



## • 파이썬 기본 연산 기호

종 류	의 미
+	숫자를 덧셈하거나 문자열을 결합
-	좌항을 우항으로 뺄(또는 부호 변경)
*	숫자를 곱하거나, 문자열을 곱한 수 만큼 반복하여 결합
**	좌항을 우항으로 거듭 제곱
/	좌항을 우항으로 나눔(실수형)
//	좌항을 우항으로 나눔(정수형)
%	좌항을 우항으로 나눈 나머지

ex)

$$2^{**}3 = 8$$

$$7/4 = 1.75$$

$$7//4 = 1$$

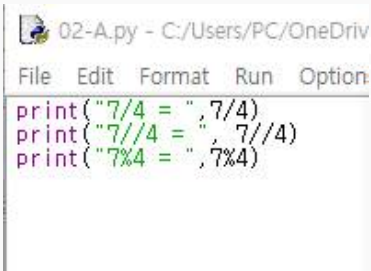
$$7\%4 = 3$$

- 파이썬 연산 순서

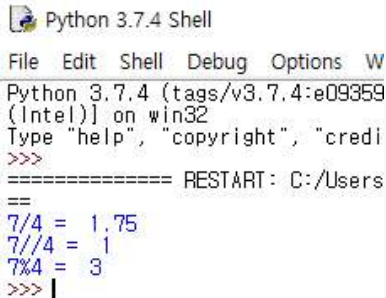
**파이썬은 소괄호만 사용!**  
**중괄호({}), 대괄호([]) xX**

$5+[4*\{3+(1+2)\}] \longrightarrow 5+(4*(3+(1+2)))$

## • 연산자 연습 : 02-A.py



```
02-A.py - C:/Users/PC/OneDriv
File Edit Format Run Option
print("7/4 = ", 7/4)
print("7//4 = ", 7//4)
print("7%4 = ", 7%4)
```

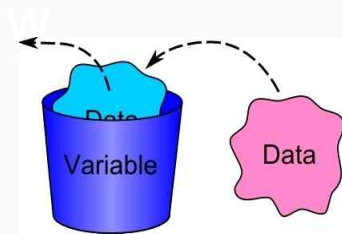


```
Python 3.7.4 Shell
File Edit Shell Debug Options W
Python 3.7.4 (tags/v3.7.4:e09359
(Intel)] on win32
Type "help", "copyright", "credi
>>>
===== RESTART: C:/Users
==
7/4 = 1.75
7//4 = 1
7%4 = 3
>>> |
```

## • 변수

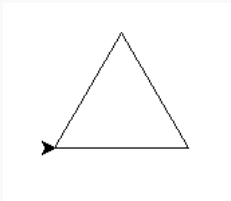
프로그램을 만드는 데  
필요한 정보 저장

정보 보관소 역할



## • 들어가기 전에

- 지난 시간 복습 : 정삼각형 그리기



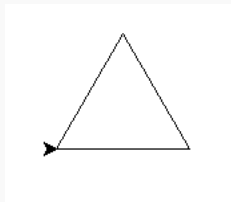
```
import turtle as t
t.shape("turtle")
t.fd(100)
t.lt(120)
t.fd(100)
t.lt(120)

t.fd(100)
t.lt(120)
```

## • 예제 : 02-B.py

- 변수를 이용해 삼각형 그리기

```
02-B.py - C:/Users/PC  
File Edit Format Run  
import turtle as t  
d=100  
t.fd(d)  
t.lt(120)  
t.fd(d)  
t.lt(120)  
t.fd(d)  
t.lt(120)  
|
```



d 수치에 따라  
삼각형 크기가 변함  
-->효율적!

• 예제 : 대화형 셀에서의 변수 활용

```
>>> a=3      #변수 a에 3 저장
>>> a        #a 확인 --> a= 으로 하면 안됨.
3
>>> b=1.1+2   #변수 b에 1.1+2=3.1 저장
>>> b        #b 확인
3.1
>>> d=2       #변수 d에 2 저장
>>> d=d+1     #d에 1더한 값을 다시 d에 저장
>>> d        #d확인
3
>>> |
```





- for문 연습 : 02-D.py

```
for x in range(3):  
    print(100)  
    print(200)  
print(300)
```

C/C++과는 달리  
괄호가 없기 때문에  
들여쓰기 주의해야 함 !!

```
100  
200  
100  
200  
100  
200  
300  
>>>
```

## • for문을 사용한 거북이 프로그램 : 02-E.py

```
import turtle as t
```

```
#삼각형
```

```
for x in range(3):  
    t.fd(100)  
    t.lt(120)
```

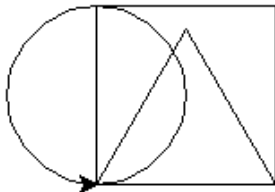
```
#사각형
```

```
for x in range(4):  
    t.fd(100)  
    t.lt(90)
```

```
#원그리기
```

```
t.circle(50)
```

```
import turtle as t
```



진

```
#원그리기
```

```
t.circle(50) #반지름이 50인 원
```

- List로 알아보는 Range?

“범위”

Ex) for문의 range(5), range(10),,,  
반복 범위

배열

```
>>> list(range(5))  
[0, 1, 2, 3, 4]  
>>> list(range(0,5))  
[0, 1, 2, 3, 4]  
>>> list(range(1,11))  
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
>>> |
```

느낌

• 완전 기본 for문 : 0

```
print("[0~4]")
for x in range(5)
    print(x) #변

print("[1~10]")
for x in range(1
    print(x) #변
```

[0~4]

0

1

2

3

4

[1~10]

1

2

3

4

5

6

7

8

9

10

>>> |

(1,2,3,4)까지 반복

까지 10번 반복  
[!!!!]

## • 2주차 퀴즈 1 : 1~10까지 sum 구하기

1. sum을 저장할 변수 s를 정의하세요.
2. for문을 이용하세요.(for x in range(?? , ??))
3. for문이 한 번 돌 때마다 현재의 x 값과 현재의 sum값을 출력하도록 만들어보세요.

### <예상 결과창>

```
x: 1    sum: 1
x: 2    sum: 3
x: 3    sum: 6
x: 4    sum: 10
x: 5    sum: 15
x: 6    sum: 21
x: 7    sum: 28
x: 8    sum: 36
x: 9    sum: 45
x: 10   sum: 55
>>> |
```

- 퀴즈 정답 : 02-Quiz.py

```
s=0
for x in range(1,11):
    s+=x
    print("x: ",x," sum: ",s)
```

- 정오각형 그리기 : 02-G.py

```
import turtle as t

n=5 #n각형을 그릴 예정
t.color("purple") #펜 색깔 보라색
t.begin_fill() #색칠할 영역 시작
for x in range(n): #n번 반복
    t.fd(100)
    t.lt(360/n)
t.end_fill() #색칠 영역 끝내기
```

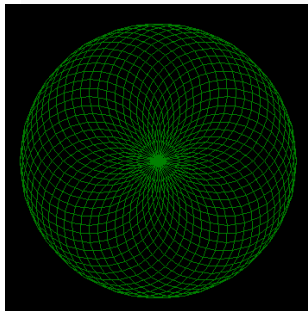




- 반복된 원 그리기 : 02-H.py

```
import turtle as t

n=50
t.bgcolor("black") #배경 검은색
t.color("green") #펜 녹색
t.speed(0) #가장 빠른거북이
for x in range(n):
    t.circle(80)
    t.left(360/n)
```



## • 반복된 선 그리기 : 02-l.py

```
import turtle as t

angle = 89 #angle 조
t.bgcolor("black")
t.color("yellow")
t.speed(0)
for x in range(200):
    t.fd(x) #x가 커지면서 선이 길어짐
    t.lt(angle) #왼쪽으로 angle만큼
```

