

입력 / 출력

1. 입력 - input
2. 출력 - print
3. 파일 입출력
4. 자동 close - with

■ 사용자 입력 (input)

- 사용자가 입력한 값을 문자열로 저장

```
text = input()
print(text)
print(type(text))
```

- 안내 문구 출력 후 사용자 입력

```
text = input('내용을 입력해주세요. ')
print(text)
print(type(text))
```

■ 연습문제

- 두 수를 입력받은 후 덧셈 결과 출력하기

```
input1 = input('숫자1 입력: ')\ninput2 = input('숫자2 입력: ')\n\n# 코드 작성\n\nprint('덧셈 결과:', result)
```

- 아래와 같이 숫자들을 입력한 후 입력된 숫자들의 합 구하기

```
# 23, 24, 1, 9, 11, 26\ntext = input('입력: ')\ntotal = 0\n\n# 코드 작성\n\nprint('합계:', total)
```

■ 출력 (print)

● 특정 값을 화면에 출력

```
num = 123
print(num)

s = 'python'
print(s)

list = [1, 2, 'a', 'b']
print(list)

tuple = (1, 2, 'a', 'b')
print(tuple)
```

```
123
python
[1, 2, 'a', 'b']
(1, 2, 'a', 'b')
```

● space, comma, plus 연산자 사용

```
print('Life' 'is' 'too' 'short')
print('Life' + 'is' + 'too' + 'short')
print('Life ' 'is ' 'too ' 'short')
print('Life', 'is', 'too', 'short')
```

```
Lifeistooshort
Lifeistooshort
Life is too short
Life is too short
```

■ 출력 (print)

● 결과값을 한줄로 출력 (줄바꿈 X)

```
print('Life', end=' ')\nprint('is', end=' ')\nprint('too', end=' ')\nprint('short', end=' ')
```

● 반복문 결과 한줄 출력

```
for i in range(1, 11):\n    print(str(i) + '번째', end = ' ')
```

■ 연습문제

- 반복문과 print 함수를 이용하여 아래와 같은 모습 출력하기

```
n = 5
```

```
# 코드 작성
```

```
*  
**  
***  
****  
*****
```

```
      *  
     **  
    ***  
   ****  
  *****
```

■ 파일

● 생성 / 열기 : open()

- 파일객체 = open('파일명', '모드')

모드	설명
r	읽기 - 파일의 내용을 읽기만 할때 사용 (기본값)
w / x	쓰기 - 파일의 내용을 쓰기만 할때 사용
a	추가 - 파일의 마지막 부분에 내용을 추가할때 사용
+	읽기/쓰기 + 텍스트/바이너리
t	텍스트 모드 (기본값)
b	바이너리(이진) 모드

● 닫기 : close()

사용 예)

```
open('test.txt', 'rt')  
open('test.bin', 'rb')  
open('test.txt', 'wt')  
open('test.bin', 'wb')
```

■ 파일

● 읽기 : read()

- 파일의 모든 내용을 읽음

```
file = open('test.txt', 'rt', encoding='utf8')  
data = file.read()  
  
print(data)  
  
file.close()
```

1번 째

2번 째

3번 째

4번 째

5번 째

6번 째

7번 째

8번 째

■ 파일

● 읽기 : readline()

- 줄바꿈 문자까지 (한줄만) 내용 읽음

```
file = open('test.txt', 'rt', encoding='utf8')
data = file.readline()

print(data)

file.close()
```

1번째

- 내용이 읽히지 않을때까지 (마지막줄까지) 내용 읽음

```
file = open('test.txt', 'rt', encoding='utf8')
while True:
    data = file.readline()
    if not data:
        break
    print(data, end='')
file.close()
```

1번째

2번째

3번째

4번째

5번째

6번째

7번째

8번째

■ 파일

● 읽기 : readlines()

- 줄바꿈 문자를 기준으로 모든 내용을 리스트 요소로 생성

```
file = open('test.txt', 'rt', encoding='utf8')  
data = file.readlines()
```

```
print(data)
```

```
file.close()
```

```
['1번째\n', '2번째\n', '3번째\n', '4번째\n',  
'5번째\n', '6번째\n', '7번째\n', '8번째']
```

- 모든 내용을 리스트로 읽은 후 요소별 출력

```
file = open('test.txt', 'rt', encoding='utf8')  
data = file.readlines()
```

```
for line in data:  
    print(line, end='')
```

```
file.close()
```

```
1번째  
2번째  
3번째  
4번째  
5번째  
6번째  
7번째  
8번째
```

■ 연습문제

- 04_numbers_data.txt의 내용을 읽은 후 숫자들의 합 구하기

04_03_numbers_	
	-----+-----
1	7
2	9
3	24
4	29
5	34
6	38

- 04_text_data.txt의 내용을 읽은 후 알파벳 t가 포함된 단어만 출력하기

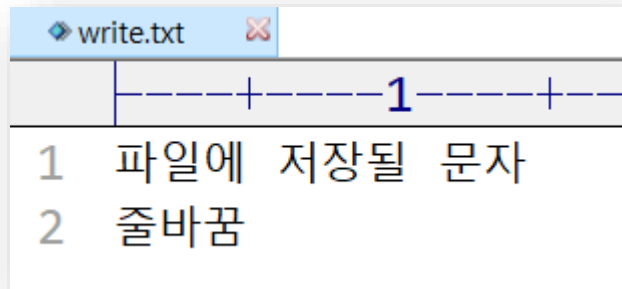
04_text_data.txt	
	-----+-----
1	python
2	java
3	c++
4	go
5	kotlin
6	fortran

■ 파일

● 쓰기 : write()

- 파일에 내용 쓰기

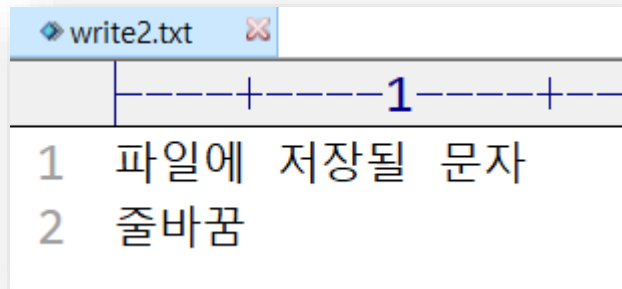
```
file = open('write.txt', 'wt', encoding='utf8')  
  
file.write('파일에 저장될 문자\n줄바꿈')  
  
file.close()
```



■ 파일

- 쓰기 : writelines()
 - 파일에 내용 쓰기

```
file = open('write2.txt', 'wt', encoding='utf8')  
  
file.writelines(['파일에 저장될 문자', '\n', '줄바꿈'])  
  
file.close()
```



■ 파일

● 자동 close

```
file = open('song.txt', 'rt', encoding='utf8')
song = file.read()

print(song)

file.close()
```



```
file = open('song.txt', 'rt', encoding='utf8')
with file:
    song = file.read()
    print(song)
```

```
with open('song.txt', 'rt', encoding='utf8') as file:
    song = file.read()
    print(song)
```