



## 02장 파이썬 프로그램 래밍의 기초, 자료형

자료형을 알고 있다면 그 언어의 절반을 터득한  
것



# Agenda

01

숫자형 데이터 타입 / 다양한 연산

02

문자열 자료형 1/2

03

문자열 자료형 2/2

04

사용자 입력과 출력

숫자형 데이터 타입  
/ 다양한 연산



# 숫자형 데이터 타입



항목	설명	예시
정수형 (Integer)	정수를 표현하는 데이터 타입으로, 양수와 음수를 포함합니다.	$x = 10, y = -5$
실수형 (Float)	소수점을 포함하는 숫자를 표현합니다.	$x = 3.14, y = -2.71$
컴퓨터식 지수 표현 방식	숫자를 지수로 표현하는 방식으로, e 또는 E를 사용하여 10의 거듭제곱을 나타냅니다.	$x = 1.2e3$ (1200), $y = 2.5E-4$
8진수 (Octal)	숫자를 8진수로 표현하며, 숫자 앞에 0o 또는 0O를 붙여 사용합니다.	$x = 0o12$ (10진수 10)
16진수 (Hexadecimal)	숫자를 16진수로 표현하며, 숫자 앞에 0x 또는 0X를 붙여 사용합니다.	$x = 0x1A$ (10진수 26)

# 다양한 연산



항목	설명	예시
사칙연산 (Arithmetic)	덧셈, 뺄셈, 곱셈, 나눗셈을 수행합니다.	$x + y$ , $x - y$ , $x * y$ , $x / y$
제곱 (Exponentiation)	숫자의 거듭제곱을 계산합니다.	$x ** y$ ( $x$ 의 $y$ 승)
나머지 연산 (%)	두 숫자를 나눈 후 나머지를 반환합니다.	$x \% y$
몫 연산 (//)	두 숫자를 나눈 후 몫만 반환합니다.	$x // y$
정수 나눗셈 (/)	두 숫자를 나눠 실수 결과를 반환합니다.	$x / y$



# 형성평가 1/3



## 문제 1

정수형(int)에 대한 설명으로 옳은 것을 고르세요.

1. 소수점을 포함한 숫자를 표현한다.
2. 10진수, 8진수, 16진수 모두를 포함한다.
3. e 또는 E를 사용하여 지수 형태로 표현한다.
4. 문자열로만 저장할 수 있다.

## 문제 2

실수형(float)에 대한 설명으로 틀린 것을 고르세요.

1. 소수점을 포함하여 숫자를 표현한다.
2. e 또는 E를 사용해 지수 표현 방식을 사용할 수 있다.
3. 항상 8진수로만 저장된다.
4. 음수를 표현할 수 있다.

# 형성평가 2/3



## 문제 3

컴퓨터식 지수 표현 방식에 대한 설명으로 옳은 것은?

1. 숫자를 16진수로 표현하는 방법이다.
2. 0o를 사용하여 숫자를 표현한다.
3. e나 E 뒤에 10의 거듭제곱 값을 붙인다.
4. 소수점 없이 정수형으로만 사용할 수 있다.

## 문제 4

8진수에 대한 설명으로 옳은 것을 고르세요.

1. 숫자 앞에 0x를 붙인다.
2. 숫자 앞에 0o를 붙인다.
3. 8 이상의 숫자는 표현할 수 없다.
4. 소수점이 포함된다.

# 형성평가 3/3



## 문제 5

연산자 %의 역할은?

1. 두 수를 나눈 몫을 반환한다.
2. 두 수를 나눈 후 나머지를 반환한다.
3. 두 수를 거듭제곱한 값을 반환한다.
4. 두 수를 나눈 후 소수점을 포함한 결과를 반환한다.

## 문제 6

연산자 //에 대한 설명으로 옳은 것을 고르세요.

1. 두 숫자를 나눈 후 정수 몫만 반환
2. 두 숫자를 나누어 실수 결과를 반환
3. 두 숫자를 곱한 값을 반환
4. 두 숫자를 거듭제곱한 값을 반환



문자열 자료형 1/2



# 문자열 자료형



항목	설명	예시
문자열 자료형 만드는 4가지 방법	작은따옴표(), 큰따옴표(), 삼중 작은따옴표(''), 삼중 큰따옴표(''')로 문자열 생성 가능.	'hello', "hello", '''hello''', '''hello'''
문자열에 따옴표 포함시키기	따옴표 안에 따옴표를 포함하려면 다른 종류의 따옴표를 사용하거나, 같은 따옴표일 경우 이스케이프 문자(\) 사용.	"He said 'hello'", 'He said "hello"', "He said \"hello\""
여러 줄로 이루어진 문자열	삼중 따옴표('' 또는 ''')로 여러 줄 문자열 생성 가능.	'''Line1\nLine2''' 또는 '''Line1\nLine2'''
문자열 더해서 연결하기	+ 연산자를 사용하여 문자열을 이어붙임.	"Hello, " + "world!"
문자열 곱하기	* 연산자를 사용하여 문자열을 반복.	"Python! " * 3

# 문자열 자료형



항목	설명	예시
인덱싱 (Indexing)	문자열의 특정 위치에 접근. 0부터 시작하며, 음수는 뒤에서부터 인덱싱.	<code>s = "Python"; s[0] → 'P', s[-1] → 'n'</code>
슬라이싱 (Slicing)	문자열의 일부분을 추출. [시작:끝:단계] 형식으로 사용, 끝 인덱스는 포함되지 않음.	<code>s = "Python"; s[0:3] → 'Pyt', s[:2] → 'Pt'</code>
문자열 포매팅	% 연산자, <code>str.format()</code> 메서드, f-string을 사용하여 문자열 내 변수 삽입 가능.	<code>"Hello, %s" % "world", "Hello, {}".format("world"), f"Hello, {x}"</code>
정렬과 공백	<code>ljust()</code> , <code>rjust()</code> , <code>center()</code> 메서드를 사용하여 문자열 정렬과 공백 추가.	<code>'Hello'.center(10) → ' Hello '</code>
소수점 표현	소수점 자릿수는 포매팅을 사용하여 조정 가능.	<code>"{: .2f}".format(3.14159) → '3.14', f"{3.14159: .2f}" → '3.14'</code>

# 형성평가 1/4



## 문제 1

문자열을 만드는 방법에 해당하지 않는 것은?

1. 작은따옴표('')
2. 큰따옴표("")
3. 백틱(`)
4. 삼중 따옴표(''', ''')

## 문제 2

문자열 안에 따옴표를 포함시키는 방법으로 옳은 것은?

1. 같은 종류의 따옴표만 사용한다.
2. 다른 종류의 따옴표를 사용하거나 이스케이프 문자(\)를 사용한다.
3. 문자열 안에 따옴표를 포함할 수 없다.
4. 따옴표 대신 특수 문자를 사용해야 한다.

# 형성평가 2/4



## 문제 3

여러 줄로 이루어진 문자열을 만드는 방법은?

1. 큰따옴표("")만 사용한다.
2. 작은따옴표('')만 사용한다.
3. 삼중 따옴표('"' 또는 ""')를 사용한다.
4. 문자열을 여러 번 연결해야 한다.

## 문제 4

문자열 연결(Concatenation)을 할 때 사용하는 연산자는?

1. \*
2. +
3. =
4. %

# 형성평가 3/4



## 문제 5

문자열을 반복하는 연산에 사용되는 기호는?

1. %
2. +
3. \*
- 4.

## 문제 6

문자열 인덱싱에서 음수 인덱스의 의미는?

1. 문자열의 맨 앞부터 숫자를 센다.
2. 문자열의 길이와 상관없이 작동하지 않는다.
3. 문자열의 맨 뒤부터 숫자를 센다.
4. 문자열의 중간부터 센다.

# 형성평가 4/4



## 문제 7

문자열 슬라이싱에서 `s[1:4]`의 결과는?  
(단, `s = "Python"`)

1. "Pyt"
2. "yth"
3. "ytho"
4. "yth"

## 문제 8

문자열을 가운데 정렬하고 공백을 추가하려면 어떤 메서드를 사용해야 하나요?

1. `ljust()`
2. `rjust()`
3. `center()`
4. `align()`



문자열 자료형 2/2



# 문자열 자료형



항목	설명	예시
<code>count</code>	문자열 내 특정 문자의 등장 횟수를 반환합니다.	<code>"hello".count('l') → 2</code>
<code>find</code>	특정 문자열이 처음 나타나는 위치를 반환하며, 없으면 -1을 반환합니다.	<code>"hello".find('e') → 1</code>
<code>index</code>	특정 문자열이 처음 나타나는 위치를 반환하며, 없으면 오류를 발생시킵니다.	<code>"hello".index('l') → 2</code>
<code>join</code>	문자열의 리스트를 특정 구분자로 연결하여 하나의 문자열로 만듭니다.	<code>",".join(['a', 'b', 'c']) → "a,b,c"</code>
<code>upper</code>	문자열의 모든 문자를 대문자로 변환합니다.	<code>"hello".upper() → "HELLO"</code>

# 문자열 자료형



항목	설명	예시
<b>lower</b>	문자열의 모든 문자를 소문자로 변환합니다.	"HELLO".lower() → "hello"
<b>strip</b>	문자열의 양쪽 끝에서 공백이나 특정 문자를 제거합니다.	" hello ".strip() → "hello"
<b>replace</b>	문자열 내 특정 문자열을 다른 문자열로 바꿉니다.	"hello world".replace('world', 'Python') → "hello Python"
<b>split</b>	문자열을 특정 구분자를 기준으로 분리하여 리스트로 반환합니다.	"a,b,c".split(',') → ['a', 'b', 'c']

# 형성평가 1/3



## 문제 1

문자열에서 특정 문자의 등장 횟수를 반환하는 메서드는 무엇인가요?

1. find
2. count
3. index
4. split

## 문제 2

문자열에서 특정 문자가 처음 나타나는 위치를 반환하며, 찾지 못하면 -1을 반환하는 메서드는?

1. index
2. find
3. replace
4. upper

# 형성평가 2/3



## 문제 3

문자열에서 특정 문자가 처음 나타나는 위치를 반환하며, 찾지 못하면 오류를 발생시키는 메서드는?

1. find
2. index
3. count
4. strip

## 문제 4

문자열의 모든 문자를 대문자로 변환하는 메서드는?

1. lower
2. upper
3. replace
4. split

# 형성평가 3/3



## 문제 5

문자열의 모든 문자를 소문자로 변환하는 메서드는?

1. upper
2. lower
3. join
4. strip

## 문제 6

문자열 내 특정 문자를 다른 문자열로 바꾸는 메서드는?

1. replace
2. count
3. join
4. find

# 사용자 입력과 출력





# 사용자 입력과 출력



항목	설명	예시
<b>input의 사용</b>	사용자가 입력한 값을 문자열로 받아옵니다.	<code>name = input("Enter your name: ")</code> → 사용자 입력을 받아 <code>name</code> 변수에 저장
<b>print 문</b>	화면에 값을 출력합니다. 쉼표(,)로 여러 값을 연결해 출력할 수 있습니다.	<code>print("Hello", "World!")</code> → Hello World!
<b>sep으로 출력하기</b>	<code>print()</code> 함수에서 출력값을 구분하는 문자열을 지정합니다. 기본 값은 공백입니다.	<code>print("apple", "banana", sep="-")</code> → apple-banana

# 형성평가 1/2



## 문제 1

사용자가 입력한 값을 문자열로 받아 오는 함수는 무엇인가요?

1. print
2. input
3. sep
4. join

## 문제 2

Python에서 값을 화면에 출력하는 함수는 무엇인가요?

1. print
2. output
3. show
4. display

# 형성평가 2/2



## 문제 3

`print("apple", "banana", sep="-")`의 출력 결과는 무엇인가요?

1. apple banana
2. apple-banana
3. apple,banana
4. applebanana

## 문제 4

`print("Hello", "Python", sep="*")`의 출력 결과는 무엇인가요?

1. Hello\*Python
2. Hello Python
3. Hello, Python
4. HelloPython