#part2.변수와 연산  
  
#변수 사용하기  
#variable.py 파일 생성  
#print('안녕! 나는 지구인이야. 나는 다리가 4개 있어')  
#powershell에서 저장한 코드 실행 python variable.py  
#print('안녕! 나는', '지구인', '이야. 나는 다리가', '4, '개 있어') \*4

#변수의 선언  
#identity = '지구인'    
#number\_of\_legs = 4  
#변수의 사용  
#print('안녕! 나는', identity, '이야. 나는 다리가', number\_of\_legs, '개 있어.')  
#identity = '한국인'  
#number\_of\_legs = 2  
#print('안녕! 나는', identity, '이야. 나는 다리가', number\_of\_legs, '개 있어.')

변수 사용하기 실습(1)  
name = '지영'  
  
변수 사용하기 실습(2)  
print문이 선언된 season출력  
seson = '겨울'  
print('지금은', season, '입니다.')  
  
주석  
코드를 설명하기 위해 코드에 적어 놓은 프로그래밍 언어가 무시하는 문자  
#을 쓰고 오른쪽에 주석 입력  
여러줄 주석 처리는 """~~~~"""로 내용 둘러싸줌  
  
주석-실습(1)  
print("주석은")  
print("이 프린트 문을 주석처리 하세요")  
print("실행되지 않습니다.")  
  
주석-실습(2)  
print("여러줄의 주석은")  
"""  
여러줄의  
주석은  
어떻게  
처리할까요?  
"""  
print("따옴표 3개로 감싸서 처리합니다.")  
  
숫자와 문자열  
my\_name = 'Python' # 문자열 (사람을 위한 텍스트를 프로그래머가 부르는 방법)  
my\_age = 2016 - 1991 # 숫자  
print(my\_name, '은 이제', my\_age, '살')

my\_next\_age = my\_age + 1  
print('내년에는', my\_next\_age, '살')  
  
multiply = 9 \* 9 # = 81  
divide = 30 / 5 # = 6  
power = 2 \*\* 10 # = 1024  
reminder = 15 % 4 # =3  
print(multiply, divide, power, reminder)  
  
text = '2015' + '1991'  
number = 2015 + 1991  
print(text)  
print(number)

text\_minus =  text - '1991' # ??

텍스트는 더하기만 가능하고 빼기나 다른 계산은 불가능해서 오류남

숫자 : 수학 연산 가능(+ - / %)  
"문자열" : 화면에 그대로 출력 가능, 따옴표로 둘러싸서 표시

숫자와문자열-실습(1)

a=33  
b=3  
  
sumation = a + b  
multiply = a \* b  
divide = a / b  
reminder = a % b  
power = 33 \*\* 3  
  
print("sumation은 {}입니다.".format(sumation))  
print("multiply은 {}입니다.".format(multiply))  
print("divide는 {}입니다.".format(divide))  
print("reminder는 {}입니다.".format(reminder))  
print("power는 {}입니다.".format(power))

숫자와문자열-실습(2)

birth\_year = '1985'  
birth\_date = '0502'  
year\_and\_date = birth\_year + birth\_date  
  
print("year\_and\_date : {}".format(year\_and\_date))

REPL  
powershell에 파일 이름 없이 python입력하면 나오는 창  
python코드들을 한 줄 씩 입력하면서 테스트 해볼 수 있음  
종료는 exit()

Read : 코드를 읽어서  
Eval : 읽은 코드를 실행하고  
Print : 실행한 결과 출력하는  
Loop : 루프

Shell사용법

pwd(Print Working Directory) : 현재 폴더 경로 출력

ls(List Segments) : 현재 폴더 내용물 출력

cd <폴더명>(Change Directory) : 다른 폴더로 이동

cd .. : 상위 폴더로 이동

cp(Copy) : 파일을 다른 이름으로 복사

rm(Remove) : 파일 삭제

파일 자동 완성 : 파일 앞 1~2글자 입력 후 tab키

Shell : 그래픽 Shell, 명령줄 Shell