



# 기초 프로그래밍 및 데이터구조

---

Day 02. 파이썬 자료형

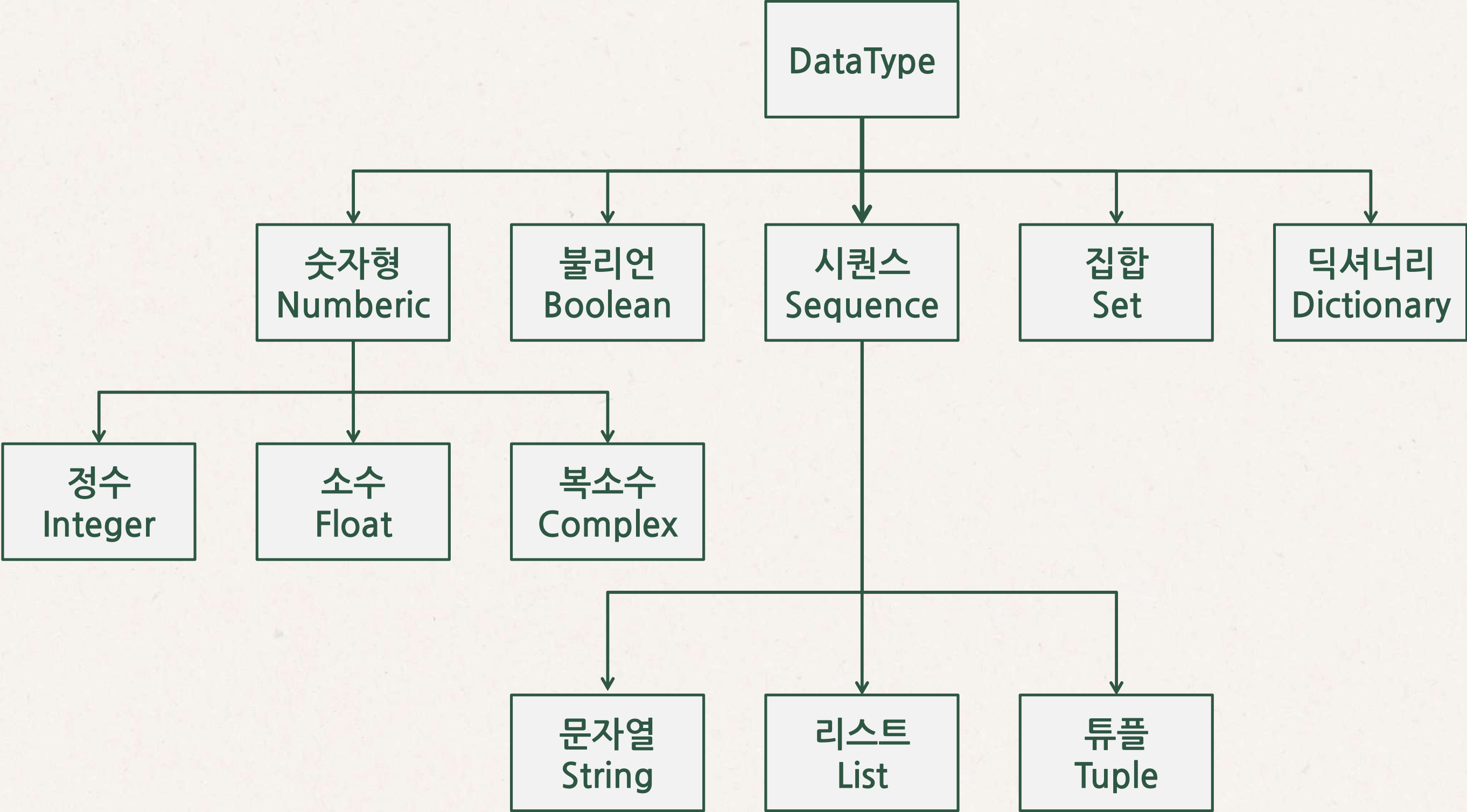
● 강의자: 김수빈

# Contents

---

- 1 python 자료형
- 2 정수형, 실수형
- 3 문자열
- 4 포매팅
- 5 Boolean

# 1 Python 자료형



# 1 Python 자료형

숫자형

자료형	예시
정수형 (int)	10
소수형 (float)	3.14
복소수형 (complex)	1 + 2j
불리언 (bool)	True
집합	{1,2,3}
딕셔너리	{'key1' : 'value1' , 'key2' : 'value2; }



# 1 Python 자료형

---

시퀀스

자료형	예시
문자열 (str)	'Hello, World!'
리스트 (list)	[1, 2, 3]
튜플 (tuple)	(1, 2, 3)

- 시퀀스 자료형 : 각각의 요소들이 연속적으로 이어진 자료형

# 1 Python 자료형

---

## type()

- Python의 내장함수
- 해당 값의 **데이터 타입**을 반환하는 역할
- 데이터 유형을 확인할 때 유용
- 객체의 다양한 타입을 판별할 수 있다.

```
a = 10  
type(a)
```

[1] ✓ 0.0s

Python

... int

```
b = 3.14  
type(b)
```

[2] ✓ 0.0s

Python

... float

## 2 정수형, 실수형

---

### 정수형 (Integer)

- 소수점이 없는 숫자, 즉 정수를 표현하는 자료형
- 자료형 `int`로 표기
- 크기에 제한이 없다
- 양수와 음수를 모두 표현할 수 있다
- Ex) 10, -5, 1000000

### 실수형 (Float)

- 소수점을 포함한 숫자를 표현하는 자료형
- 자료형 `float`으로 표기
- 소수점을 사용하여 정밀한 계산이 가능하다
- 양수와 음수를 모두 표현할 수 있다.
- Ex) 10.5, 31.4, -30.1



5.0은 정수? 실수?

## 2 정수형, 실수형

---

### 기본 산술 연산

- 덧셈 (+)
- 뺄셈 (-)
- 곱셈 (\*)
- 나눗셈 (/)

### 나머지, 몫

- 나머지 (%)
- 몫 (//)

### 거듭제곱

- 거듭제곱 (\*\*)



# 3 문자열

---

## 문자열 (String)

- 문자, 단어, 문장의 집합
- **작은따옴표( ' ' ), 큰따옴표 ( “ ” )**로 감싸서 표현한다.
- 문장 안에 작은따옴표나 큰따옴표가 포함되어야 하는 경우 서로 다른 따옴표로 묶어준다.
- 만약 작은따옴표, 큰따옴표가 모두 들어가는 긴 문장이라면? **따옴표를 3번 사용하여 묶어준다.**
- 문자 하나부터 여러 문자 조합까지 다양한 길이의 문자열을 표현할 수 있다.
- 자료형 **srt**
- Ex) 'Hi', “안녕”, ‘100’, “!!!”

# 3 문자열

---

연결

반복

인덱싱

슬라이싱

인덱싱 : 문자열의 특정 위치에 있는 문자 참조

슬라이싱 : 부분 문자열 추출

# 3 문자열

## 이스케이프 코드 ( Escape Codes)

- 문자열 내에서 특수한 의미를 갖는 문자들을 표현하기 위해 사용되는 코드
- 줄바꿈, 탭, 따옴표 등 특수 문자를 문자열 안에서 표현할 때 사용

이스케이프 코드	설명
\\	백슬래시
\'	작은따옴표
\"	큰따옴표
\n	줄바꿈
\t	탭

# 3 문자열

---

## len()

- Python의 내장함수
- 주어진 객체의 길이를 반환
- 문자열 외에도 리스트, 튜플, 딕셔너리 등 다양한 데이터 타입에서 사용 가능하다.
- 객체는 반드시 가변적인 크기를 갖는 시퀀스 타입이어야 한다.

```
a = 'Hello, World!'
len(a)
```

[1] ✓ 0.0s Python

... 13

```
text = 'Text...'
```

```
len(text)
```

[2] ✓ 0.0s Python

... 7



# 3 문자열

---

## upper( ), lower( )

- 문자열을 모두 대/소문자로 변환합니다.

## strip( )

- 문자열 양쪽에서 특정 문자를 제거합니다.

## replace( )

- 특정 문자를 다른 문자로 바꿉니다.

## split( )

- 문자열을 특정 구분자를 기준으로 나눕니다.

# 4 포매팅

---

## 문자열 포매팅 ( String Formatting)

- 문자열 안에 변수나 표현식을 삽입하여 원하는 형식으로 출력하는 방법
- 다양한 형식의 데이터를 하나의 문자열로 결합하거나 출력할 때 사용된다.
- ‘%’ 연산자 포매팅
- str.format( ) 매서드
- f-strings

# 4 포매팅

## % 연산자 포매팅

% 연산자를 사용해 문자열 안에 값을 삽입하는 방식

```
name='수빈'  
age=6  
print("저는 %s 이고 %d 살이에요." % (name, age))
```

[5] ✓ 0.0s

Python

... 저는 수빈 이고 6 살이에요.





# 4 포매팅

## str.format( )

str.format( ) 매서드를 사용하는 보다 유연한 포매팅 방식

```
name = "수빈"  
age = 6  
print("저는 {} 이고 {} 살이에요.".format(name, age))
```

[6] ✓ 0.0s

Python

... 저는 수빈 이고 6 살이에요.





# 4 포매팅

## f-strings

최신 포매팅 방법으로 가장 빠르고 간결하다.

```
name = "수빈"  
age = 6  
print(f"저는 {name} 이고 {age} 살이에요.")
```

[7] ✓ 0.0s

Python

... 저는 수빈 이고 6 살이에요.



# 5 Boolean

---

## 불리언 (Boolean)

- 참(True)와 거짓(False)을 표현하는 자료형
- 조건문, 반복문, 비교연산 등에서 논리적인 판단을 위해 사용한다.
- True : 참을 나타내는 값, 1로 평가
- False : 거짓을 나타내는 값, 0으로 평가

# 5 Boolean

## 비교 연산자

- 결과는 항상 Boolean값

비교 연산자	설명
==	두 값이 같은지 비교
!=	두 값이 다른지 비교
>	왼쪽 값이 큰지 비교
<	오른쪽 값이 큰지 비교
>=	왼쪽 값이 크거나 같은지 비교
<=	오른쪽 값이 크거나 같은지 비교

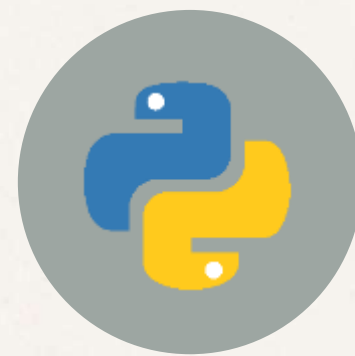
# 5 Boolean

## 논리 연산자

- 결과는 항상 Boolean값

논리 연산자	설명
and	두 조건이 모두 참일 때 True
or	두 조건 중 하나라도 참이면 True
not	참을 거짓으로, 거짓을 참으로 변환





감사합니다.

Thank You

---