

데이터 처리를 위한 Python 프로그래밍 입문

2-1강. 연산

ERICA 2018-2

강의 내용

- ▶ 산술 연산자
 - ▶ 산술 연산자 기호
 - ▶ 숫자 연산
 - ▶ 문자열 연산
- ▶ 결합 법칙

산술 연산자 기호

▶ 연산자 기호별 기능

연산자	내 용
+, -, *, /	더하기, 빼기, 곱하기, 나누기
%	나머지 (Module)
//	몫
**	제곱

숫자 연산(1/2)

▶ 산술 연산자 실행

```
>>> 9*4
36
>>> 9/4
2.25
>>> 9//4
2
>>> 9%4
1
>>> 3+4
7
>>> 3-4
-1
>>> |
```

숫자 연산(2/2)

- ▶ 함수에 산술연산자 활용
 - ▶ 함수 괄호 () 안에 문자열과 연산자를 넣음
 - ▶ 구분 : 콤마(,)

```
>>> print("9를 4로 나누면?", 9/4)
9를 4로 나누면? 2.25
>>>
>>>
>>> print("3에서 4를 빼면?", 3-4)
3에서 4를 빼면? -1
>>>
>>>
>>> print("9를 4로 나누면?", 9//4, "나머지는?", 9%4)
9를 4로 나누면? 2 나머지는? 1
>>>
>>>
```

문자열 연산(1/4)

- ▶ 문자열과 문자열 더하기(+)
 - ▶ 문자열과 문자열이 연결

```
>>> head = "데이터 분석을 위한"  
>>> tail = "파이썬 프로그램"  
>>>  
>>> head + tail  
'데이터 분석을 위한파이썬 프로그램'  
>>>  
>>>
```

문자열 연산(2/4)

- ▶ 문자열 곱셈(*)
 - ▶ 문자열이 곱셈만큼 반복 적용

```
>>>  
>>> a = "Good!!!"  
>>>  
>>> a*3  
'Good!!!Good!!!Good!!!'  
>>>  
>>>
```

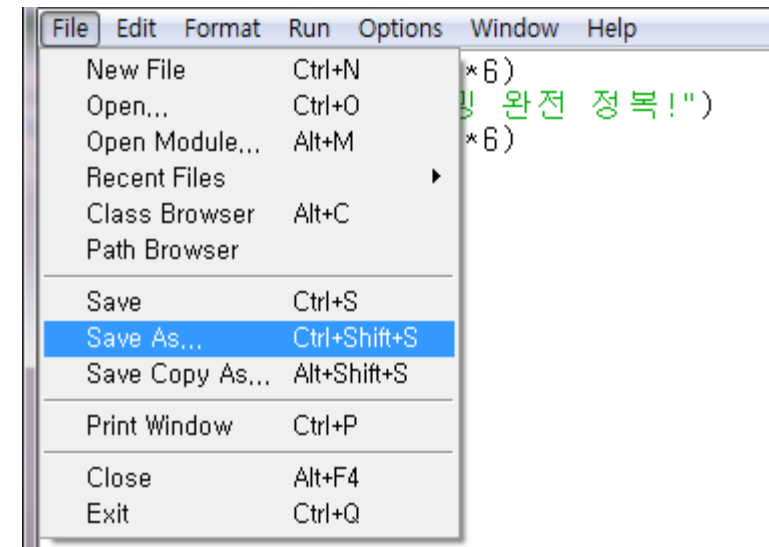
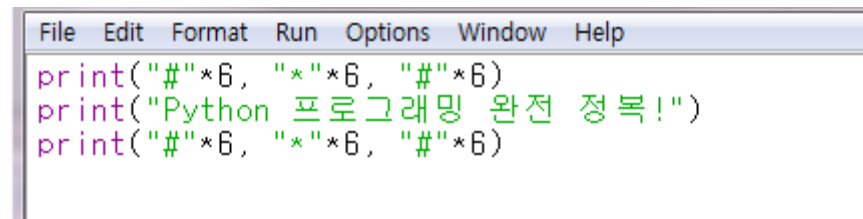
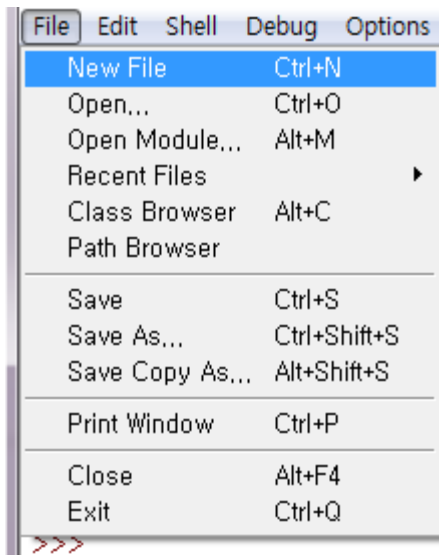
```
print("#"*6, "*"*6, "#"*6)
```

```
print("Python 프로그래밍 완전 정복!")
```

```
print("#"*6, "*"*6, "#"*6)
```

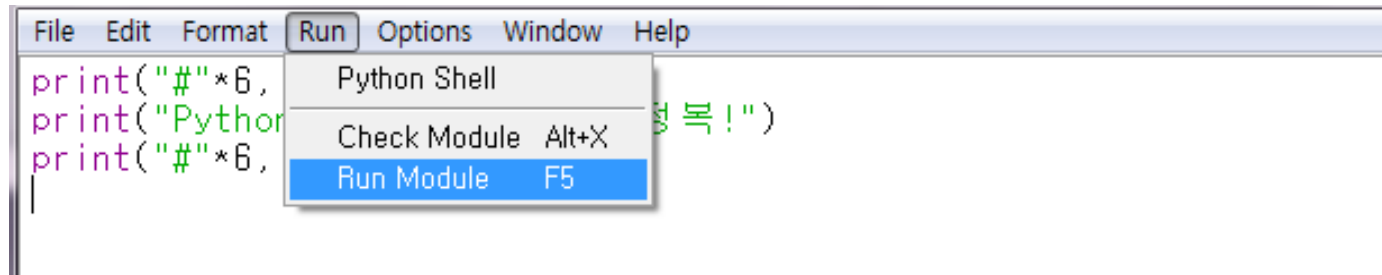

문자열 연산(3/4)

- ▶ 스크립트 모드로 실행
 - ▶ File-New File 선택하여 스크립트 모드 실행
 - ▶ 코드 입력
 - ▶ 다른 이름으로 저장 : 확장자 .py 확인



문자열 연산(4/4)

- ▶ 스크립트 모드로 실행(계속)
 - ▶ 스크립트 모드에서 Run-Run Module 선택



- ▶ 대화형 모드에서 실행 결과 확인

```
##### ***** #####  
Python 프로그래밍 완전 정복!  
##### ***** #####  
>>>
```

- 프로그램 만들어 실행했을 때
화면 상단에 프로그램 제목을 표현하는데 응용

결합 법칙(1/3)

▶ 연산자 결합 법칙(Operator Associativity)

연산자	결합 법칙
** (제곱)	오른쪽에서 왼쪽으로
- (음수)	왼쪽에서 오른쪽으로
* (곱), / (나누기), // (몫), % (나머지)	왼쪽에서 오른쪽으로
+ (더하기), - (빼기)	왼쪽에서 오른쪽으로

▶ 예

▶ $2^{**}3^{**}4$: 실제 $2^{**}(3^{**}4)$

▶ $4*2*7$: 실제 $(4*2)*7$

▶ 사용자 괄호로 우선 계산 지정 : ()

▶ 예 : $(2+4)*5$, $4*(26-9)$

결합 법칙(2/3)

▶ 연산자 결합 법칙 실행

```
>>> (2+4)*5
30
>>> 4*(26-9)
68
>>> (4+5)**(9-6)
729
>>> 4*6-5
19
>>> 9+54/6
18.0
```

```
>>> 2**4+5
21
>>> 6*5**2
150
>>> 5*4+6
26
>>> 26+20/2
36.0
>>> (511//31)**(9-6)
4096
>>> 8.2*4+2
34.8
>>> 20+9/4.5
22.0
>>> (15.1/4)**(6-4)
14.250625
>>>
```

결합 법칙(3/3)

- ▶ 연산자 결합 법칙 실행(계속)
 - ▶ 괄호() 지정으로 혼돈 해결

```
>>>  
>>> 5+(4*(3+(1+3)))  
33  
>>> (5+4)*(3+(1+3))  
63  
>>>
```

Thank you