

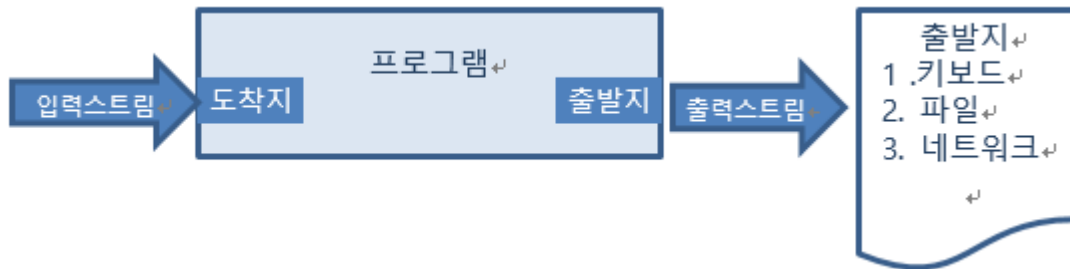
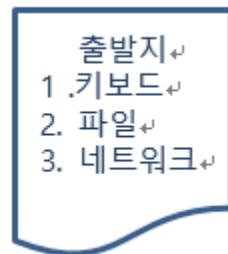
학습 내용

2부. 프로그래밍 언어 활용

6장. 모듈과 패키지

7장. 객체지향 프로그래밍

8장. 예외 처리



9장. 파일 입/출력 프로그래밍

- 1. 파일에 데이터 저장하고 불러오기
- 2. CSV 형식 파일 읽기/쓰기

파일 입출력

1절. 파일에 데이터 저장하고 불러오기

- `open()` : **파일 객체**를 반환
- `file_pointer = open(file_name, mode, encoding='ASCII')`
- 구문에서...
 - `file_pointer` : 열린 파일 객체,
파일 객체의 `readline()` 또는 `readlines()` 함수를 이용하여 파일로부터 데이터를 읽음
`write()` 함수는 데이터를 씀
 - `mode` : 파일 열기 모드를 의미
 - r : 읽기 모드 - 파일을 읽기만 할 때 사용
 - w : 쓰기 모드 - 파일의 내용을 쓸 때 사용
 - a : 추가 모드 - 파일의 마지막에 추가할 때 사용
 - b : 바이너리 모드 - 피클 등을 사용하여 저장하거나 불러올 때는 바이너리 모드로 지정해야 함
 - `encoding` : 파일의 인코딩을 지정
UTF-8 인코딩으로 저장되어 있는 파일이라면 `encoding='UTF8'`을 사용

파일 입출력

1절. 파일에 데이터 저장하고 불러오기

```
1 f = open("sample.txt", "w")
```

파일을 쓰기 모드로 연다.

```
1 print(f.writable())
```

True

```
1 f.write("Hello\nWorld\n")
```

파일에 텍스트를 쓴다.

12

```
1 f.close()
```

파일을 닫아준다.

파일 입출력

1절. 파일에 데이터 저장하고 불러오기

```
1 f = open("sample.txt", "a")
```

파일을 추가 모드로 연다.

```
1 print("프린트 함수로 쓸 수 있습니다.", file=f)
```

print() 함수의 file 인수에 파일 객체를 지정하면 파일로 저장이 가능함

```
1 f.close()
```

```
1 f = open("sample.txt", "r")
```

파일을 읽기 모드로 연다.

```
1 lines = f.readlines()
2 for line in lines:
3     print(line.strip())
```

- ❖ readlines()는 파일의 모든 라인을 한 번에 읽는다.
- ❖ strip()는 앞/뒤의 공백을 없애준다.

He l l e

W o r l d

프린트 함수로 쓸 수 있습니다.

```
1 f.close()
```

CSV 파일 읽기

2절. CSV 형식 파일 읽기

```
import pandas as pd
member = pd.read_csv('data/ch09_member1.csv', encoding='UTF-8', header=None)
member.columns = ['이름', '나이', '메일', '주소']
member
```

[18]:

	이름	나이	메일	주소
0	홍길동	20	kildong@hong.com	서울시 강동구
1	윤길동	25	yoonkil@hong.com	서울시 강서구
2	신길동	30	shinkil@hong.com	서울시 영등포구