

# 파이썬의 내장 자료형



## 1 자료형이란?



프로그래밍이란 **자료(Data)**를 **처리**하는 일을  
주로 함



파이썬에서는 자료를 손쉽게 다룰 수 있도록  
**내장 자료형을 제공**

### 숫자(수치) 자료형

정수(int),  
실수(float),  
복소수(complex)

### 불(Bool) 자료형

True, False

### 군집 자료형

문자열(str),  
리스트(list),  
튜플(tuple),  
사전(dict),  
집합(set)

# 파이썬의 내장 자료형



## 2 내장 자료형의 특징



### 내장 자료형의 구분

1

기억장소의 크기

2

저장되는 데이터의 형태

3

저장 방식

4

값의 범위



# 파이썬의 내장 자료형

## 2 내장 자료형의 특징



### 내장 자료형의 구분

분류 기준	종류
데이터 저장 방법	직접 표현, 시퀀스, 매핑
변경 가능성	변경 가능, 변경 불가능
저장 개수	리터럴(한 가지), 컨테이너(여러가지 저장)

#### 직접 표현

직접 데이터를  
표현(정수, 실수 등)

#### 시퀀스

여러 데이터를 포함  
(순서가 있음)

#### 매핑

여러 데이터를 포함  
(순서가 없음)



# 파이썬의 내장 자료형

## 2 내장 자료형의 특징



동적 자료형 지원 →

프로그래머가 자료형을 직접 설정할 필요가 없음

### C언어

같은 숫자라고 해도  
int, short, unsigned int,  
float, double, long 등  
메모리나 표현 방식 등에  
따라 세분화

### 파이썬

데이터를 입력하면  
데이터 타입을 알아낸 후  
입력된 데이터의 특성에  
맞는 영역을 확보하여 줌

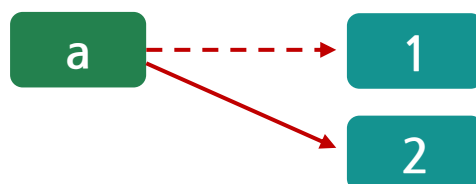
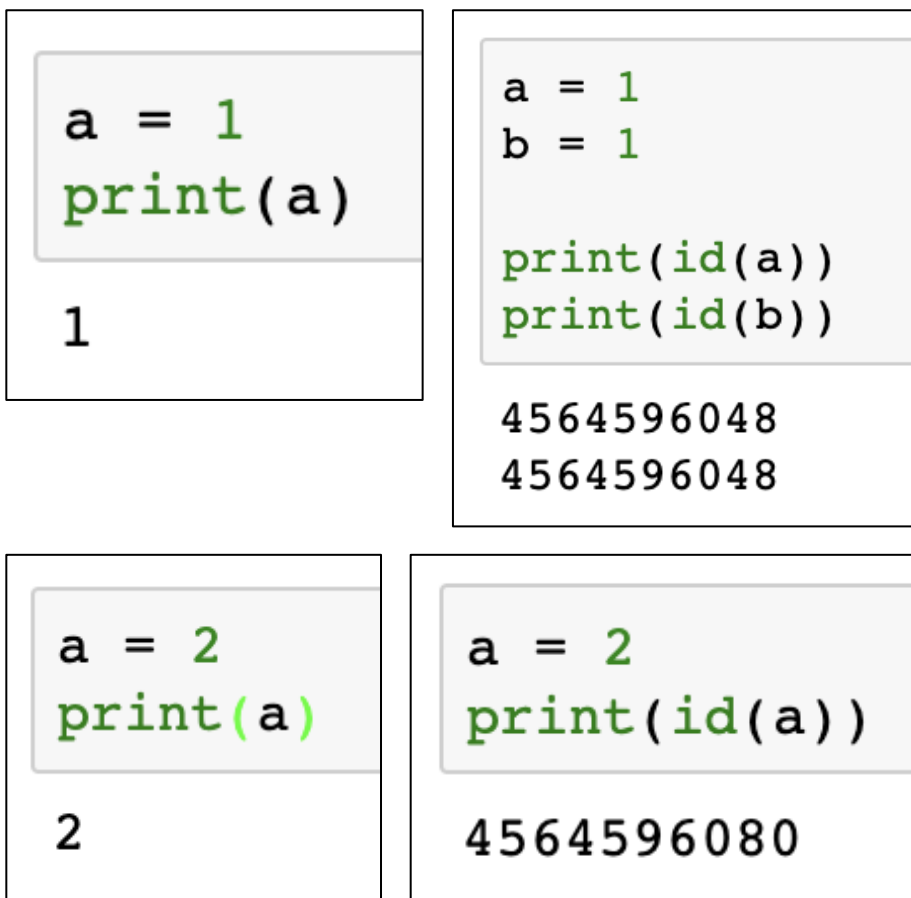
# 파이썬의 내장 자료형

## 2 내장 자료형의 특징



객체지향형 언어로 자료형 또한  
객체와 레퍼런스(Reference)로 관리

- a라는 객체는 1이라는 객체의 레퍼런스를 갖고 있다가 2라는 객체의 레퍼런스로 값을 변경하게 됨
- 프로그래머는 객체의 생성과 소멸 (메모리에 확보 및 시스템에 반환) 을 신경 쓸 필요가 없음



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### ① 정수 자료형(int)

소수점이 없는 숫자(양수, 0, 음수)

```
a = 0
print(type(a))
print(a)
```

```
<class 'int'>
0
```

```
b = -11
print(type(b))
print(b)
```

```
<class 'int'>
-11
```

# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### ① 정수 자료형(int)



기본으로 10진수

- 접두어를 활용해 2, 8, 16진수 등으로 표현 가능

```
a = 12345 #10진수
print(type(a))
print(a)
```

```
<class 'int'>
12345
```

```
b = 0b11 # 2진수
print(type(b))
print(b)
```

```
<class 'int'>
3
```

```
c = 0o12 # 8진수
print(type(c))
print(c)
```

```
<class 'int'>
10
```

```
d = 0x23 # 16진수
print(type(d))
print(d)
```

```
<class 'int'>
35
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### ① 정수 자료형(int)



내장 함수 `int()`를 활용해 정수 자료형으로 변경 가능

```
a = "123"
print(type(a))
print(a)
```

```
<class 'str'>
123
```

```
b = int(a)
print(type(b))
print(b)
```

```
<class 'int'>
123
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### ① 정수 자료형(int)



범위의 제한이 없음

- 파이썬 버전 3부터 Long형 또한 정수형으로 통합

```
a = 9223372036854775808045146362345
print(type(a))
print(a)
```

```
<class 'int'>
```

```
9223372036854775808045146362345
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### 2 실수 자료형(float)

소수점이 있는 숫자

```
a = float("0.12")  
print(type(a))  
print(a)
```

```
<class 'float'>  
0.12
```

# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### 2 실수 자료형(float)



지수 표현 가능(e)

```
b = 2e-4  
print(type(b))  
print(b)
```

```
<class 'float'>  
0.0002
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### 2 실수 자료형(float)



내장 함수 **float()**를 활용해 실수 자료형으로 변경 가능

```
c = 3e3  
print(type(c))  
print(c)
```

```
<class 'float'>  
3000.0
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### 3 복소수 자료형(complex)

실수와 허수로 구성된 숫자

```
a = 10 + 2j  
print(type(a))  
print(a)
```

```
<class 'complex'>  
(10+2j)
```

# 파이썬의 수치형 자료형, 문자열 자료형

## 1 파이썬의 수치형 자료형

### 3 복소수 자료형(complex)



실수부 + 허수부j

```
b = 5 - 4j  
print(type(b))  
print(b)
```

```
<class 'complex'>  
(5-4j)
```

# 파이썬의 수치형 자료형, 문자열 자료형



## 2 파이썬의 문자열 자료형

문자, 단어 등으로 구성된 문자들의 집합

```
a = '1'
print(type(a))
print(a)
```

```
<class 'str'>
1
```

# 파이썬의 수치형 자료형, 문자열 자료형



## 2 파이썬의 문자열 자료형



큰 따옴표(“)와 작은 따옴표(‘) 모두 사용 가능

```
b = "Hello, World !"  
print(type(b))  
print(b)
```

```
<class 'str'>  
Hello, World !
```



# 파이썬의 수치형 자료형, 문자열 자료형

## 2 파이썬의 문자열 자료형



내장 함수 `str()`을 활용해 문자열 자료형으로 변경 가능

```
c = 12345
c = str(c)
print(type(c))
print(c)
```

```
<class 'str'>
12345
```

# 파이썬의 수치형 자료형, 문자열 자료형



## 2 파이썬의 문자열 자료형

### ① 문자열 안에 따옴표를 넣는 방법

**1**

이스케이프 문자 사용(w, \)

```
a = "안녕하세요"  
print(a)
```

안녕하세요



```
d = "\ '안녕하세요\' "  
print(d)
```

'안녕하세요'

# 파이썬의 수치형 자료형, 문자열 자료형

## 2 파이썬의 문자열 자료형

### 1 문자열 안에 따옴표를 넣는 방법

**2**

따옴표를 다르게 사용

```
a = "안녕하세요"  
print(a)
```

안녕하세요



```
b = "'안녕하세요'"  
print(b)
```

'안녕하세요'

```
c = '"안녕하세요"'  
print(c)
```

"안녕하세요"

# 파이썬의 수치형 자료형, 문자열 자료형

## 2 파이썬의 문자열 자료형

### 2 여러 줄의 문장을 표현하는 방법

1

이스케이프 문자 사용(w, \)

안녕하세요  
제 이름은       입니다.  
잘 부탁드립니다.



```
a = "안녕하세요 \n제 이름은       입니다. \n잘 부탁드립니다."
print(a)
```

안녕하세요  
제 이름은       !입니다.  
잘 부탁드립니다.



# 파이썬의 수치형 자료형, 문자열 자료형

## 2 파이썬의 문자열 자료형

### 2 여러 줄의 문장을 표현하는 방법

2

따옴표를 다르게 사용

안녕하세요  
제 이름은           입니다.  
잘 부탁드립니다.



```
a = """안녕하세요
제 이름은           입니다.
잘 부탁드립니다."""
print(a)
```

안녕하세요  
제 이름은           입니다.  
잘 부탁드립니다.

# 파이썬의 연산자

## 1 숫자 연산자

1

사칙 연산자(덧셈, 뺄셈, 곱셈, 나눗셈)

5 + 3

8

5 - 3

2

5 / 3

1.666666

5 \* 3

15

2

숫자를 나누고, 소수점 이하의 자릿수를 버리는  
나누기 연산자(//)

5 // 3

1

# 파이썬의 연산자

## 1 숫자 연산자

### 3 나머지 연산자(%)

 $5 \% 3$ 

2

`divmod(5, 3)`

(1, 2)

내장 함수를 활용해  
연산자와  
같은 결과 출력 가능

### 4 제곱 연산자(\*\*)

 $5 ** 3$ 

125

`pow(5, 3)`

125

# 파이썬의 연산자

## 1 숫자 연산자

### 5 복합 대입 연산자(+=, -=, \*=, /=, %=, \*\*=)

```
a = 5  
a += 3  
print(a)
```

8

```
a = 5  
a **= 3  
print(a)
```

125





# 파이썬의 연산자

## 1 숫자 연산자

### 6 math 모듈 사용

```
import math
```

```
#상용로그
print(math.log10(10))
```

1.0

```
#제곱근
print(math.sqrt(8))
```

2.8284271247461903

Tab 키를 눌러  
활용할 수 있는  
함수 목록  
확인 가능

```
math.acos
math.acosh
math.asin
math.asinh
math.atan
math.atan2
math.atanh
math.ceil
math.copysign
math.cos
math.|
```

# 파이썬의 연산자

## 2 문자열 연산자

### 1 연결 연산자(+)

```
'1' + '2'
```

```
'12'
```



```
1 + 2
```

```
3
```

```
a = 'Hello! '
```

```
b = "World"
```

```
print(a+b)
```

```
Hello! World
```

# 파이썬의 연산자

## 2 문자열 연산자

2

반복 연산자(\*)

'2' \* 3

'222'



2 \* 3

6

'안녕하세요' \* 3

'안녕하세요안녕하세요안녕하세요'

# 파이썬의 연산자

## 2 문자열 연산자

### 3 선택 연산자(인덱싱)

문자	안	녕	하	세	요
인덱스	0	1	2	3	4

```
a = '안녕하세요'
print(a[0])
```

안

```
a = '안녕하세요'
print(a[-1])
```

요

문자열은  
시퀀스 자료형으로  
인덱스가 있고,  
인덱스로  
값의 접근이 가능

# 파이썬의 연산자

## 2 문자열 연산자

### 4 범위 선택 연산자(슬라이싱)

문자	안	녕	하	세	요
인덱스	0	1	2	3	4

```
a = '안녕하세요'
print(a[1:3])
```

녕하

```
a = '안녕하세요'
print(a[0:5:2])
```

안하요

변수[시작(이상):끝(미만):스텝]

## Run! 프로그래밍



## Mission 1

## 여러 줄의 문장을 표현하기

```
print("안녕하세요 \n학습자님 \n이번 회차도  
수고하셨습니다. \n")
```

또는

```
print("""안녕하세요  
학습자님  
이번 회차도 수고하셨습니다.  
""")
```

## Mission 2

## 반복 연산자(\*) 활용하기

```
a = '104'  
print(3*a + a + a)
```

정답

104104104104104