

제3장 변수와 데이터형

print() 함수의 서식

```
print("안녕하세요")
print("100")
print("%d" % 100)
print("100+100")
print("%d" %(100+100))
```

안녕하세요
100
100
100+100
200

- 예 : `print("%d" % (100, 200))`
- 예: `print("%d" "%d" % (100))`
- 예 : `print("%d / %d = %d" % (100, 200, 0.5))`

print() 함수를 사용한 다양한 출력

print() 함수에서 사용할 수 있는 서식

서식	예	설명
%d, %x, %o	10, 100, 1234	정수(10진수, 16진수, 8진수)
%f	0.5, 1.0, 3.14	실수(소수점)
%c	"b", "한"	한 글자
%s	"안녕", "abcd", "a"	한 글자 이상 문자열

- 예 : `print("%d / %d = %5.1f" % (100, 200, 0.5))`

```
=====
100 / 200 = 0.5
```

print() 함수를 사용한 다양한 출력

print() 함수를 사용한 깔끔한 출력(Code03-01.py)

```
print("%d" % 123)
print("%5d" % 123)
print("%05d" % 123)
print("%f" % 123.45)
print("%7.1f" % 123.45)
print("%7.3f" % 123.45)
print("%s" % "Python")
print("%10s" % "Python")
```

123
123
00123
123.450000
123.5
123.450
Python
Python

print() 함수를 사용한 다양한 출력

이스케이프 문자

이스케이프문자	역할	설명
\n	새로운 줄로 이동	엔터키 누른 효과
\t	다음 탭으로 이동	Tab 키 누른 효과
\b	뒤로 한 칸 이동	Backspace 누른 효과
\\	출력	
\'	'출력	
\"	"출력	

```
print("\n줄 바꿈\n연습 ")
print("\t탭 키\n연습")
print("글자가 \"강조\"되는 효과1")
print("글자가 \"강조\"되는 효과2")
print("\\\\\\\\ 역슬래시 세 개 출력")
print(r"\n \t \" \"를 그대로 출력")
```

줄 바꿈
연습
탭 키
연습
글자가 "강조"되는 효과1
글자가 "강조"되는 효과2
\\\\ 역슬래시 세 개 출력
\n \t " "를 그대로 출력

변수의 선언과 사용

■ 변수의 선언

- 변수는 어떠한 값을 저장하는 메모리 공간
- 변수를 선언하지 않아도 되지만 긴 코드를 작성할 때는 사용될 변수를 미리 선언하는 것이 더 효율적

• 예

```
boolVar, intVar, floatVar, strVar = True, 0, 0.0, ""
```

```
boolVar = True
intVar = 0
floatVar = 0.0
strVar = ""
```

- 가장 많이 사용하는 변수 : 불형(Boolean), 정수형, 실수형, 문자열

변수의 선언과 사용

■ 변수명 규칙

- 대문자, 소문자 구분
 - 예: `myVar` ≠ `MyVar`
- 문자, 숫자, 언더바(_) 포함 가능. 숫자로 시작 못함.
 - 예: `var2(O)`, `_var(O)`, `var_2(O)`, `2Var(X)`
- 예약어는 변수명으로 사용 불가능
 - `True`, `False`, `None`, `and`, `or`, `not`, `break`, `continue`, `return`, `if`, `else` 등

- `type()` 함수 : 변수가 `bool`, `int`, `float`, `str`형으로 생성된 것을 확인 가능

```
type(boolVar), type(intVar), type(floatVar), type(strVar)
```

출력 결과

```
(<class 'bool'>, <class 'int'>, <class 'float'>, <class 'str'>)
```

변수의 선언과 사용

■ 변수의 사용

- 변수는 값을 담으면(대입하면) 사용 가능. 변수에 있던 기존 값은 없어지고 새로 입력한 값으로 변경됨
- 변수에는 변수의 값이나 계산 결과를 넣을 수도 있음
 - 예: `var2 = 200;` `var1 = var2`
 `var1 = 100 + 100`
- 변수에는 숫자와 변수의 연산을 넣을 수도 있음
 - 예: `var1 = var2 + 100`
- 변수에 연속된 값을 대입하는 방식
 - 예: `var1 = var2 = var3 = var4 = 100`

```
var4 = 100
var3 = var4
var2 = var3
var1 = var2
```

변수의 선언과 사용

■ 변수의 사용

- 변수에 연산 결과를 자신의 값으로 다시 대입하는 방식
 - 예: `var1 = var1 + 200`
- 변수의 데이터 형식은 값을 넣는 순간마다 변경될 수 있음
 - 예: `myVar = 100` # 정수형 변수 생성
 `myVar = 100.0` # 실수형 변수로 변경
- 대입 연산자(=)의 왼쪽에는 무조건 변수만 올 수 있고, 오른쪽에는 무엇이든(값, 변수, 수식, 함수 등) 올 수 있음

기본 데이터형

- 파이썬은 변수 선언이 없으며 변수에 값을 넣는 순간에 변수의 데이터형이 결정됨
- 기본형 : 정수형, 실수형, 불형, 문자열
- 확장형 : 리스트, 튜플, 딕셔너리, 집합
- 숫자형
 - 정수형 : 소수점 없는 수, 정수 크기에 제한없음, 16(8,2)진수
 - 실수형 : 소수점 있는 수
 - 연산 : +, -, *, /, ** (제곱), %(나머지), //(나눈 후 소수점 이하 버림)

기본 데이터형

```

• 예) a = 123;          print(a, "\n")
      a = 100 ** 100 ; print(a, "\n")
      a = 0xff; b = 0o77; c = 0b1111 ; print(a, b, c, "\n")
      a = 3.14; b = 3.14e5;          print(a, b, "\n")
      a = 10; b=20;                  print(a+b, a-b, a/b, "\n")
      a, b = 9, 2;                   print(a**b, a%b, a//b, "\n")

```

123

[illegible]

255 63 15

3.14 314000.0

30 -10 0.5

81 1 4

기본 데이터형

- 불형(bool), 논리형 : 참, 거짓

- 예:

```
a = (100 == 100)
b = (10 > 100)
print(a, b)
```

```
True False
```

- 문자열 : “ ” 혹은 ‘ ’ 사용

```
a = "파이썬"  
b = '파이썬'  
c = ""파이썬""  
d = "파이\n썬"  
e = "파이\\썬"  
print(a, b, c, d, e)
```

파이썬 파이썬 '파이썬' 파이
썬 파이'썬

- 한 줄 주석: #

- 예: # 반지름

- 여러 줄 주석

- 큰따옴표 3개(" ~ "), 작은따옴표 3개(' ~ ')

실습 예제 : codeEx3.py

1. 서일대학교 소프트웨어공학과 1학년 2022**** 홍길동

2. 서일대학교
소프트웨어공학과
1학년 2022****
홍길동

3. 서일대학교 소프트웨어공학과 1학년 2022**** 홍길동

3. 서일대학교 소프트웨어공학과 1학년 2022**** '홍길동'

4. "서일대학교" 소프트웨어공학과 1학년 2022**** '홍길동'

1. 다음 각 보기를 실행하면 다른 값이 출력하는 것을 하나 고르세요.

- ① `print("100")` ② `print(100)`
- ③ `print(50+50)` ④ `print("50+50")`

2. 다음 코드를 실행하면 출력되는 결과를 고르세요.

```
print('%d / %d = %d' % ( 5, 10, 5/10 ))
```

- ① `5 / 10 = 0.5` ② `5 / 10 = 0.50000`
- ③ `5 / 10 = 0` ④ `5 / 10 = 0.00000`

5. 다음 보기 중에서 설명하는 이스케이프 문자를 각각 고르세요.

```
"\n\t\b\\'\\" data-bbox="590 83 659 104"/>
```

- (1) 다음 탭으로 이동
- (2) 뒤로 한 칸 이동
- (3) \를 출력

6. 파이썬에서 제공되는 데이터 형식입니다. 거리가 먼 것 하나?

- ① 불형 : `bool` ② 정수 : `int` ③ 실수 : `float`
- ④ 문자 : `character` ⑤ 문자열 : `str`

7. 다음 코드 중에서 오류가 발생하는 것을 모두 고르세요.

- ① `num1 = 100`
- ② `100 = num1`
- ③ `num1 = num2 = 100`
- ④ `num1 = 100 = num2 = 100`
- ⑤ `num1 = num2 = num3 = 100`

8. 코드를 실행하면 오류가 발생한다. 그 이유를 설명하시오.

```
a=b=10=c=d=20
```

14. 다음은 데이터 형에 대한 설명입니다. 설명이 옳바르지 않은 것을 모두 고르세요.

- ① 정수형은 `<class 'int'>`로 표현된다.
- ② 정수형은 크기가 무제한이다.
- ③ 불형은 `True`, `False`, `Null` 세 값 중 하나가 저장된다.
- ④ 실수형은 소수점이 있다.
- ⑤ 문자는 작은 따옴표, 문자열은 큰 따옴표로 지정한다.