
Python

제어문/함수

김선녕(sykim.lecture@gmail.com)

제어문

함수/사용자 입력과 출력
파일읽고쓰기

비교연산자	설명
$x < y$	x가 y보다 작다
$x > y$	x가 y보다 크다
$x == y$	x와 y가 같다
$x != y$	x와 y가 같지 않다
$x \geq y$	x가 y보다 크거나 같다
$x \leq y$	x가 y보다 작거나 같다

```
In [2]: 1 money = 1
        2 if money:
        3     print("택시를 타고 가라")
        4 else:
        5     print("걸어 가라")
```

택시를 타고 가라

```
In [3]: 1 x = 3
        2 y = 2
        3 x > y
```

Out [3]: True

```
In [4]: 1 x < 2
```

Out [4]: False

```
In [5]: 1 x == y
```

Out [5]: False

```
In [6]: 1 x != y
```

Out [6]: True

```
In [7]: 1 money = 2000
        2 card = 1
        3 if money >= 3000 or card:
        4     print("택시를 타고 가라")
        5 else:
        6     print("걸어 가라")
```

택시를 타고 가라

연산자	설명
x or y	x와 y 둘 중에 하나만 참이면 참이다
x and y	x와 y 모두 참이어야 참이다
not x	x가 거짓이면 참이다

in	not in
x in 리스트	x not in 리스트
x in 튜플	x not in 튜플
x in 문자열	x not in 문자열

```
In [8]: 1 1 in [1,2,3]
```

```
Out[8]: True
```

```
In [9]: 1 1 not in [1,2,3]
```

```
Out[9]: False
```

```
In [10]: 1 'a' in ('a','b','c')
```

```
Out[10]: True
```

```
In [11]: 1 'j' not in 'python'
```

```
Out[11]: True
```

```
In [12]: 1 pocket = ['paper', 'cellphone', 'money']  
2 if 'money' in pocket:  
3     print("택시를 타고 가라")  
4 else:  
5     print("걸어 가라")
```

```
택시를 타고 가라
```

```
In [13]: 1 pocket = ['paper', 'cellphone']
          2 card = 1
          3 if 'money' in pocket:
          4     print("택시를 타고 가라")
          5 else:
          6     if card:
          7         print("택시를 타고 가라")
          8     else:
          9         print("걸어 가라")
```

택시를 타고 가라

```
In [14]: 1 pocket = ['paper', 'cellphone']
          2 card = 1
          3 if 'money' in pocket:
          4     print("택시를 타고 가라")
          5 elif card:
          6     print("택시를 타고 가라")
          7 else:
          8     print("걸어 가라")
```

택시를 타고 가라

In [15]:

```
1 treeHit = 0
2 while treeHit < 15:
3     treeHit = treeHit + 1
4     print("나무를 %d 번 찍었습니다" % treeHit)
5     if treeHit == 15:
6         print("나무 넘어갑니다.")
```

나무를 1 번 찍었습니다
나무를 2 번 찍었습니다
나무를 3 번 찍었습니다
나무를 4 번 찍었습니다
나무를 5 번 찍었습니다
나무를 6 번 찍었습니다
나무를 7 번 찍었습니다
나무를 8 번 찍었습니다
나무를 9 번 찍었습니다
나무를 10 번 찍었습니다
나무를 11 번 찍었습니다
나무를 12 번 찍었습니다
나무를 13 번 찍었습니다
나무를 14 번 찍었습니다
나무를 15 번 찍었습니다
나무 넘어갑니다.

while 문 직접만들기

```
In [13]: 1 prompt = ""  
2         1.Add  
3         2.Del  
4         3.List  
5         4.Quit  
6         Enter number:""
```

```
In [14]: 1 number = 0  
2 while number != 4:  
3     print(prompt)  
4     number = int(input())
```

```
1     1.Add  
2     2.Del  
3     3.List  
4     4.Quit  
5     Enter number:  
6
```

```
1     1.Add  
2     2.Del  
3     3.List  
4     4.Quit  
5     Enter number:  
6  
7  
8  
9  
10  
11  
12  
13  
14
```

```
In [*]: # break문 이용하여 자판기 작동 과정 만들기
coffee = 10
while True:
    money = int(input("돈을 넣어 주세요."))
    if money == 300:
        print("커피를 줍니다")
        coffee = coffee - 1
    elif money > 300:
        print("거름 돈 %d를 주고 커피를 줍니다." % (money - 300))
        coffee = coffee - 1
    else:
        print("돈을 다시 돌려주고 커피를 주시 않습니다.")
        print("남은 커피의 양은 %d개입니다." % coffee)
    if not coffee:
        print("커피가 다 떨어졌습니다. 판매를 중지합니다.")
        break
```

돈을 넣어 주세요.500

거름 돈 200를 주고 커피를 줍니다.

돈을 넣어 주세요.300

커피를 줍니다

돈을 넣어 주세요.100

돈을 다시 돌려주고 커피를 주시 않습니다.

남은 커피의 양은 8개입니다.

돈을 넣어 주세요.

```
In [1]: 1 a = 0
        2 while a < 10:
        3     a = a + 1
        4     if a % 2 == 0:
        5         continue
        6     print(a)
```

```
1
3
5
7
9
```

```
In [1]: 1 test_list = ['one', 'two', 'three']
        2 for i in test_list:
        3     print(i)
```

```
one
two
three
```

```
In [2]: 1 a = [(1,2),(3,4),(5,6)] # first, last
        2 for (first, last) in a:
        3     print(first + last)
```

```
3
7
11
```

```
In [3]: 1 # 점수를 차례로 검사해서 합격여부를 통보해 주는 프로그램
        2 marks = [90,25,67,45,80]
        3 number = 0
        4 for mark in marks:
        5     number = number + 1
        6     if mark >= 60:
        7         print("%d번 학생은 합격입니다." % number)
        8     else:
        9         print("%d번 학생은 불합격입니다." % number)
```

```
1번 학생은 합격입니다.
2번 학생은 불합격입니다.
3번 학생은 합격입니다.
4번 학생은 불합격입니다.
5번 학생은 합격입니다.
```

for문과 continue, range() 함수

13

```
In [6]: 1 # continue문
        2 marks = [90, 25, 67, 45, 80]
        3 number = 0
        4 for mark in marks:
        5     number = number + 1
        6     if mark < 60: continue
        7     print("%d번 학생 축하합니다. 합격입니다." % number)
```

1번 학생 축하합니다. 합격입니다.
3번 학생 축하합니다. 합격입니다.
5번 학생 축하합니다. 합격입니다.

```
In [8]: 1 # range 함수
        2 a = range(10) # 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
        3 a
```

Out [8]: range(0, 10)

```
In [9]: 1 a = range(1, 11) # 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
        2 a
```

Out [9]: range(1, 11)

```
In [10]: 1 sum = 0
        2 for i in range(1, 11):
        3     sum = sum + i
        4 print(sum)
```

55

리스트 안에 for문(구구단)

```
In [12]: 1 # 구구단
          2 for i in range(2,10):
          3     for j in range(1,10):
          4         print(i*j, end=" ") # end는 해당 결과값을 출력할 때 다음줄로 넘기지 않고 그 줄에서 계속해서 출력
          5         print('')        # 2단, 3단 구분하기 위하여 다음줄 부터 출력

2 4 6 8 10 12 14 16 18
3 6 9 12 15 18 21 24 27
4 8 12 16 20 24 28 32 36
5 10 15 20 25 30 35 40 45
6 12 18 24 30 36 42 48 54
7 14 21 28 35 42 49 56 63
8 16 24 32 40 48 56 64 72
9 18 27 36 45 54 63 72 81
```

```
In [14]: 1 a = [1,2,3,4]
          2 result = []
          3 for num in a:
          4     result.append(num*3)
          5 print(result)
```

[3, 6, 9, 12]

```
In [15]: 1 result = [num * 3 for num in a]
          2 print(result)
```

[3, 6, 9, 12]

```
In [17]: 1 result = [num * 3 for num in a if num % 2 == 0] # 짝수에만 3을 곱하여..
          2 print(result)
```

[6, 12]

```
In [20]: 1 result = [x*y for x in range(2,10) # 구구단의 모든 결과를 리스트에....
          2             for y in range(1,10)]
          3 print(result)
```

[2, 4, 6, 8, 10, 12, 14, 16, 18, 3, 6, 9, 12, 15, 18, 21, 24, 27, 4, 8, 12, 16, 20, 24, 28, 32, 36, 5, 10, 15, 20, 25, 30, 35, 40, 45, 6, 12, 18, 24, 30, 36, 42, 48, 54, 7, 14, 21, 28, 35, 42, 49, 56, 63, 8, 16, 24, 32, 40, 48, 56, 64, 72, 9, 18, 27, 36, 45, 54, 63, 72, 81]

A 학급에 총 10명의 학생이 있다. 이 학생들의 중간고사 점수는 다음과 같다.

[70,60,55,75,95,90,80,80,85,100]

for문을 이용하여 A학급의 평균 점수를 구해보자

제어문

함수/사용자 입력과 출력

파일읽고쓰기


```
In [19]: 1 # 함수의 결과값(리턴값)
          2 def sum_and_mul(a,b):
          3     return a+b, a*b
```

```
In [21]: 1 result = sum_and_mul(3,4)
          2 print(result) # 튜플값(a+b, a*b) 리턴

(7, 12)
```

```
In [26]: 1 sum, mul = sum_and_mul(3,4)
          2 print("%d, %d" % (sum, mul))

7, 12
```

```
In [29]: 1 def say_nick(nick):
          2     if nick == "바보":
          3         return
          4     print("나의 별명은 %s입니다." % nick)
```

```
In [30]: 1 say_nick('야호')

나의 별명은 야호입니다.
```

```
In [31]: 1 say_nick('바보')
```

```
In [32]: 1 # 입력인수에 초깃값 미리 설정하기
2 def say_myself(name, old, man=True):
3     print("나의 이름은 %s입니다" % name)
4     print("나이는 %d입니다." % old)
5     if man:
6         print("남자입니다.")
7     else:
8         print("여자입니다.")
```

```
In [33]: 1 say_myself("박항서", 27)
```

나의 이름은 박항서입니다
나이는 27입니다.
남자입니다.

```
In [35]: 1 say_myself("박항서", 27, True)
```

나의 이름은 박항서입니다
나이는 27입니다.
남자입니다.

```
In [36]: 1 say_myself("박항서", 27, False)
```

나의 이름은 박항서입니다
나이는 27입니다.
여자입니다.

In [37]:

```
1 a = 1
2 def vartest(a):
3     a = a + 1
4
5 vartest(a)
6 print(a)
```

1

In [38]:

```
1 a = 1
2 def vartest(a):
3     a = a + 1
4     return a
5
6 vartest(a)
7 print(a)
```

1

In [40]:

```
1 a = 1
2 def vartest():
3     global a # global 명령어 사용하지 않는 것이 좋다.
4     a = a + 1
5
6 vartest()
7 print(a)
```

2

```
In [1]: # 입력되는 모든 것은 문자열로 취급  
a = input()
```

Life is too short, you need python

```
In [5]: a
```

```
Out [5]: 'Life is too short, you need python'
```

```
In [3]: number = input("숫자를 입력하세요.")
```

숫자를 입력하세요.3

```
In [6]: print(number)
```

3

```
In [1]: # 큰 따옴표(")로 둘러싸인 문자열은 +연산과 동일  
print("life" "is" "too short")
```

lifeistoo short

```
In [2]: print("life" + "is" + "too short")
```

lifeistoo short

```
In [3]: # 문자열 띄어쓰기는 콤마로 한다  
print("life", "is", "too short")
```

life is too short

```
In [4]: # 한 줄에 결과값 출력하기  
for i in range(10):  
    print(i, end='')
```

0123456789

제어문

함수/사용자 입력과 출력

파일읽고쓰기

```
In [1]: # 파일 생성
f = open("새파일.txt", 'w')
f.close()
```

```
In [3]: # 파일을 쓰기 모드로 열어 출력값 적기
f = open("C:/Temp/새파일.txt", 'w')
for i in range(1,11):
    data = "%d번째 줄입니다.\n" % i
    f.write(data)
f.close()
```

```
In [4]: for i in range(1,11):
        data = "%d번째 줄입니다.\n" % i
        print(data)
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

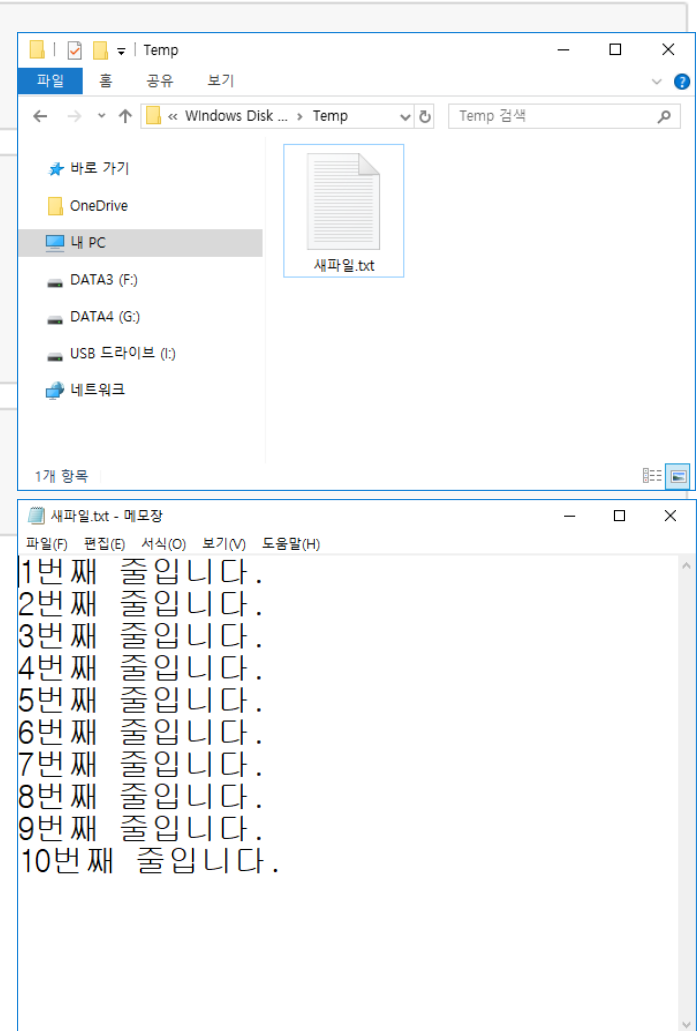
6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

9번째 줄입니다.

10번째 줄입니다.



```
In [5]: f = open("C:/Temp/새파일.txt", 'r')
        line = f.readline()
        print(line)
        f.close()
```

1번째 줄입니다.

```
In [6]: f = open("C:/Temp/새파일.txt", 'r')
        while True:
            line = f.readline()
            if not line: break
            print(line)
        f.close()
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

9번째 줄입니다.

10번째 줄입니다.


```
In [7]: # readlines() : 파일의 모든 라인을 읽어서 각각의 줄을 요소로 갖는 리스트로 리턴
f = open("C:/Temp/새파일.txt", 'r')
lines = f.readlines()
for line in lines:
    print(line)
f.close()
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

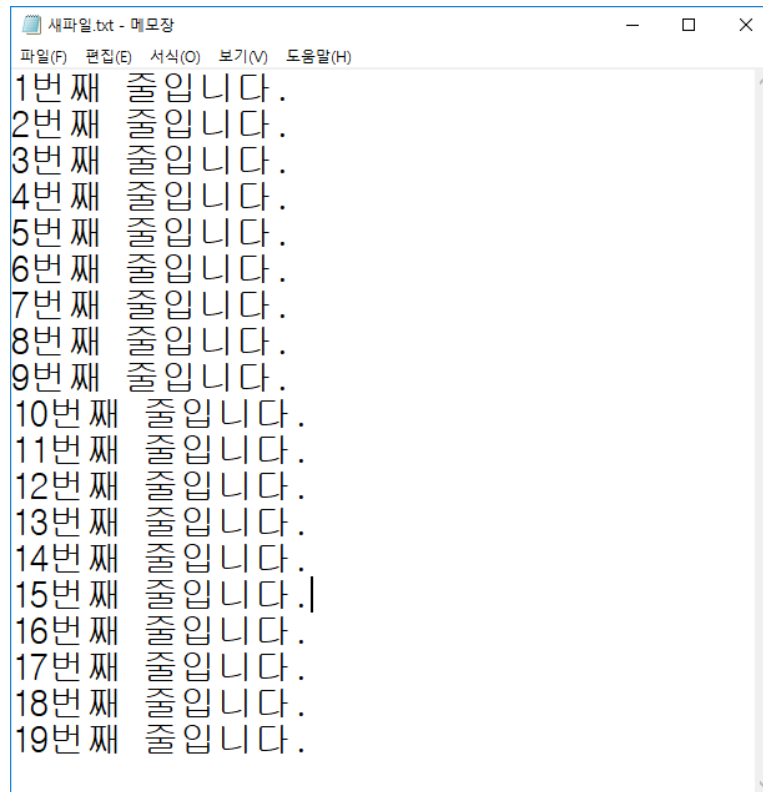
9번째 줄입니다.

10번째 줄입니다.

```
In [8]: # read() : 파일의 내용 전체를 문자열로 리턴  
f = open("C:/Temp/새파일.txt", 'r')  
data = f.read()  
print(data)  
f.close()
```

1번째 줄입니다.
2번째 줄입니다.
3번째 줄입니다.
4번째 줄입니다.
5번째 줄입니다.
6번째 줄입니다.
7번째 줄입니다.
8번째 줄입니다.
9번째 줄입니다.
10번째 줄입니다.

```
In [9]: f = open("C:/Temp/새파일.txt", 'a')
        for i in range(11,20):
            data = "%d번째 줄입니다.\n" % i
            f.write(data)
        f.close()
```



```
In [11]: f = open("foo.txt", 'w')  
f.write("Life is too short, you need python")  
f.close()
```

```
In [12]: # with문은 파일을 열고 닫는 것을 close()없이 자동으로 처리해 준다.  
with open("foo.txt", "w") as f:  
    f.write("Life is too short, you need python")
```

```
In [13]: # 입력인수를 직접 줄 수 있는 방법
# sys1.py
import sys
args = sys.argv[1:]
for i in args:
    print(i)

# 실행예제
# C:\python>python sys1.py aaa bbb ccc
# aaa
# bbb
# ccc
```

```
In [1]: # 파일 생성
f = open("새파일.txt", 'w')
f.close()
```

```
In [3]: # 파일을 쓰기 모드로 열어 출력값 적기
f = open("C:/Temp/새파일.txt", 'w')
for i in range(1,11):
    data = "%d번째 줄입니다.\n" % i
    f.write(data)
f.close()
```

```
In [4]: for i in range(1,11):
        data = "%d번째 줄입니다.\n" % i
        print(data)
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

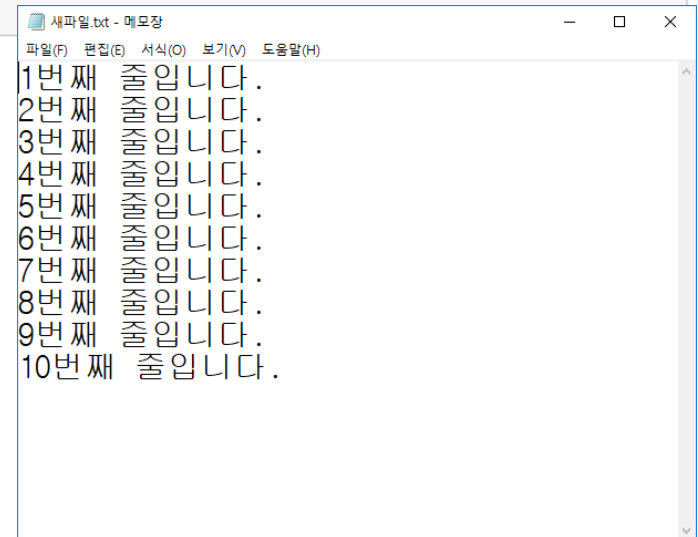
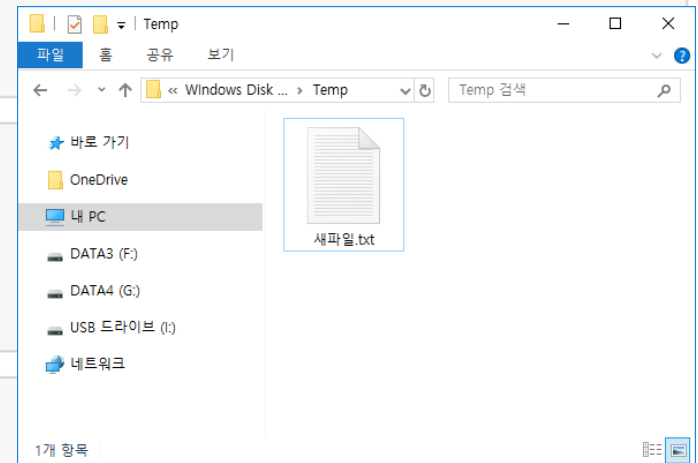
6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

9번째 줄입니다.

10번째 줄입니다.



```
In [5]: f = open("C:/Temp/새파일.txt", 'r')
        line = f.readline()
        print(line)
        f.close()
```

1번째 줄입니다.

```
In [6]: f = open("C:/Temp/새파일.txt", 'r')
        while True:
            line = f.readline()
            if not line: break
            print(line)
        f.close()
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

9번째 줄입니다.

10번째 줄입니다.

파일읽기:readlines()함수

```
In [7]: # readlines() : 파일의 모든 라인을 읽어서 각각의 줄을 요소로 갖는 리스트로 리턴
f = open("C:/Temp/새파일.txt", 'r')
lines = f.readlines()
for line in lines:
    print(line)
f.close()
```

1번째 줄입니다.

2번째 줄입니다.

3번째 줄입니다.

4번째 줄입니다.

5번째 줄입니다.

6번째 줄입니다.

7번째 줄입니다.

8번째 줄입니다.

9번째 줄입니다.

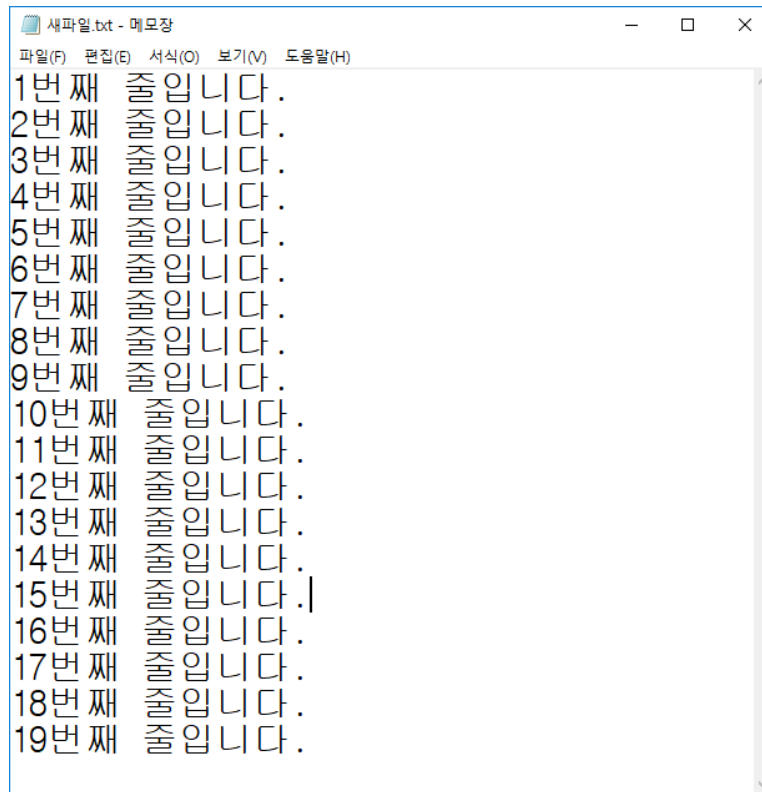
10번째 줄입니다.


```
In [8]: # read() : 파일의 내용 전체를 문자열로 리턴  
f = open("C:/Temp/새파일.txt", 'r')  
data = f.read()  
print(data)  
f.close()
```

1번째 줄입니다.
2번째 줄입니다.
3번째 줄입니다.
4번째 줄입니다.
5번째 줄입니다.
6번째 줄입니다.
7번째 줄입니다.
8번째 줄입니다.
9번째 줄입니다.
10번째 줄입니다.

파일에 내용 추가

```
In [9]: f = open("C:/Temp/새파일.txt", 'a')
        for i in range(11,20):
            data = "%d번째 줄입니다.\n" % i
            f.write(data)
        f.close()
```



```
In [11]: f = open("foo.txt", 'w')
          f.write("Life is too short, you need python")
          f.close()
```

```
In [12]: # with문은 파일을 열고 닫는 것을 close()없이 자동으로 처리해 준다.
          with open("foo.txt", "w") as f:
              f.write("Life is too short, you need python")
```

```
In [13]: # 입력인수를 직접 줄 수 있는 방법
# sys1.py
import sys
args = sys.argv[1:]
for i in args:
    print(i)

# 실행예제
# C:\#python>python sys1.py aaa bbb ccc
# aaa
# bbb
# ccc
```