

3주차 실습

기본 연산자 연습

IDLE에서 다음 수식을 직접 실행시켜보자.

1) $31 + 26$

2) $511 - 54$

3) $5 * 15$

4) $96 / 4$

- 다음 문장의 실행 결과를 예측하여 보고 결과를 출력하라

>>> 20 % 6

정답 :

>>> 21 / 5

정답 :

>>> 21 // 5

정답 :

** 연산자

다음 수식을 직접 실행시켜보자.

1) $2 ** 4$

2) $4 ** 6$

다음 수식을 직접 실행시켜보자.

1) $3.1 ** 4.2$

2) $2.6 ** 5.4$

연산자 우선 순위

다음 수식을 직접 실행시켜보자.

1) $(2 + 4) * 5$

2) $4 * (26 - 9)$

3) $(4 + 5) ** (9 - 6)$

다음 수식을 직접 실행시켜보자.

1) $4 * 6 - 5$

2) $9 + 54 / 6$

다음 수식을 직접 실행시켜보자.

1) $2 ** 4 + 5$

2) $6 * 5 ** 2$

연산자들의 복합

다음 수식을 직접 실행시켜본 후 결과를 확인해보자.

1) $5 * 4 + 6$

2) $26 + 20 / 2$

3) $(511 // 31) ** (9 - 6)$

실수가 포함된 수식 계산하기

다음 수식을 직접 실행시켜본 후 결과를 확인해보자.

- 1) $8.2 * 4 + 2$
- 2) $20 + 9 / 4.5$
- 3) $(15.1 / 4) ** (6 - 4)$

- 다음 문장의 실행 결과를 예측해보고 실행하여 비교해 보아라

>>> 5 + 12 // 7 * 3

정답 :

>>> 15 / 2 + 21 % 4 - 2 ** 3

정답 :

자료형 확인

`type`함수를 이용하여 다음에 해당하는 자료형을 확인해 보자.

- 1) 10
- 2) 10.0

string 형의 예

"Boy"

"I am a boy"

```
>>> tom = 'Boy'
>>> tom
'Boy'
>>> bob = 'I am a boy'
>>> bob
'I am a boy'
```

string 형의 예

- string형은 따옴표인 quote를 이용하여 나타낸다.
- Python에서는 string형을 만들 때 다음 4가지 형식의 Quate가 모두 허용

' '	" "	''' '''	""" """
-----	-----	---------	---------

"	string 내에 " 를 포함해야 하는 경우
"""	string 내에 "를 포함해야 하는 경우
''' '''	여러 문장을 사용하고, string 내에 '''를 포함해야 하는 경우
""" """	여러 문장을 사용하고, string 내에 "를 포함해야 하는 경우

string 형의 예

→ "He is a smart boy." my teacher said

```
>>> print("He is a smart boy." my teacher said')
" He is a smart boy." my teacher said
>>>
```

→ He's a smart and diligent boy.

```
>>> print("He's a smart and diligent boy.")
He's a smart and diligent boy.
>>>
```

→ print('He's a smart and diligent boy.')

```
>>> print('He's a smart and diligent boy.')
SyntaxError: invalid syntax
>>> # " " 대신 ' ' 를 사용하는 경우 Syntax 에러 발생
```

string 형의 예

- 두 문자열 연결하여 새로운 문자열 생성

```
>>> print("안녕"+"하세요")
안녕하세요
>>> print("안녕하세요"+"!")
안녕하세요!
```

- 문자열과 숫자 사이에는 사용할 수 없음

```
>>> print("안녕하세요"+1)
Traceback (most recent call last):
  File "<pyshell#8>", line 1, in <module>
    print("안녕하세요"+1)
TypeError: can only concatenate str (not "int") to str
```

- 문자열은 문자끼리, 숫자는 숫자끼리 연결
- 문자열과 숫자 연결하여 연산하려면 큰따옴표 붙여 문자열로 인식

string 형의 예

“문자(열)” + “문자(열)” + ... + “문자(열)”

각 문자 또는 문자열을 병합하여 하나의 문자로 생성

제목 표시줄을 출력해 보자.

- 1) 출력할 내용은 '====Python Program====' 이다.
- 2) '====', 'Python Program', '====' 세 개의 문자열로 나뉘어져 있다.
- 3) '+' Operator를 사용해서 출력한다.

```
>>> start = '====='  
>>> title = 'Python Program'  
>>> finish = '====='  
>>> print(start + title + finish)  
====Python Program====  
>>>
```

- 다음 문장의 실행 결과를 예측해보고 실행하여 비교해 보아라

```
>>> name = 'Gildong'
```

```
>>> location = 'Seoul'
```

```
>>> print('Hello, My name is '+ name +'. And I live in ' + location + '.')  
(      )
```

string 형의 예

“문자(열)” * n (n은 정수)

문자 또는 문자열을 n의 배수만큼 반복하여 하나의 문자로 생성

제목 표시줄을 출력해 보자.

- 1) 출력할 내용은 '====Python Program====' 이다.
- 2) '=' , 'Python Program', '=' 세 개의 문자열로 나뉘어져 있다.
- 3) '*' Operator를 사용해서 출력한다.

```
>>> start = '=' * 10
>>> title = 'Python Program'
>>> finish = '=' * 10
>>> print(start + title + finish)
====Python Program====
>>>
```


- 다음 문장의 실행 결과를 예측해보고 실행하여 비교해 보아라

```
>>> string1 = 'Hello,'  
>>> string2 = ' Python'  
>>> star = '*****'  
>>> star = '***'  
>>> print(star * 3 + string1 + string2 + star*3)  
(          )
```

입력 예제

variable_name=input()	사용자로부터 입력을 받는다.
variable_name=input('문자열')	'문자열'에 해당하는 내용을 출력 후 사용자로부터 입력을 받는다.

```
>>> name = input()  
Gildong  
>>> name  
'Gildong'  
>>>
```

```
>>> name = input('What is your first name? ')  
What is your first name? Gildong  
>>> name  
'Gildong'  
>>>
```

입력 예제

다음 input 명령문을 활용하여 스케줄을 작성해보자.

- 1) applicant = input("Enter the applicant's name: ")
- 2) interviewer = input("Enter the interviewer's name: ")
- 3) time = input("Enter the appointment time: ")

```
>>> applicant=input("Enter the applicant's name: ")
Enter the applicant's name: Gildong
>>> interviewer=input("Enter the interviewer's name: ")
Enter the interviewer's name: Minwoo
>>> time=input("Enter the appointment time: ")
Enter the appointment time: 14:00
>>> applicant
'Gildong'
>>> interviewer
'Minwoo'
>>> time
'14:00'
>>>
```

- 다음 소스 코드를 완성하여 정수 3개를 입력 받고 합계가 출력되게 하라.

```
print(a + b + c)
```

실행결과

```
정수 3개를 입력하시오 : -10 20 30  
40
```

출력 예제

<code>print('문자열')</code>	'문자열'을 화면에 출력해준다.
<code>print(variable_name)</code>	변수 <code>variable_name</code> 에 해당하는 값을 화면에 출력해준다.
<code>print('문자열', variable_name)</code>	'문자열'과 변수 <code>variable_name</code> 에 해당하는 값을 연속해서 화면에 출력해준다.

```
>>> print('True Love')
```

```
True Love
```

```
>>>
```

```
>>> number1 = 26
>>> number2 = 31
>>> sum = number1 + number2
>>> print(sum)
57
>>>
```

출력 예제

과목의 성적을 나타내주는 프로그램을 작성하고자 한다.

- 1) math = 54
- 2) english = 72
- 3) history = 96
- 4) print(math, english, history)

```
>>> math = 54
>>> english = 72
>>> history = 96
>>> print(math, english, history)
54 72 96
>>>
```

출력 예제

길이와 폭으로 넓이를 구해서 나타내주는 프로그램을 작성하고자 한다.

- 1) length = 9
- 2) width = 6
- 3) print('area = ', length * width)

```
>>> length = 9
>>> width = 6
>>> print('area = ', length * width)
area = 54
>>>
```

- 다음 문장의 실행 결과를 예측해보고 실행하여 비교해 보아라

```
>>> name = 'Gildong'
```

```
>>> print('Hello, my name is', name)
```

```
(    )
```


- 다음 소스코드를 완성하여 날짜와 시간이 출력되게 만들어라.

```
year = 2000
```

```
month = 10
```

```
day = 27
```

```
hour = 11
```

```
minute = 43
```

```
second = 59
```

```
print(year, month, day, ①_____ )
```

```
Print(hour, minute, second ②_____ )
```

실행결과

```
2000/10/27 11:43:59
```

강제 형변환

```
>>> n = 7
>>> n = float(n)
>>> type(n)
<class 'float'>
>>> n
7.0
>>> n = input('enter the number : ')
enter the number : 7
>>> n = int(n)
>>> type(n)
<class 'int'>
>>> n
7
```

자료형 변환

연필과 펜의 구입하는 개수에 따라 총 가격을 반환해주는 프로그램을 작성해보자.

- 1) 연필은 1000원 펜은 2000원이다.
- 2) 변수 num_pencil, num_pen은 input함수를 통해 각각 연필과 펜의 개수를 입력 받는다.
- 3) 변수 total_price는 총 가격을 나타낸다.

```
>>>
```

```
연필 = 1000원
```

```
펜 = 2000원
```

```
연필은 몇 개를 구입하시겠습니까? 2
```

```
펜은 몇 개를 구입하시겠습니까? 4
```

```
총 가격은 10000 원 입니다.
```

```
>>>
```

Boolean형 과 비교/논리 연산자

어느 놀이 공원에서는 여자 신입생에게는 20% 할인 혜택을 제공한다고 한다.

▷ 수지는 여자이고 2학년이다. 수지가 할인 혜택을 받을 수 있는 지 Boolean식을 사용하여 확인해보자.

(여성은 female, 남성은 male로 한다.)

- 문제 해결 알고리즘

수지 = '여자'

수지의 학년 = 2

만약 (수지의 성별 == '여자') and (수지의 학년 == 1)

수지는 할인을 받는다.

- 프로그램

```
>>> student_gender = 'female'
>>> grade = 2
>>> (student_gender == 'female') and (grade == 1)
False
>>>
```

Boolean형 과 비교/논리 연산자

어느 회사의 입사 조건은 토익 점수가 800점 이상이거나 교양 영어에서 A학점이 만족되어야 지원 가능하다고 한다.

길동이는 토익이 900점이고 교양 영어에서 B학점이다. 길동이가 이 회사에 지원이 가능한지 Boolean 식을 사용하여 확인해보자.

- 문제 해결 알고리즘

길동의 토익 점수 = 900
길동의 영어 학점 = 'B'
만약 (길동의 토익 점수 >= 800) or (길동의 영어 학점 == 'A')
길동은 회사에 지원할 수 있다.

- 프로그램

```
>>> toeic_score = 900
>>> gradeOfEnglish = 'B'
>>> (toeic_score >= 800) or (gradeOfEnglish == 'A')
True
>>>
```

- 표준 입력으로 국어, 영어, 수학, 과학 점수를 입력합니다.
평균 점수를 출력하는 프로그램을 만드세요.
단, 평균 점수를 출력 할 때는 소수점 이하 자리는 버립니다. (정수로 출력)
코드를 작성하여 실행결과대로 나오는지 확인

- 실행결과**

정수 4개를 입력하시오 : 83 92 87 90
88

정수 4개를 입력하시오 : 32 53 22 95
50

- 표준 입력으로 년, 월, 일, 시, 분, 초가 입력되게 합니다.
입력된 날짜와 시간을 년-월-일 T시:분:초 형식으로 출력하라.
코드를 작성하여 실행결과대로 나오는지 확인

- 실행결과**

년, 월, 일, 시, 분, 초를 입력하시오 2017 10 27 11 43 59
2017-10-27T11:43:59

년, 월, 일, 시, 분, 초를 입력하시오 2019 09 15 10 47 56
2019-09-15T10:47:56

실습 제출 형식

- 파일 이름 : p본인학번_주차.py or p본인학번_주차.txt
- 주석 필수