

1. 숫자 입력&출력

1.1 한 줄에 하나씩 숫자 입력받기

코드 1-1 한 줄에 하나씩 숫자 입력받기

```
a=int(input('a입력:'))
b=int(input('b입력:'))
c=int(input('c입력:'))
```

```
print(a,b,c,a+b+c)
```

```
a입력:3
b입력:6
c입력:9
3 6 9 18
```

- 만약에 int()로 감싸지 않으면 문자열로 처리합니다.

코드 1-2 한 줄에 하나씩 숫자 입력받기

```
a=input('a입력:')
b=input('b입력:')
c=input('c입력:')
```

```
print(a,b,c,a+b+c)
```

```
a입력:3
b입력:6
c입력:9
3 6 9 369
```

1.2 한 줄에 여러 개의 숫자 입력받기

코드 1-3 한 줄에 여러 개의 숫자 입력받기

```
a,b,c = map(int,input('a b c값 입력').split())
```

```
print(a,b,c,a+b+c)
```

```
a b c값 입력3 6 9
3 6 9 18
```

1.3 문자1.split(문자2) : 문자2를 기준으로 문자1을 자르기

- split() 메서드 안에 지정한 문자열을 기준으로 문자열을 자릅니다.
- split() 메서드 안에 지정한 문자열이 공백이면 공백을 기준으로 문자열을 자릅니다.

코드 1-4 split() 메서드

```
text = input('날짜 입력 yyyy.mm.dd')
```

```
y,m,d = text.split('.')
```

```
print(text, y, m, d)
```

```
날짜 입력 yyyy.mm.dd2020.02.29
2020.02.29 2020 02 29
```

1.4 map(함수, 집합 형태-iterable객체)

- map() 함수의 첫 번째 인자는 함수, 두 번째 인자는 집합 형태가 들어갑니다.
- 예를 들어 ['1','2','3']에서 '1'을 꺼내서 int 함수를 적용하고, '2'를 꺼내서 int 함수를 적용하고, '2'를 꺼내서 int 함수를 적용합니다. 각각 1, 2, 3을 a, b, c 변수에 대입합니다.

코드 1-5 map() 함수

```
a,b,c = map(int, ['1','2','3'])
```

```
print(a,b,c,a+b+c)
```

```
1 2 3 6
```

- 코드 1-5를 다음과 같이 하나씩 적용해서 볼 수 있습니다.

코드 1-6 map() 함수를 하나씩 적용해서 보기

```
text = input('a b c 값 입력')
print(text)
```

```
text = text.split()
print(text)
```

```
a,b,c = map(int, text)
print(a,b,c,a+b+c)
```

```
a b c 값 입력 3 6 9
3 6 9
['3', '6', '9']
3 6 9 18
```

1.5 숫자 출력하기

- 변수에 있는 값을 콤마로 분리해서 출력합니다.
- 두 번째 print()는 계산 결과가 아닌 작성한 문자열 그대로 출력된 것입니다.

코드 1-7 숫자 출력하기

```
x=3
y=5
print(x, y, x+y)
print('3과 5의 합은 8이다.')
```

```
3 5 8
3과 5의 합은 8이다.
```

코드 1-8 숫자와 문자 함께 출력하기 (1) 콤마 & 자료형 변환

```
x=3
y=5

print(str(x)+'과'+str(y)+'의 합은 '+str(x+y)+'이다.')
```

```
3과5의 합은 8이다.
```

- end=""은 줄 바꿈을 없앱니다.

코드 1-9 숫자와 문자 함께 출력하기 (2) end=""

```
x=3
```

```
y=5
```

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

```
print('과', end='')
```

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

```
print('의 합은', end='')
```

```
print(x+y, end='')
```

```
print('이다.')
```

3과5의 합은8이다.

- {}에는 format() 메서드안의 인자가 순서대로 대입됩니다.
따라서 첫 번째{}에는 3, 두 번째 {}에는 5, 세 번째 중괄호에는 x+y의 결과 8이 대입됩니다.

코드 1-10 숫자와 문자 함께 출력하기 (3) format()

```
x=3
```

```
y=5
```

```
print('{}과 {}의 합은 {}이다.'.format(x,y,x+y))
```

3과 5의 합은 8이다.

2. 연산자 우선순위

산술 연산자 > 관계 연산자 > 논리 연산자

<code>a ** b</code>	지수	<code><</code>	작다	<code>not x</code>	x가 참이면 거짓 x가 거짓이면 참
<code>a * b</code> <code>a / b</code> <code>a // b</code> <code>a % b</code>	곱하기 나누기 몫 나머지	<code><=</code>	작거나 같다	<code>x and y</code>	x, y 모두 참이어야 참
<code>a + b</code> <code>a - b</code>	더하기 빼기	<code>></code>	크다	<code>x or y</code>	x나 y 둘 중 하나만 참이면 참
		<code>>=</code>	크거나 같다		
		<code>==</code>	같다		
		<code>!=</code>	같지 않다		

Ex) $2*5 > 2+5$ and not $3*3 > 10$ → True

3. 수학 관련 함수

3.1 반올림

- `round(a)`, `round(a,b)`

코드 3-1 `round()` 함수

```
print(round(3.33))    # 5보다 작아서 버림
print(round(3.66))    # 5보다 커서 반올림
print(round(3.66, 1)) # 소수 첫째 자리까지 반올림
```

```
3
4
3.7
```

3.2 절대값

- `abs(a)`

코드 3-2 `abs()` 함수

```
print(abs(3))  
print(abs(-3))
```

```
3  
3
```

3.3 제곱

- `pow(a,b)`

코드 3-3 `pow()` 함수

```
print(pow(3,2))  # 3의 2제곱  
print(3**2)     # 3을 2번 곱한다.
```

```
9  
9
```

3.4 나눗셈

- `divmod(7,2)`

코드 3-2 `divmod()` 함수

```
x,y = divmod(7,2)  
print(x)  # 몫  
print(y)  # 나머지
```

```
3  
1
```

3.5 최대값

- `max(a,b,c,d,...)`

코드 3-5 `max()` 함수

```
print(max(7,5,1,3))
```

7

3.6 최소값

- `min(a,b,c,d,...)`

코드 3-6 `min()` 함수

```
print(min(7,5,1,3))
```

1

3.7 합

- `sum(집합 형태: Iterable)`

코드 3-7

```
print(sum([7,5,2,3]))
```

17