

<한눈에 끝내는 파이썬 3>

1. 시작하기에 앞서

* 프로그래밍이란?

특정 목적을 위해 컴퓨터에 내리는 명령의 집합.

* 프로그램을 만드는 행위를 프로그래밍, 개발이라고 한다.
* 코드 : 컴퓨터 언어로 작성하는 논리적인 글
* 코딩 : 코드를 작성하는 과정
* 파이썬

1. 엡 프로그래밍 : Django(장고) 라는 오픈 소스 기반으로 웹 프레임워크를 사용한다.

오픈소스 : 소스코드를 누구나 참고할 수 있도록 공부한 것.

프레임워크 : 기본형태를 제공함으로써 개발자의 반복적인 작업을 줄여주는 뼈대 코드

1. 빅데이터 분석: 대용량 데이터를 분석하고 데이터 간의 패턴을 알려주며 이를 시각화하여 보여주는 분야. 기계학습(인공지능) 개발에 많이 사용한다.

* 인터프리터 언어 : 해석기

소스코드를 바로 실행하는 컴퓨터 프로그램 혹은 그러한 환경.

* 인덴트(띄어쓰기, 공백)

파이썬은 다른 프로그래밍 언어보다 인덴트가 민감한 언어이다.

\* indent 에러가 발생했다면 띄어쓰기와 공백을 확인해 본다.

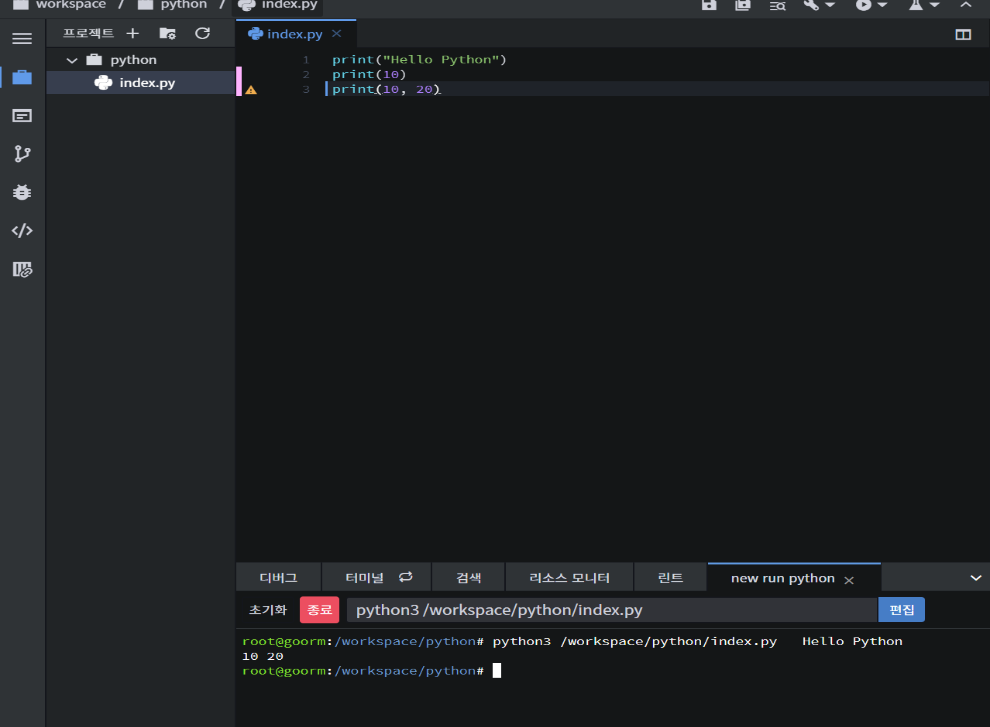
* 반복문 : 일을 되풀이 할 때 사용한다.
  1. 순서도 : 여러 종류의 상자와 이를 이는 화살표를 이용하여 명령을 어떤 순서대로 어떻게 처리해야 하는지 보여주는 도표를 의미.
* 타원형 상자 : 순서도의 시작과 끝/ 육각형 상자 : 작업하기 전 초기설정/

마름모 상자 : 조건에 따라 참/거짓 구분 / 사각형 상자 : 처리해야 할 작업

* 조건문 : if조건문 사용 if옆의 괄호 속에 조건의 참,거짓을 판별한다.
* 주석 : 일종의 부가 설명 (코드에 설명) / 컴파일 과정을 거치지 않는다.
  1. 한 줄만 주석 처리 : #
  2. 여러문장을 주석처리 할 때 “””~”””

1. 콘솔 출력의 기본

* 콘솔 출력(‘콘솔’은 컴퓨터와 소통하는 창)

콘솔 출력 함수 : print(출력하고자 하는 데이터)/여러 데이터를 나열할 때는 콤마를 사용한다.

* 변수의 선언과 초기화
  1. 변수 : 변하는 수, 특정한 형식을 가진 값을 저장하는 공간

- 자료형: 변수가 가지는 특정한 형식

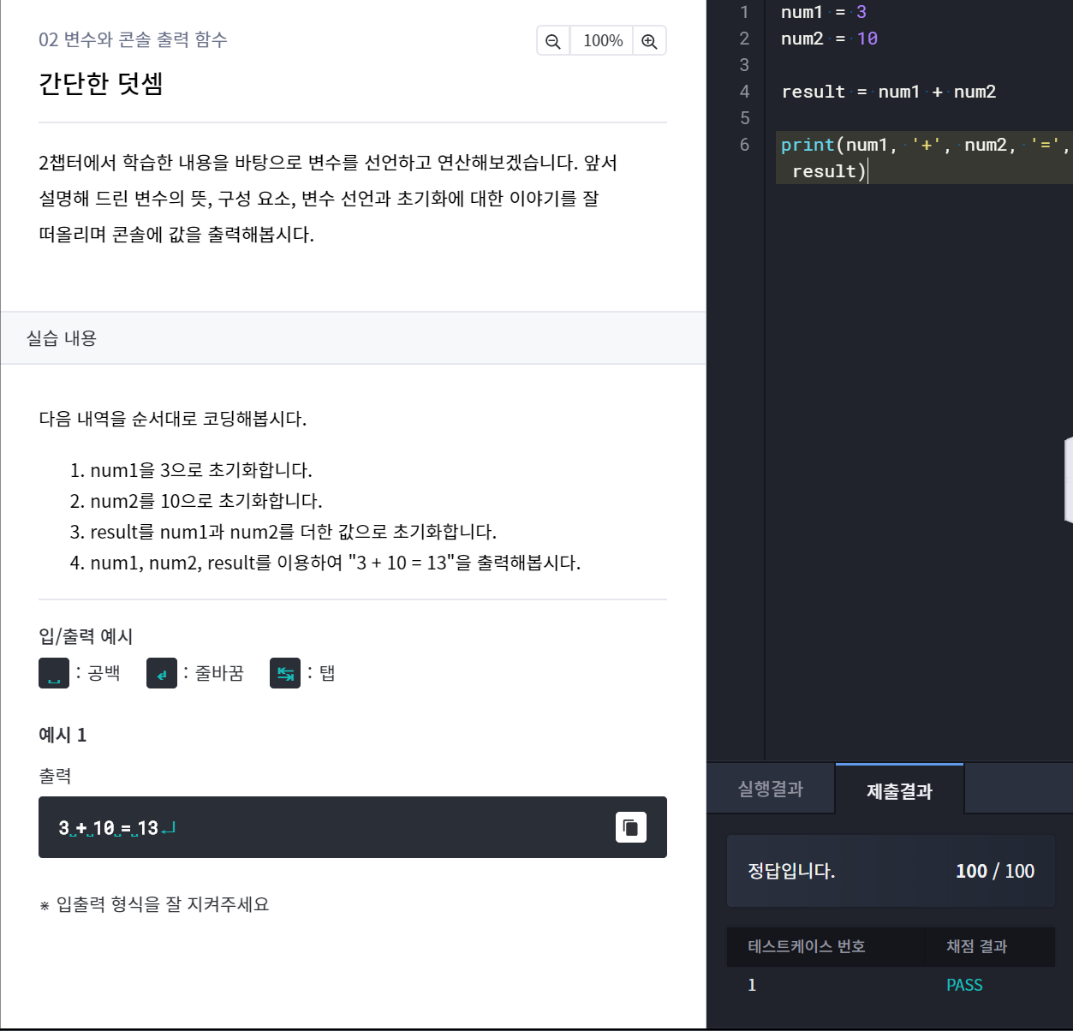
- 변수명 : 변수의 이름

- 변숫값 : 변수에 저장한 값

* 1. 변수 초기화 : 새로 생성된 변수에 처음 값을 입력하는 과정

\* 파이썬은 자료형(데이터의 타입,형식)을 자동으로 추론한다.

- (“”) 문자열을 표현

- +기호로 문자열 사이를 연결할 수 있다.

1. 기본적인 자료형과 연산자

* 콘솔 입력의 기본

콘솔 입력 함수

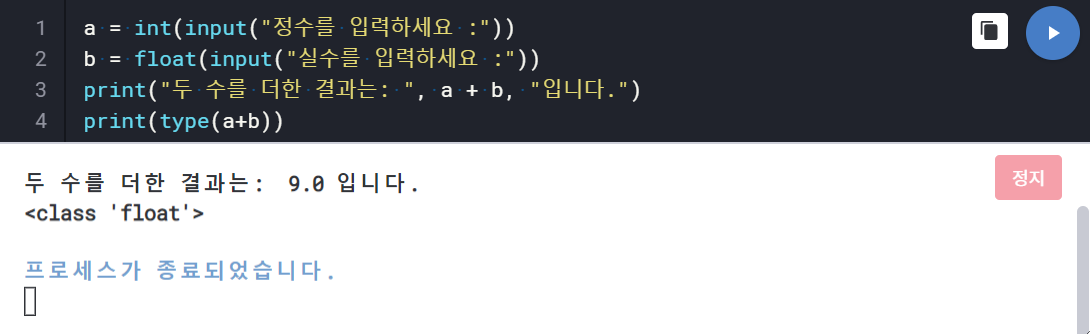
* 1. input()

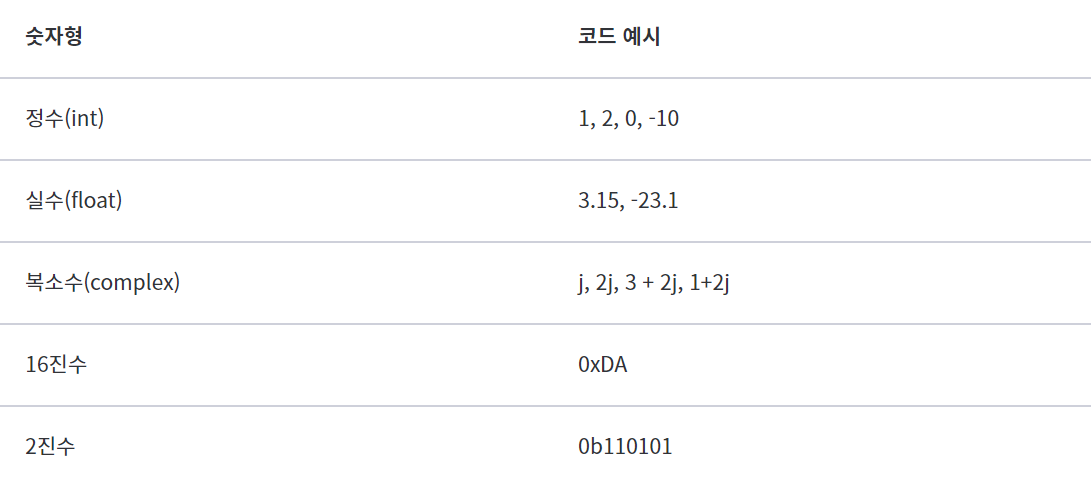
사용자가 콘솔 창에 직접 입력한 값을 변수에 할당(=저장)한다.

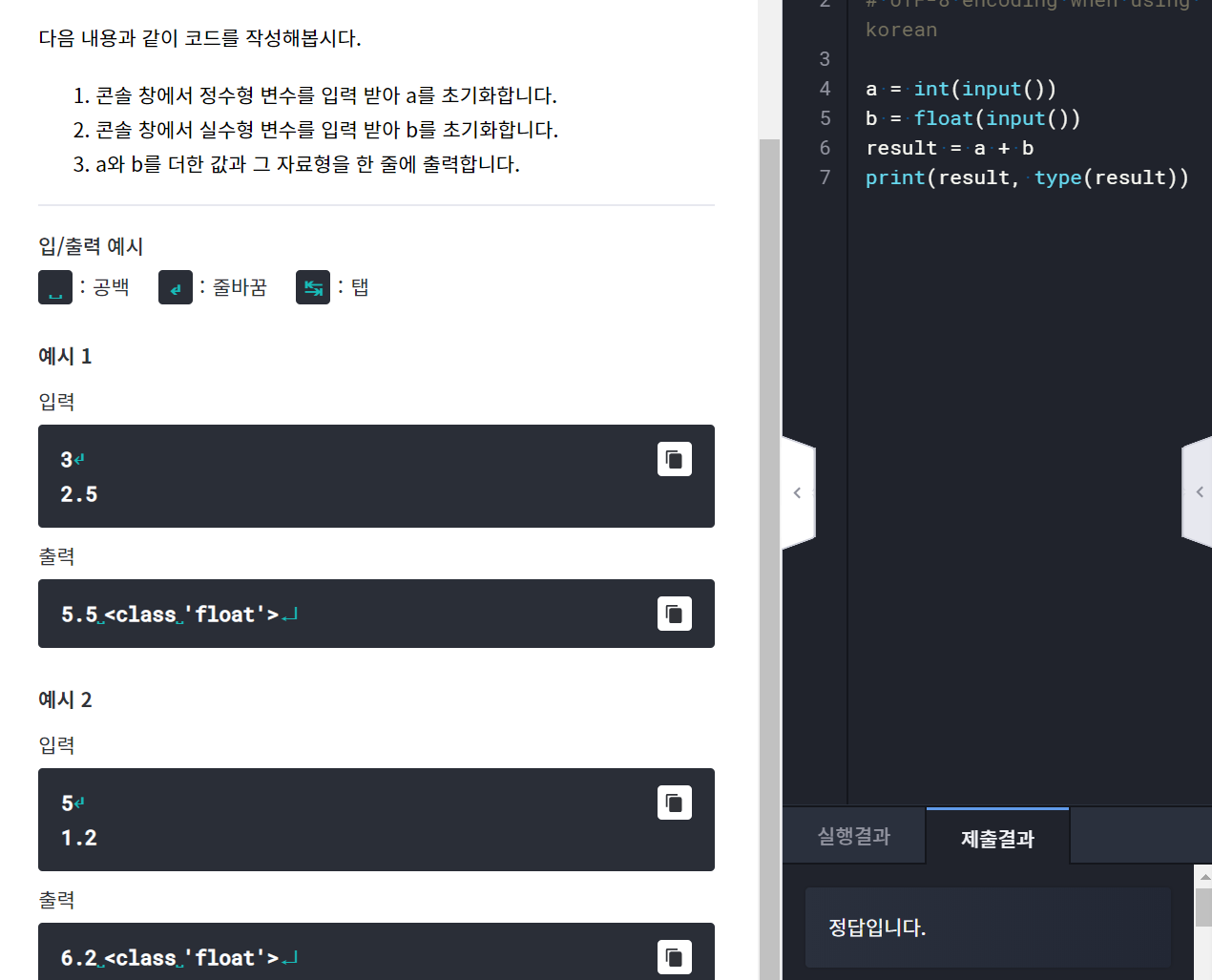
변수 = input(입력받는 값)을 코드에 작성해 넣는다.

입력한 값을 ‘문자열’로 저장한다

변수의 자료형을 확인하고 싶다면 type()함수로 알아 낼 수 있다.



* 숫자형 : 숫자 형태로 이루어진 자료형

1. 정수형 : 일반적인 정수, int형이라 저장
2. 실수형 : 소수점이 포함된 숫자, float형이라 저장
3. 복소수형 : 허수를 포함한 수 / 실수부 “변수.real”, 허수부는 “변수.imag” 켤레복소수 “변수.conjugate()”로 반환
4. 16진수는 0x1A형태로, 2진수는 1과 0으로 표현

* 불(bool)형

참과 거짓을 다룬다.

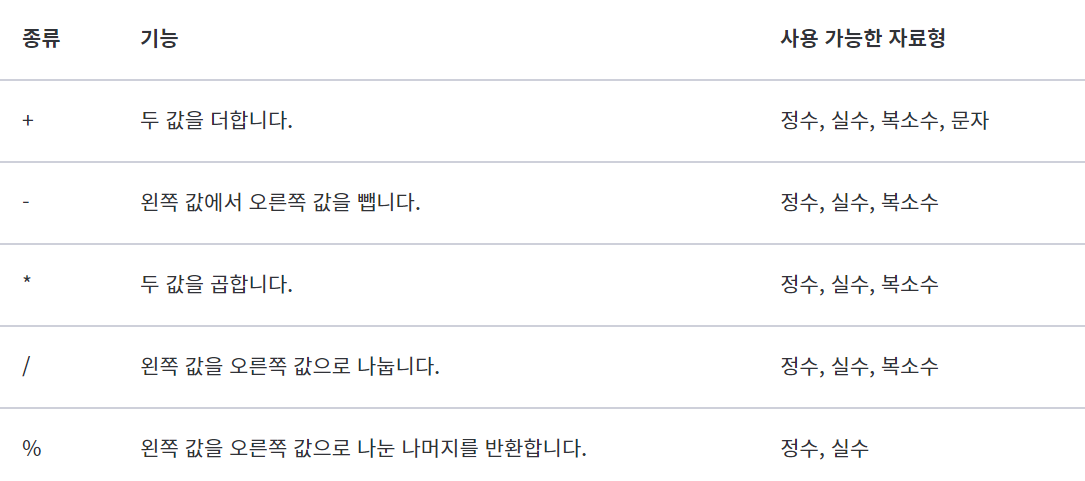
bool형을 초기화하는 형식 “변수=True/False” 앞 글자는 반드시 대문자로 입력

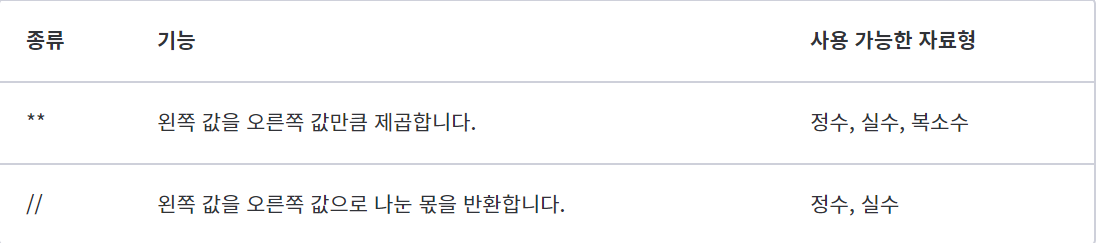
\* 파이썬에서는 값이 존재하면 참, 존재하지 않으면 거짓으로 판별한다.



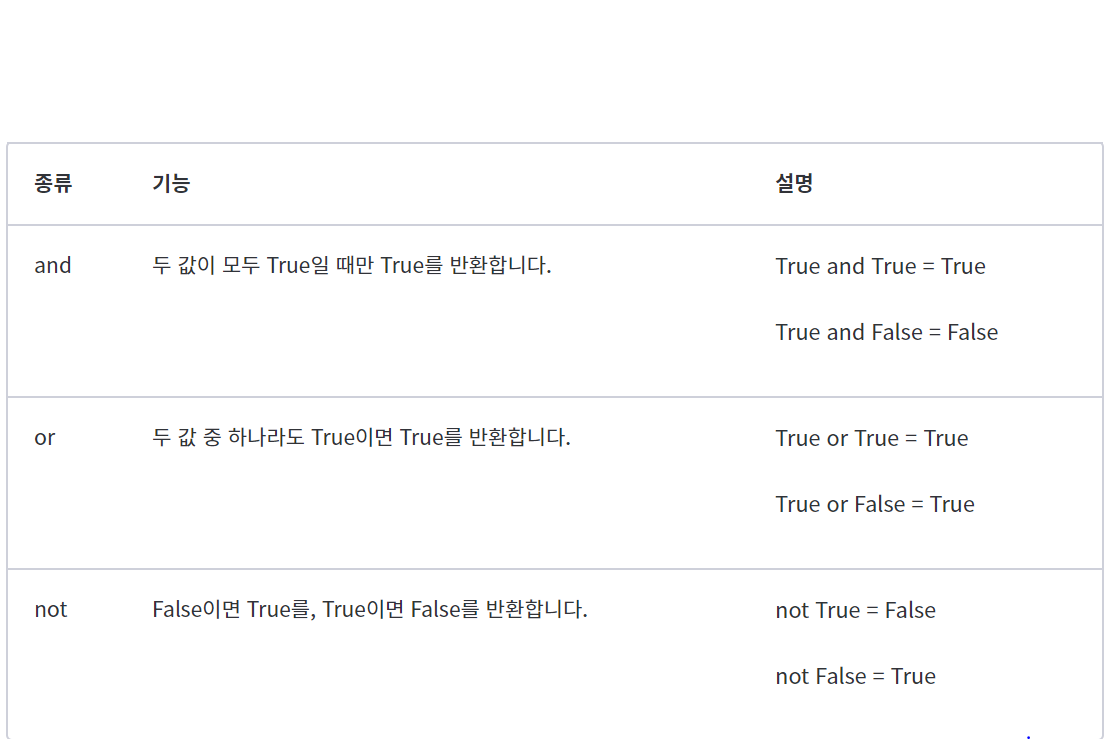
==는 양쪽의 값을 같다고 비교하는 연산자를 의미

* 수식 연산자

두개의 피연산자를 요구하는 이항 연산자

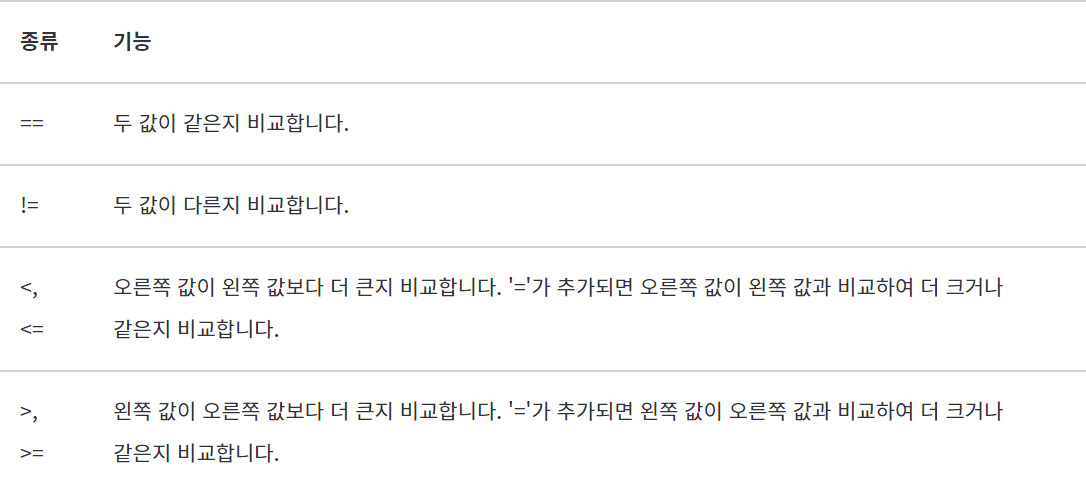


\*문자열+숫자는 불가능/ 연산 대상인 변수가 모두 숫자라면 자유롭게 연산가능

* 할당 연산자 : ‘=’는 수식연산자와 함께 사용할 수 있는데 이를 복합대입연산자라고 한다.
* 논리연산자 : 합집합, 교집합, 차집합과 동일한다.

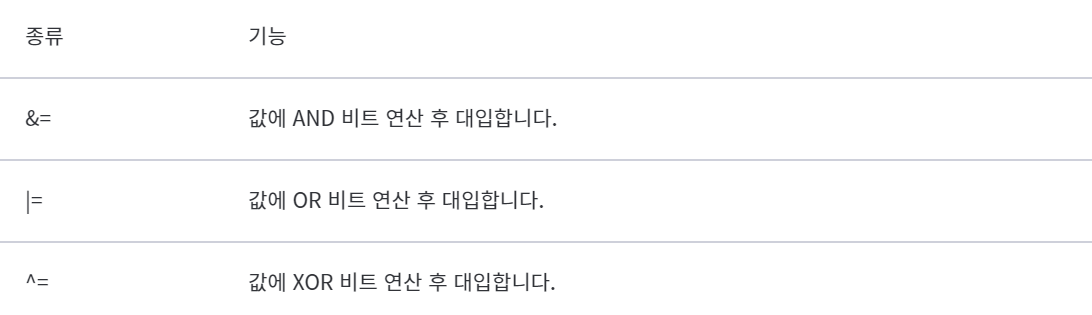
\* 논리 연산자는 추후 조건문과 함께 사용한다.

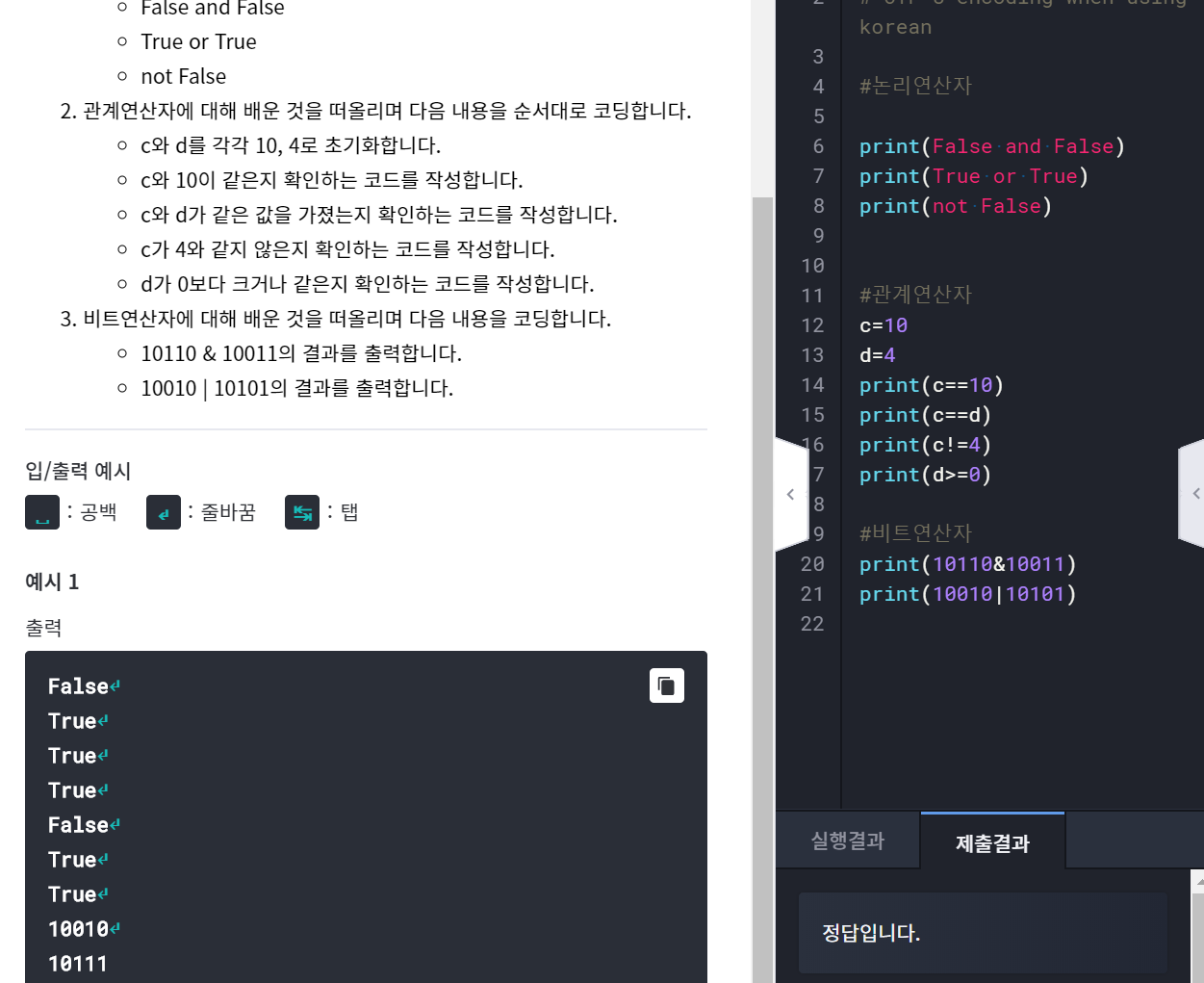
* 관계연산자 : 두 값의 대소 혹은 동등의 관계를 따지는 연산자이다.



* 비트연산자

2진수로 변경한 뒤 연산한다.





1. 문자열

* 문자열 자료형

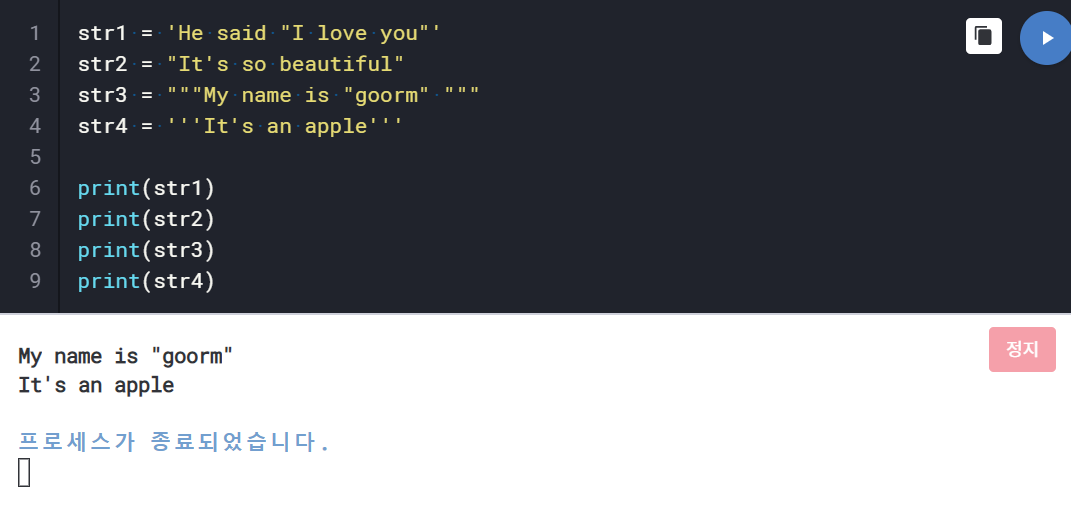
문자열 : 따옴표로 감싸져서 표현된 문자, 단어들의 묶음

\* 따옴표로 감싸진 문자열에서 문자가 특별하게 인식되는 경우

* 1. 따옴표 안에 같은 종류의 따옴표를 사용한 경우
  2. 이스케이프 시퀸스를 사용한 경우
  3. 문자열 포매팅을 한 경우

1. 따옴표 안에 같은 종류의 따옴표를 사용한 경우

* 작은따옴표로 감싸기(‘문자열’)
* 큰따옴표로 감싸기(“문자열”)
* 작은따옴표 세 개로 감싸기(‘’’문자열’’’)
* 큰 따옴표로 세 개로 감싸기 (“””문자열”””) : 문장 내 엔터를 넣기 위해서 사용할 수 있다.



1. 이스케이프 시퀸스를 사용한 경우

: 문자열에서 표현하기 어려운 특장 문자를 백슬래쉬로 표현하는 것



* 문자열 연산

Concatenation: 문자열을 덧셈 부호를 이용해 연결 한 것.

문자열 곱셈 : “문자열\*정수” 형태로 문자열을 반복한다는 의미이다.(파이썬만 해당)

* 인덱싱과 슬라이싱

Index : 수많은 데이터 중 원하는 항목을 쉽게 찾도록 기록된 표기.

* 메모리와 데이터

문자열은 아스키코드를 사용한다. / 문자열은 하나로 보일지라도 문자마다 인덱싱이 가능하다.

* 인덱싱 : 문자열에 포함된 각 문자에 대해 순차적으로 번호를 매기는 것

“변수 이름[인덱스 값]”/ 인덱스 값은 인덱스 번호를 뜻하며 인덱스 번호는 0부터 시작한다.

‘-‘를 붙임으로 뒤 요소부터 접근 가능하다.

* 공백도 ‘32’라는 10진수 아스키코드 값으로 메모리에 저장된다.

\* 문자여 자료형은 lmmutable타입으로 한 번 초기화하면 사용자 임의로 값을 바꿀 수 없다.

* 슬라이싱

단어 그대로 지정 범위만큼 데이터 요소를 잘라내는 기능

“변수명[첫 인덱스 번호:마지막 인덱스 번호]”

인덱스 번호에 ‘-‘를 붙임으로 뒤 요소부터 접근 가능

* 문자열 포매팅 1

사용자가 문자열의 포맷을 지정하는 것

<포맷 코드>

문자열 안에 %d, %s와 같은 포맷 코드를 입력한 뒤, 문자열 뒤에 %를 시작으로 변숫값을 작성한다.

\* 문자열은 숫자형 포맷 코드를 사용하여 표현 할 수 없다.

\* %.(숫자)f 를 기록하여 소수점을 몇 자리까지 표시할 것인가 지정할 수 있다.

\* 정수형은 실수형으로, 실수형을 정수형으로 포매팅 가능하다.

\* 출력 폭을 지정할 수 있다. %(폭)d 형식으로 입력하면 오른쪽 정렬되고 왼쪽에는 공백으로 채워진다.

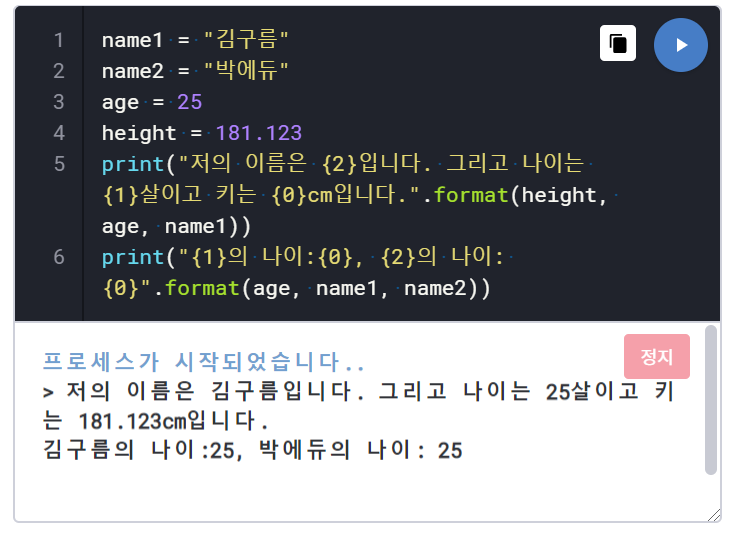
\* 왼쪽으로 정렬하고 싶으면 %-(폭)d형식으로 입력한다.

* 문자열 포매팅 2

%문자열 포매팅은 문자열 안에서 사용한 포맷코드와 뒤에 입력한 변수가 1대 1대응을 해야한다.

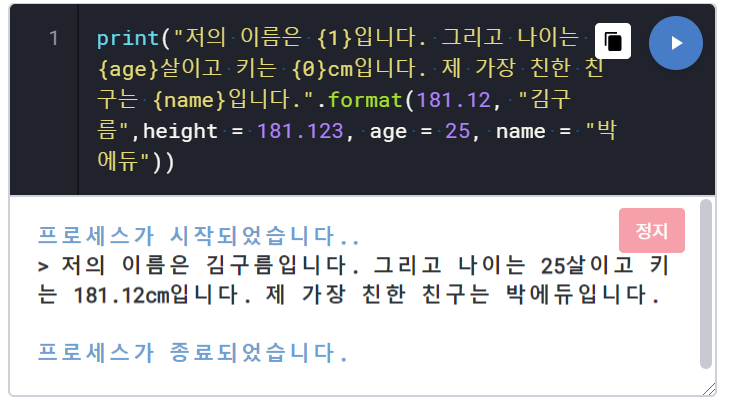
* 인덱스값으로 바로 대입

Format()함수는 %d 같은 포맷 대신에 “{인덱스값}”을 사용한다.

사용형식 : 문자열.format(요소1,요소2…)

* 변수 이름으로 대입

Format()함수 내에서 변수를 선언하고 초기화하면 그 값을 “{변수명}” 형식으로 바로 받아 올 수 있다.



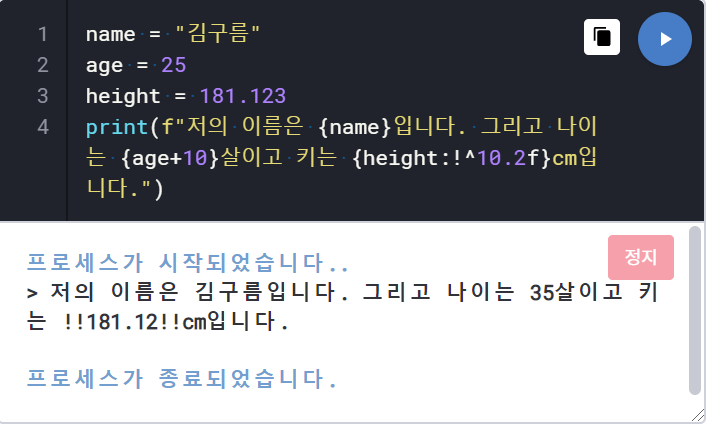
출력 폭 지정 : “{인덱스값 or 변수(출력형식)}”.format(요소)”

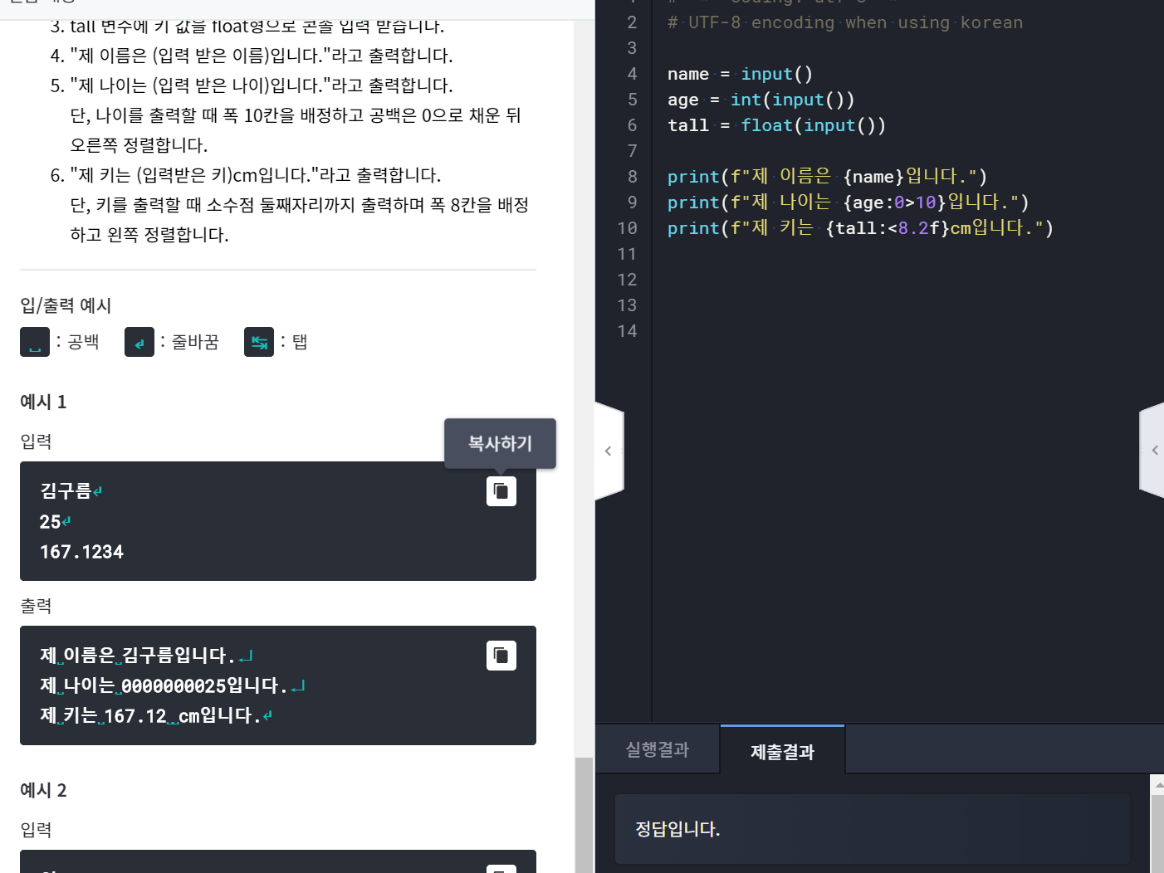


\* 괄호 또한 문자열로 취급할 때 : {}를 한번 더 추가하여 ‘{{‘나’}}’처럼 2개를 연속해서 사용

* f문자열 포매팅

문자열 앞에 f만 붙여 쉽게 사용할 수 있다





* 문자열 함수

기본함수(유용하고 직관적인 내장함수)를 가지고 있다.

함수 : 특정 기능을 수행하기 위해 제작된 명령 묶음이다.

전달 인자 : 함수명 옆의 괄호 안에 들어가는 값

\* 함수에 따라 전달 인자가 없을 수도 있다.



형태 : 함수는 변수 이름(혹은 문자열 자체).함수( )

Len(변수 혹은 값 자체) : 문자열 뿐만 아니라 여러 값이 모여있는 자료형의 길이를 반환하는 함수

