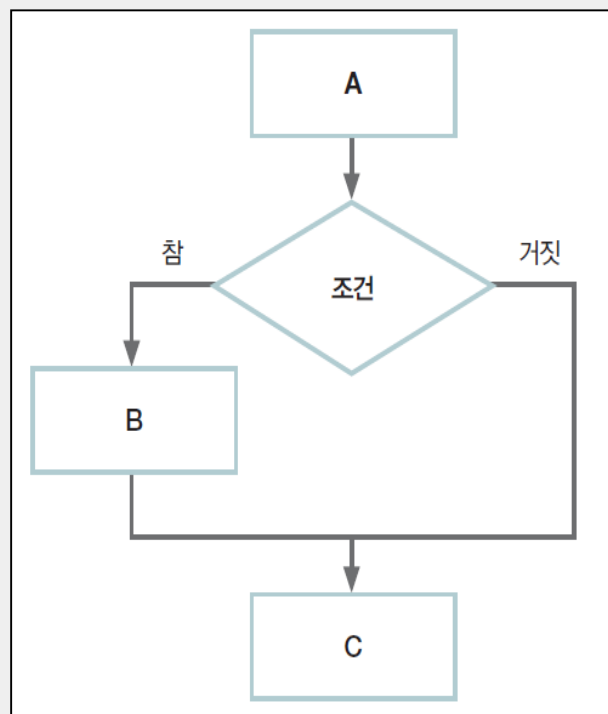
제어문



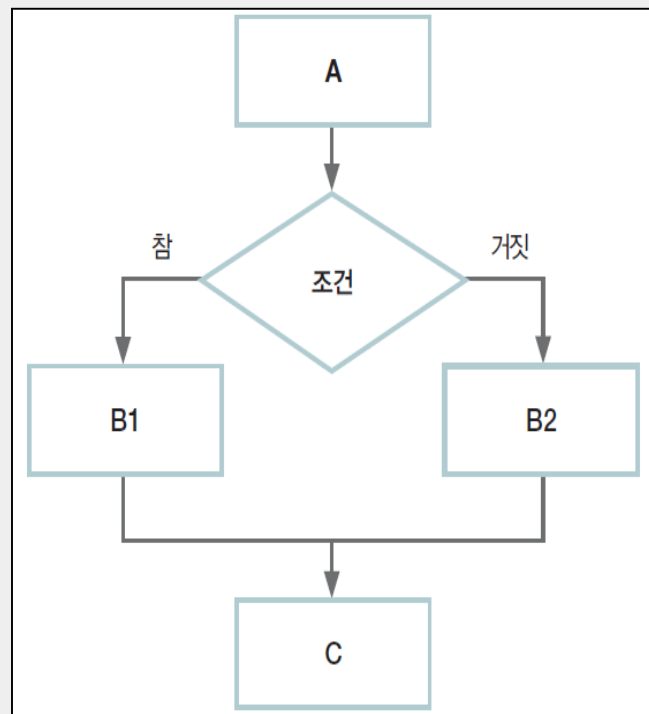
인하대학교



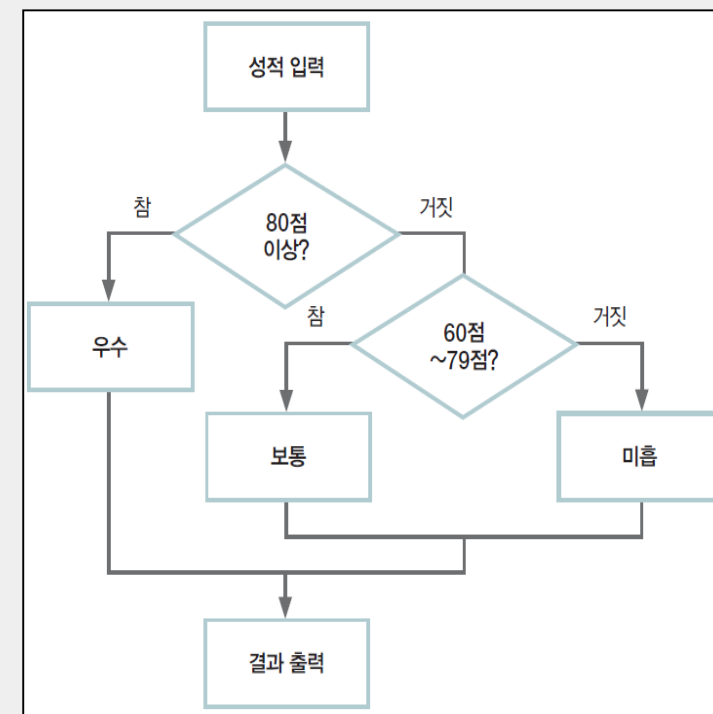
조건문



단순선택



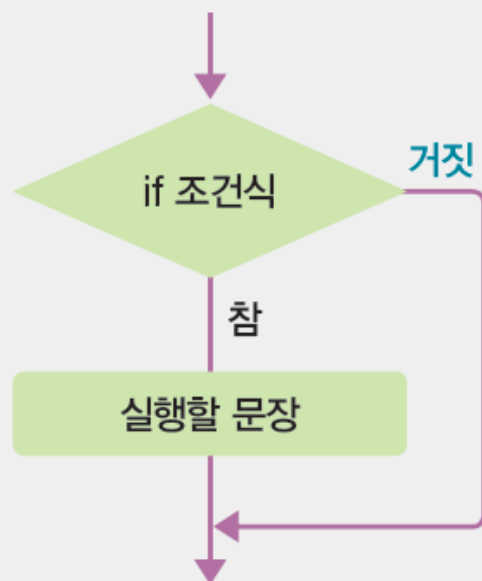
양자택일



다중선택

기본 if문

- **if 조건식** : 조건식이 참이면 실행할 문장이 처리되고, 거짓이면 아무것도 실행하지 않고 종료



```
if 조건:  
    명령1  
    명령2  
    ...  
프로그램 계속
```

짝수 판단


정수를 입력하시오: 10
짝수입니다.
끝!

정수를 입력하시오: 7
끝!

```
num = int(input("정수를 입력하시오: "))  
if num % 2 == 0 :  
    print("짝수입니다.")  
print('끝')
```

오류

```
num = int(input("정수를 입력하시오: "))  
if num % 2 == 0 :  
    print("짝수입니다.")  
print('끝')
```

```
num = int(input("정수를 입력하시오: "))  
if num % 2 == 0 :  
      
print('끝')
```

```
num = int(input("정수를 입력하시오: "))  
if num % 2 == 0 :  
    pass  
print('끝')
```

결과화면(15세 이상 입장)

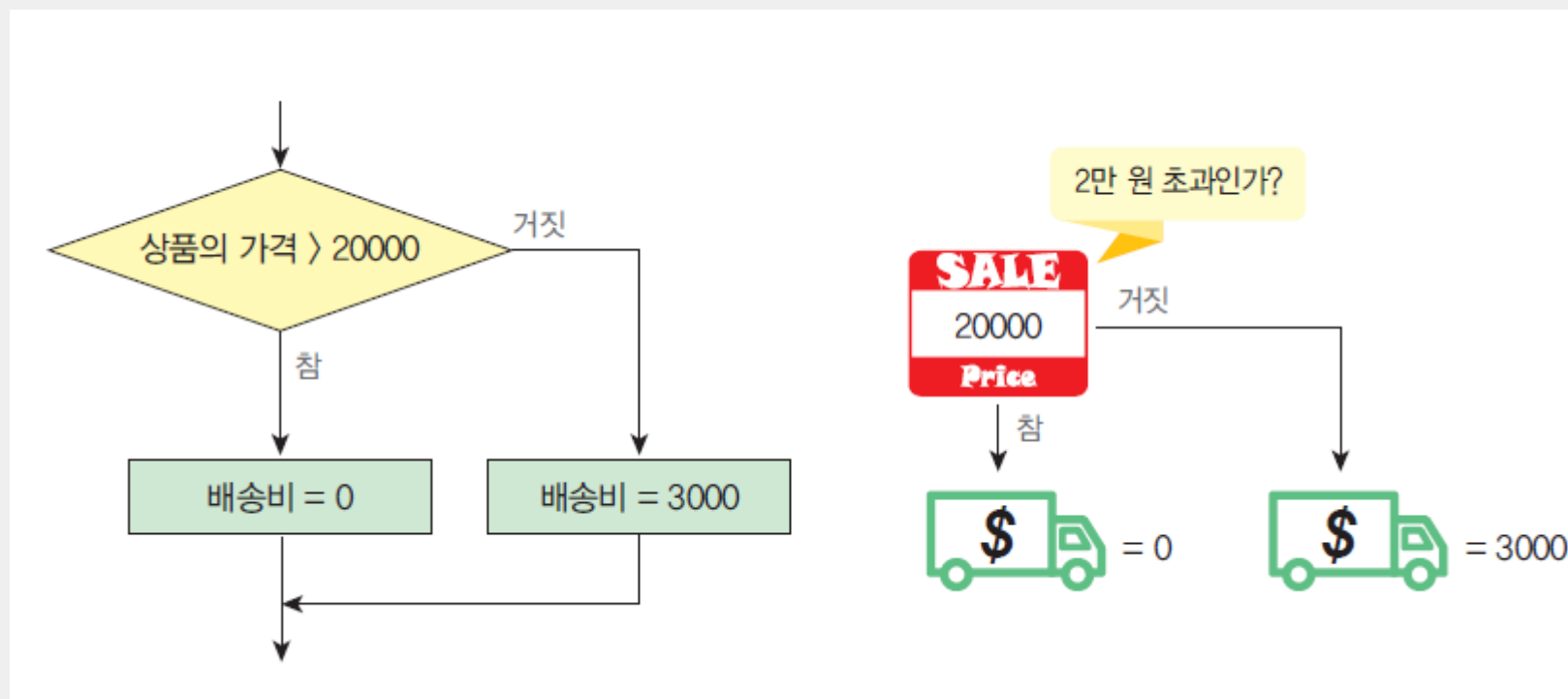
나이를 입력하세요: 13
15세가 안되었군요~ 입장불가입니다.
끝

나이를 입력하세요: 17
끝

```
age = int(input("나이를 입력하세요: "))  
  
if age < 15 :  
  
    print("15세가 안되었군요~ 입장불가입니다")  
  
print("끝")
```


if-else 문

* 참일 때 실행하는 문장과 거짓일 때 실행하는 문장이 다를 때 사용



if-else 문

Syntax: if-else 문

형식 if 조건식 :
 문장1
 else :
 문장2

참이나 거짓으로 계산되는 조건식,
관계 연산자 == != < > <= >= 을 사용한다.

예 if price > 20000 :

콜론(:)은 복합문을 의미한다.

shipping_cost = 0

조건식이 참이면 실행되는 문장

else :

shipping_cost = 3000

조건식이 거짓이면 실행되는 문장

else 절은 생략될 수도 있다.

if와 else는 같은 위치여야 한다.

배송비 계산 프로그램



인하대학교

상품의 가격: 30000
배송비 = 0

```
# 사용자로부터 상품의 가격을 입력받는다.  
price = int(input("상품의 가격: "))
```

```
# 배송비를 결정한다.  
if price > 20000 :  
    shipping_cost = 0  
else :  
    shipping_cost = 3000
```

```
# 배송비를 출력한다.  
print("배송비 = ", shipping_cost)
```

```
price = int(input("상품의 가격: "))
```

```
if price > 20000 :  
    shipping_cost = 0  
    print("배송비 = ", shipping_cost)  
else :  
    shipping_cost = 3000  
    print("배송비 = ", shipping_cost)
```



실습 1/3 : 5의 배수

```
num = int(input("정수를 입력하시오: "))  
if num % 5 == 0 :  
    print("5의 배수입니다. ")  
else:  
    print("5의 배수가 아닙니다.")
```



실습2/3

- 사용자로부터 3개의 점수(0에서 100점 사이)를 입력받아 평균을 계산한다.
- 평균점수가 60점 이상이면 “축하합니다 합격입니다”
- 60점 미만이면 “아쉽군요 불합격입니다”를 출력하는 프로그램

```
score1 = int(input("점수1: "))
score2 = int(input("점수2: "))
score3 = int(input("점수3: "))
ave=(score1+score2+score3)/3
if ave >=60 :
    print("축하합니다 합격입니다")
else:
    print("아쉽군요 불합격입니다")
```



실습3/3

사용자로부터 아이디를 받아서 프로그램에 저장된 아이디와 일치하는지 여부를 출력하는 프로그램을 작성

결과화면

아이디를 입력하시오: python
환영합니다.

아이디를 입력하시오: ruby
아이디를 찾을 수 없습니다.

```
id = "python"
s = input("아이디를 입력하시오: ")
if s == id:
    print("환영합니다.")
else:
    print("아이디를 찾을 수 없습니다.")
```