

베이직 파이썬 (2)

Today's point!



- ✓ 파이썬 데이터 타입
- ✓ 파이썬 변수의 정의
- ✓ 변수의 특성 이해하기
- ✓ 파이썬 연산자

파이썬 변수의 정의

1

파이썬 데이터 타입



int

float

string

bool

tuple

set

list

dictionary

파이썬 변수의 정의

```
Apple = 1  
print (Apple)
```

```
Apple = '사과'  
print (Apple)
```

변수의 이름

변수 정의 시, 변수 이름에는 영문 대소문자, 숫자, 언더스코어를 사용할 수 있다.

- 단, '약속어' 는 변수의 이름으로 사용할 수 없다.
- 변수에는 파이썬의 데이터 타입(자료형)이 들어갈 수 있다.
- 변수는 정의 이후 변경될 수 있다.

파이썬 변수의 정의 - 정수 및 소수

```
A = 1.2  
print (A)  
  
B = 2  
print (B)  
  
print (A+B)
```

Info

- 변수에 정수(int), 소수(float)를 정의할 수 있다.
- 정수, 소수로 정의된 변수는 사칙연산 및 수학적 연산에 활용할 수 있다.
- +(덧셈) -(뺄셈) *(곱셈) /(나눗셈) **(거듭제곱) //(나눗셈 몫) %(나눗셈 나머지) 등의 기본 연산을 적용할 수 있다.
- 정수와 정수, 소수와 소수, 정수와 소수 간의 연산 결과는 변수에 저장할 수 있다.
- 소수와 소수, 정수와 소수 간의 연산 결과는 소수가 된다.

파이썬 변수의 정의 - 문자열 (string)

```
word = 'hello'
print (A)

sentence = "good morning."
print (B)

print (A+B)
```



Info

- 변수에 문자 하나, 문자열(문자 여러 개)를 작은따옴표(' ') 혹은 큰따옴표(" ")를 사용하여 정의할 수 있다.
- 작은 따옴표로 시작하면 작은 따옴표로, 큰 따옴표로 시작하면 큰 따옴표로 문자열 정의를 끝내야 한다.
- * 일부 프로그래밍 언어와 달리, 파이썬은 작은 따옴표와 큰따옴표의 활용에 제한을 두지는 않는다.
- 문자열도 덧셈이 가능하다.
- 파이썬에는 Escape string이라는 특수 문자가 정의되어 있으며, 이를 코딩에 사용한다.

파이썬 변수의 정의 - Bool

```
A = True
print (A)

B = False
print (B)

print (A==B)
```

A	연산자	B	결과
True	and	True	True
False	and	True	False
True	and	False	False
True	or	False	True
False	or	True	True
True	not	-	False
-	not	True	False

Info

- 변수에 참/거짓을 의미하는 bool 값을 정의할 수 있다.
- bool은 값을 조건문 및 반복문 등에 활용할 수 있다.
- == (같은가?) !=(다른가?) 의 파이썬 문법을 활용하여 참/거짓을 구분할 수 있다.
- 논리연산자 and, or, not 등을 활용하여 참/거짓을 연산할 수 있다.
 - and(변수가 모두 True여야 True)
 - or(변수 중 하나만 True여도 True)
 - not(변수를 반대로 뒤집음)

파이썬 연산자

* 할당연산자 (`=`) , 산술연산자 (`>`, `<`) , 논리연산자 (`and`) , 멤버연산자 (`in`) , 식별연산자 (`is`)

A	연산자	B	의미
A	<code>+=</code>	1	A가 정수/소수인 경우, A의 값이 1 증가
A	<code>-=</code>	1	A가 정수/소수인 경우, A의 값이 1 감소
A	<code>></code>	B	A가 B보다 크거나 같은가?
A	<code><</code>	B	A가 B보다 작거나 같은가?
A	<code>==</code>	B	A와 B가 같은가?
A	<code>!=</code>	B	A와 B가 다른가?
A	<code>in</code>	B	A가 B 안에 있는가?
A	<code>not in</code>	B	A가 B 안에 없는가?
A	<code>is</code>	B	A는 B인가?
A	<code>is not</code>	B	A는 B가 아닌가?

파이썬 연산자

* 할당연산자 (+=)

A	연산자	B	의미
A	+=	1	A가 정수/소수인 경우, A의 값이 1 증가
A	-=	1	A가 정수/소수인 경우, A의 값이 1 감소

```
A = 1  
A = A+1  
print (A)
```

```
B = 1  
B += 1  
print (B)
```

파이썬 연산자

* 산술연산자 (>=)

A	연산자	B	의미
A	>=	B	A가 B보다 크거나 같은가?
A	<=	B	A가 B보다 작거나 같은가?

```
A = 1
B = 32
print (A>B)

C = -3
D = -0.2
print (C<=D)
```

파이썬 연산자

* 산술연산자 (>=)

A	연산자	B	의미
A	==	B	A와 B가 같은가?
A	!=	B	A와 B가 다른가?

```
A = 1  
B = 32  
print (A==B)
```

```
C = -3  
D = -0.2  
print (C!=D)
```

파이썬 연산자

* 논리연산자 (and, or, not)

A	연산자	B	결과
True	and	True	True
False	and	True	False
True	and	False	False
True	or	False	True
False	or	True	True
True	not	-	False
-	not	True	False

```
A = True
B = False
print(A and B)
print(A or B)
print(not A)
print(not B)
```

파이썬 연산자

* 멤버연산자 (in)

A	연산자	B	의미
A	in	B	A가 B 안에 있는가?
A	not in	B	A가 B 안에 없는가?

```
A = '사과'
B = ['사과', '배', '참외', '수박', '복숭아']
print(A in B)

C = 'apple'
D = ['Apple', 'pear', 'watermelon', 'peach']
print(C in D)
```

파이썬 연산자

* 식별연산자 (is)

A	연산자	B	의미
A	is	B	A는 B인가?
A	is not	B	A는 B가 아닌가?

```
A = '사과'
B = '사과'
print(A is B)
```

```
C = 12
D = 12
print(C is not D)
```


Summary



- 변수를 정의할 때는 = 를 사용한다.
- 변수 명에는 영어 소문자, 영어 대문자, 숫자, 언더스코어(_)를 사용할 수 있다. (단, 숫자는 맨 처음에 올 수 없다)
- 몇 가지 특수한 영어 단어(예약어)는 변수명으로 사용할 수 없다.
- 변수에는 int, float, bool, string & tuple, set, list, dictionary 의 자료형이 들어갈 수 있다.
- int, float, bool, string 은 수치 자료형으로, 단일 데이터를 저장한다.
- tuple, set, list, dictionary는 컬렉션 자료형으로, 여러 데이터를 저장한다.
- 변수는 연산이 가능하다.
- 파이썬 연산자는 할당연산자 (+=) , 산술연산자 (>=) , 논리연산자 (and) , 멤버연산자 (in) , 식별연산자 (is) 가 있다.