



## 변수 할당

오른쪽 값을 왼쪽 변수에 저장

a = 3	a에 3 저장
a=b=c=0	a,b,c에 0 저장
a+=3	a=a+3
a=None	undefined
del a	a 변수 삭제

## 기본 자료형

int	72	정수형
float	9.2	실수형
bool	True False	논리형
str	"Hi" 'Sohi'	문자형

## 자료형()

() 안 값을 해당 자료형으로 변환

int("15")	15
int(15.5)	15
float("3.2")	3.2
str(5)	"5"

## type(expression)

() 안 값의 자료형 알아냄

type(5)	Int
type(15.5)	float
type(True)	bool
type("15")	str

## print

() 안에 있는 데이터를 출력,  
sep=" " (item separator), end="\n"(end of print)

print("hello")	hello
print("hi","sohi")	hi sohi
print("a"+"b")	ab
print("a",end=",") print("b")	a,b
print("a","b","c",sep=":")	a:b:c

## input

input(): 데이터가 문자(str)로 입력됨  
input('메시지'): 사용자에게 메시지 보여줌

a=input()	사랑
a=input("좋아하는 음식")	김치
a=input("숫자입력") b=input("숫자입력") print(a+b)	4 2 42

## 비교연산자

<	작다
<=	작거나 같다
==	같다
>=	크거나 같다
>	크다
!=(or<>)	다르다

## 산술연산자

3+1	4	덧셈
3-1	2	뺄셈
3*2	6	곱셈
3**2	9	거듭제곱
3/2	1.5	나눗셈
3//2	1	몫
3%2	1	나머지

## 논리연산자

and	모두 True면 True
or	하나라도 True면 True
not	True면 False로 변경 False면 True로 변경

## 변수명 규칙

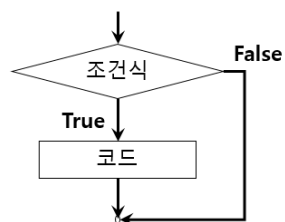
영문, 숫자, \_ 사용  
대소문자 구분됨  
특수문자, 키워드 사용안됨  
숫자로 시작할 수 없음

## Indentation

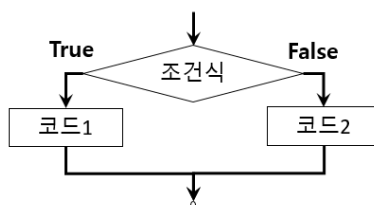
블록 들여쓰기 조절방법

증가	:
유지	Enter 키
감소	Backspace키

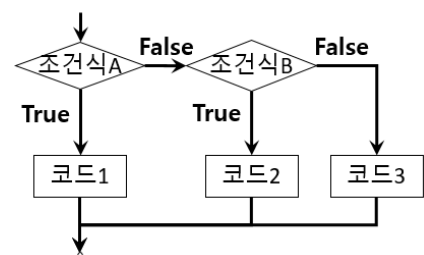
**if** 조건식:  
코드



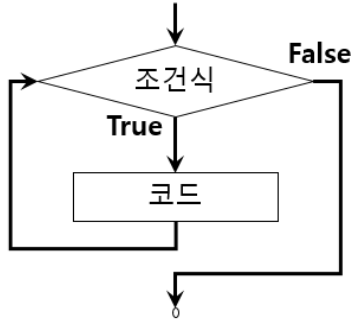
**if** 조건식:  
코드1  
**else** :  
코드2



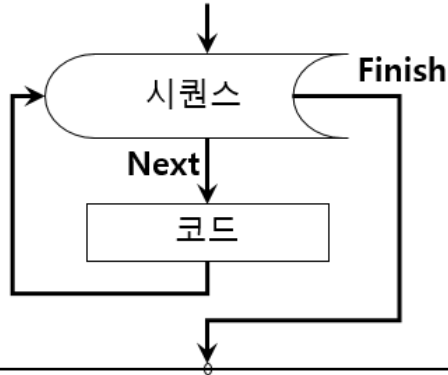
**if** 조건식A:  
코드1  
**elif** 조건식B:  
코드2  
**else** :  
코드3



**while** 조건식:  
반복할 코드



**for** 조건식 **in** 시퀀스:  
반복할 코드



range()

range([start,] stop[, step]) :  
start부터 stop-1까지  
step만큼 증가하는 범위

```
for x in range(5) :
    print(x)
```

0~4까지 범위의 숫자를  
하나씩 꺼낸 후  
반복할 코드 실행

```
for x in [0,1,2,3,4] :
    print(x)
```

0~4까지 출력

## 집합 자료형

리스트	list	[4,2, 'x']	동일한 목적을 갖는 유사한 항목들의 묶음 서로 다른 형인 경우도 허용함
튜플	tuple	(4,2)	원소를 변경할 수 없는 리스트
문자열	str	'Hello'	문자가 모인 문자열
사전	dict {"key" : "value"}	{a=3, b=4, k="v"}	사전과 비슷한 형식의 자료형 원소는 한 쌍의 key:value로 표현
집합	set {"key1", "key2"}	{1,9,3,0}	수학에서의 집합, 원소들 간에 순서가 없으며, 중복 을 허용하지 않음

a[0] a[1] a[2] a[3] a[4]  
a 10 20 30 40 50  
a[-5] a[-4] a[-3] a[-2] a[-1]

## 인덱싱

리스트[인덱스] : 리스트 요소에 접근할 때 인덱스를 지정한다.

a[3]	40
a[-1]	50

## 슬라이싱

리스트[시작인덱스:끝인덱스:증감] :  
시작인덱스~끝인덱스-1 요소값을 가져온다.

a[0:3]	[10, 20, 30]
a[:]	[10, 20, 30, 40, 50]
a[::-2]	[50, 30, 10]

## 문자열 함수

s = 'A B C' print(s.split())	['A', 'B', 'C']
date = '1907-10-01' split_date = date.split('-') print(split_date[1])	10

```
lst=[]
```

```
for n in range(1,31) :
    if(n%9==0) :
        lst.append('짝짝짝')
    elif(n%6==0) :
        lst.append('짝짝')
    elif(n%3==0) :
        lst.append('짝')
    else :
        lst.append(n)
print(lst)
```

[1, 2, '짝', 4, 5, '짝짝', 7, 8, '짝짝짝', 10, 11, '짝짝', 13, 14, '짝', 16, 17, '짝짝짝', 19, 20, '짝', 22, 23, '짝짝', 25, 26, '짝짝짝', 28, 29, '짝짝']

## txt 파일

open() 함수 : 파일을 여는 함수  
close() 함수 : 파일을 닫는 함수  
read() 함수 : 전체 내용 읽어오기  
encoding : UTF-8 / CP949 / EUC-KR

```
f = open('data.txt', encoding='utf8')
print(f.read())
f.close()
```

## csv 파일

csv.reader(csvfile) : csvfile에서 읽기  
csv.writer(csvfile) : csvfile에 쓰기

```
import csv
f = open('data.csv')
data = csv.reader(f)
for row in data :
    print(row)
f.close()
```

## csv 파일 사용 예

```
import csv
import datetime

# 파일 내용 읽어오기
file_name="2015univ.csv"
file = open(file_name,'r',encoding='utf8')
csv_file = csv.reader(file)

# 제목을 저장한다.
header = next(csv_file)
# 데이터를 리스트로 저장한다.
data = list(csv_file)

# 파일을 닫는다.
file.close()

# 제목 출력하기
print(header)
# 데이터 리스트 출력하기
for row in data :
    print(row)
```

## plot함수

값을 서로 연결해서 라인 형태의 그래프를 그리는 함수

```
import matplotlib.pyplot as plt
data = [1,2,3,4]
plt.plot(data)
plt.show() #그래프 보여주기
```

```
# 제목 표시
plt.title('title')
```

```
# 폰트 지정
plt.rc('font', family='Malgun Gothic')
```

```
# 범례 표시
plt.plot([4,3,2,1], label = 'type B')
plt.legend()
```

```
# 범례 위치 지정
Location String : best / upper right / upper left /
lower left / lower right / right / center left / center
right / lower center / upper center / center
plt.legend(loc='upper right')
```

```
# 파일 저장
plt.savefig('data.png')
```

```
# 그래프 크기 설정
plt.figure(figsize = (4,2), dpi = 150)
```

## 다양한 그래프

```
import matplotlib.pyplot as plt
# 선(line)
plt.plot([1,2,3,4])
# 막대(bar)
plt.bar([1,2,3],[10,20,30])
# 히스토그램(histogram)
plt.hist([1,1,2,4,5,7,8,9], bins=9)
# 상자 수염 그림(boxplot)
plt.boxplot([2,7,10,9,5,4,6,8])
# 산포도 (scatter)
plt.scatter([10,30,20,25,40],[25,38,47,29,38])
# 원 그래프(pie)
plt.pie([10,30,25,40])
```

## 간단한 그래프 사용 예

```
import matplotlib.pyplot as plt
plt.figure(figsize = (4,2), dpi = 150) # 그래프 크기 및 해상도 조절
plt.rc('font',family='Malgun Gothic') # 한글 폰트설정
plt.plot([1, 2, 3, 4], label = 'type A')
plt.plot([4, 3, 2, 1], label = 'type B')
plt.title('제목') # 제목 넣기
plt.xlim(0,3) # x축 값 범위 : 0~3
plt.ylim(1,4) # y축 값 범위 : 1~4
plt.xticks(range(4),['가','나','다','라']) # x축 내용
plt.yticks(range(5)) # y축 내용
plt.xlabel('x축') # x축 레이블
plt.ylabel('y축') # y축 레이블
plt.legend() # 범례
plt.savefig('data.png') # 파일 저장
plt.show() # 그래프 보여주기
```