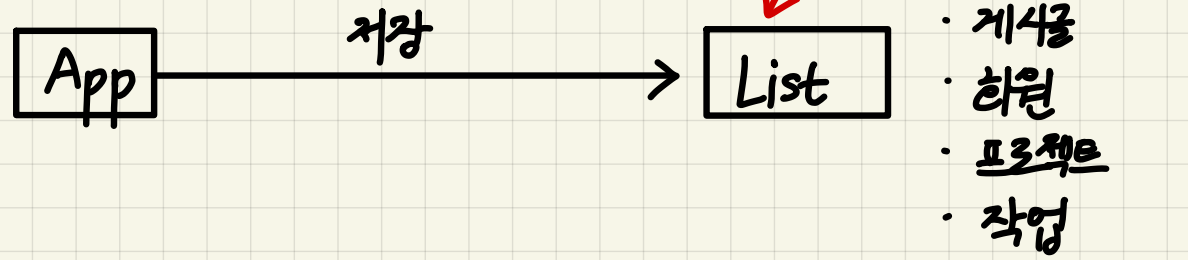


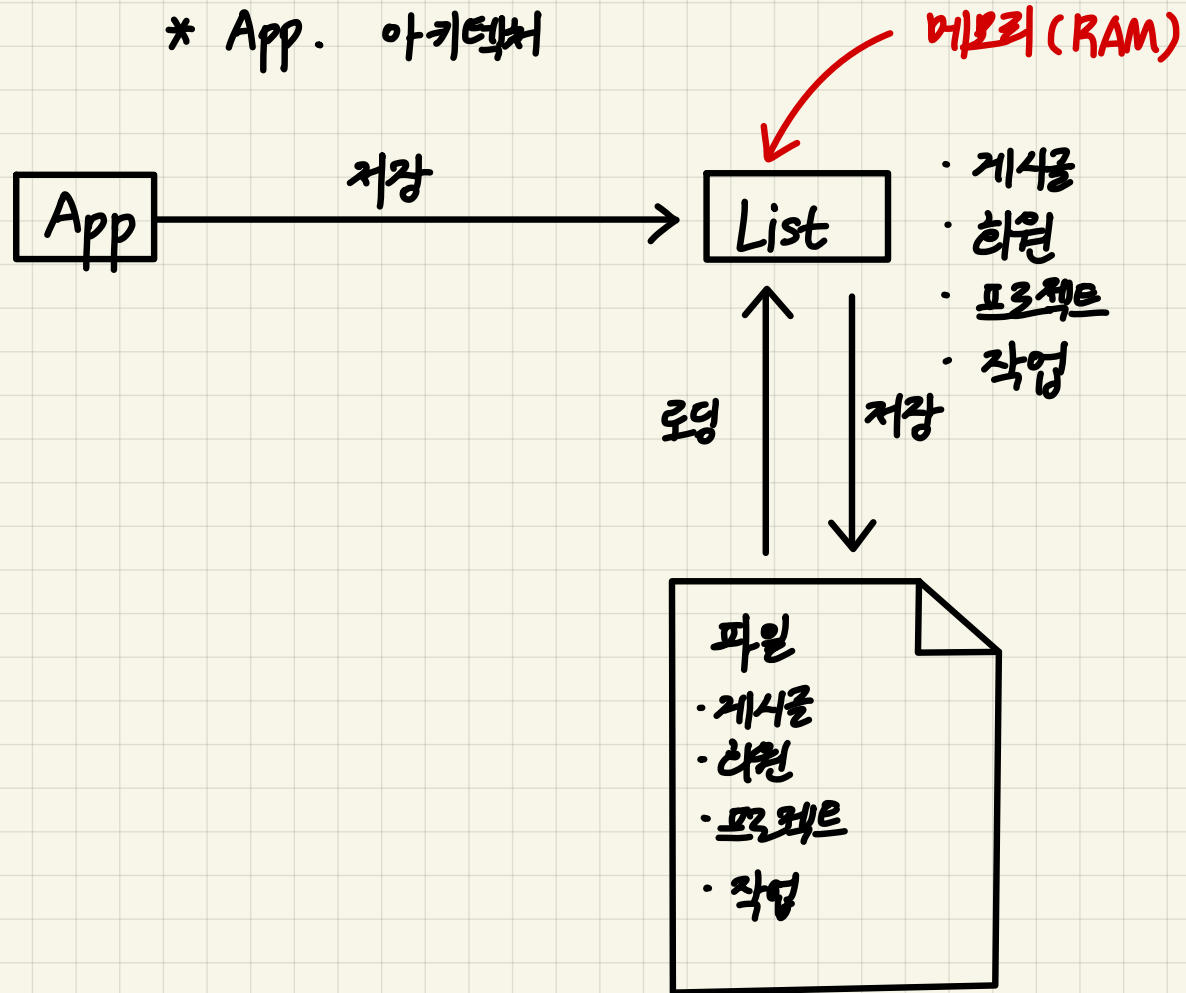

2021.9.7 (51 일화) 9:30

* App. 아키텍처



2021.9.7 (51 일화) 9:35

* App. 아키텍처



2021.9.7 (5/월) 9:40

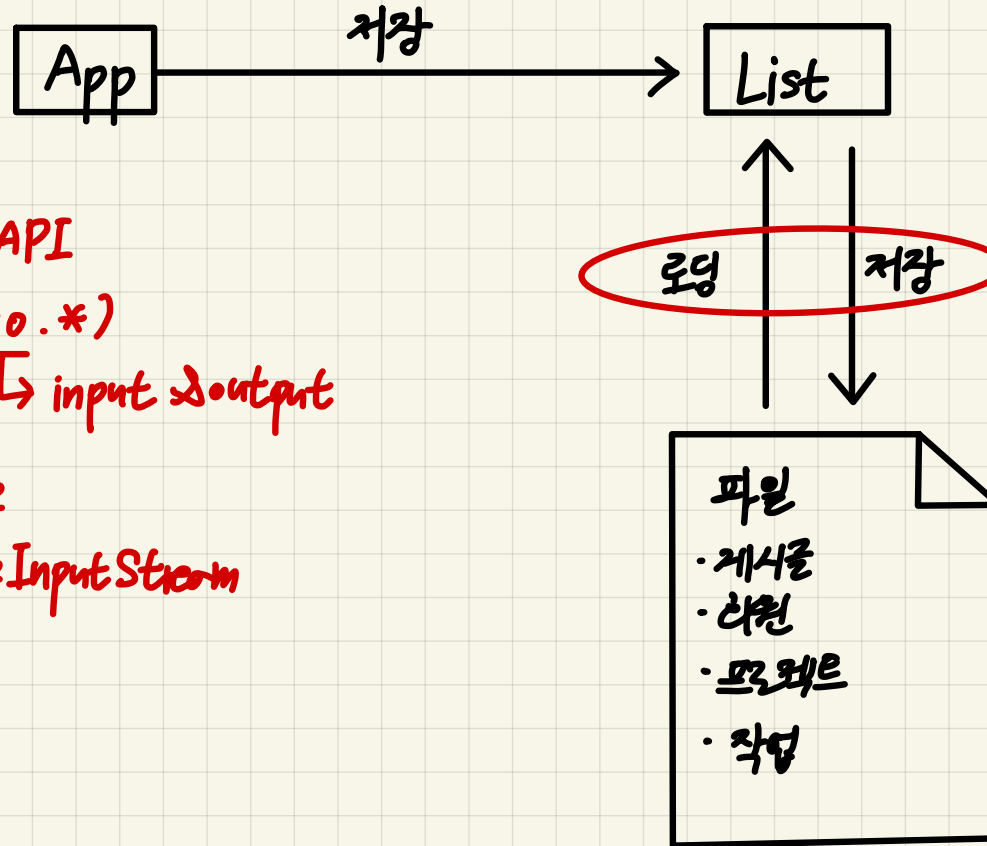
* App. 아키텍처

각자 스트림 API

(java.io.*)

↳ input & output

- File
- FileInputStream
-



2021.9.7 (5/일)

* Data I/O Stream API 주요 클래스

✓ File → (파일 디렉토리) 정보 다룰

Data, System, Car, Menu, Board, Project

✓ FileInputStream → 파일에서 데이터 읽어오는 역할

✓ FileOutputStream → 파일로 데이터 출력하는 역할

└
↓
16-a

2021.9.7 (5/월)

* 바이너리파일 VS 텍스트파일

바이너리 파일 : .pdf .doc .xls .ppt .gif .jpg .mp3 .hwp 등

✓ 사람이 직접 읽을 수 없다

✓ 전용 App을 사용해야 읽고 쓸 수 있다.

텍스트 파일 : .java .txt .md .html .css .js .c .otf

.xml docx .xlsx .pptx .properties 등

✓ 사람이 직접 읽을 수 있다

✓ 전용 App 없이 일반 텍스트 에디터로 읽고 쓸 수 있다.

2021.9.7 (5/월) 10:00

* 16-a: 바이트리 형식으로 데이터 입출력하기



{

✓ 파일에서 게시물 가져오기

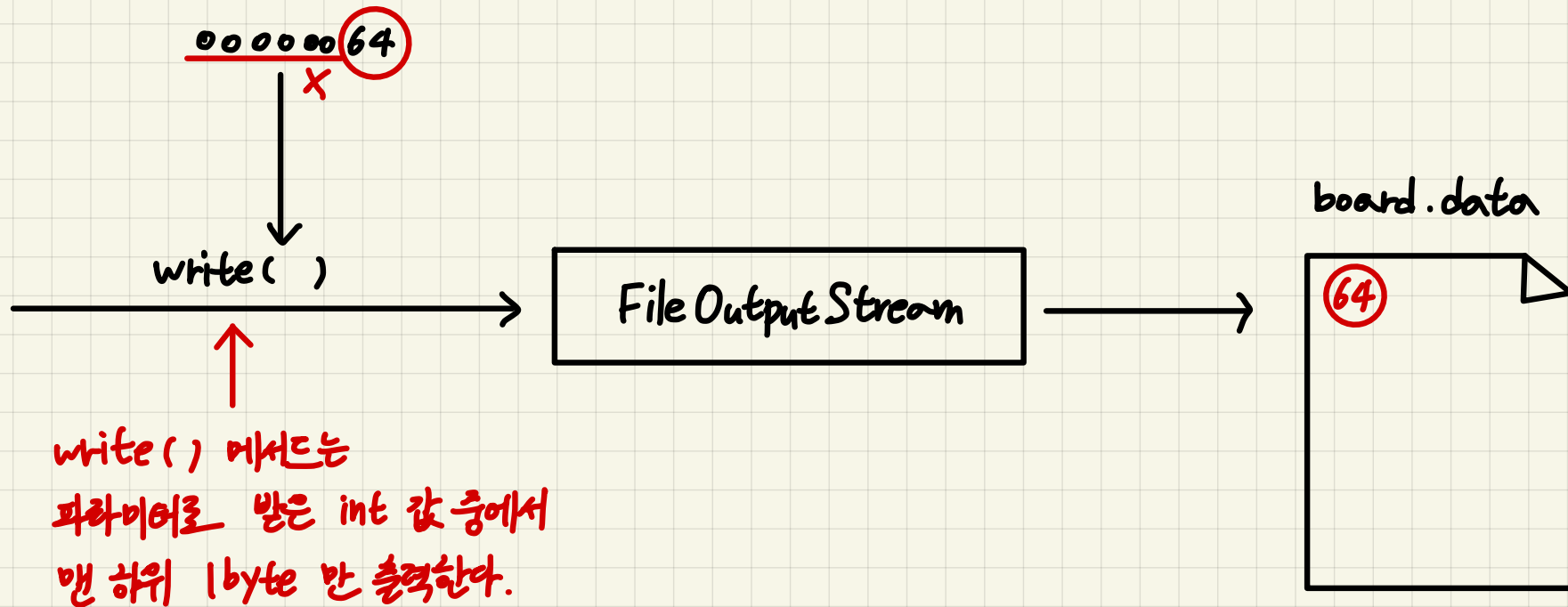
· 메뉴실행

✓

}

2021.9.7 (5/월) 10:30

* File Output Stream

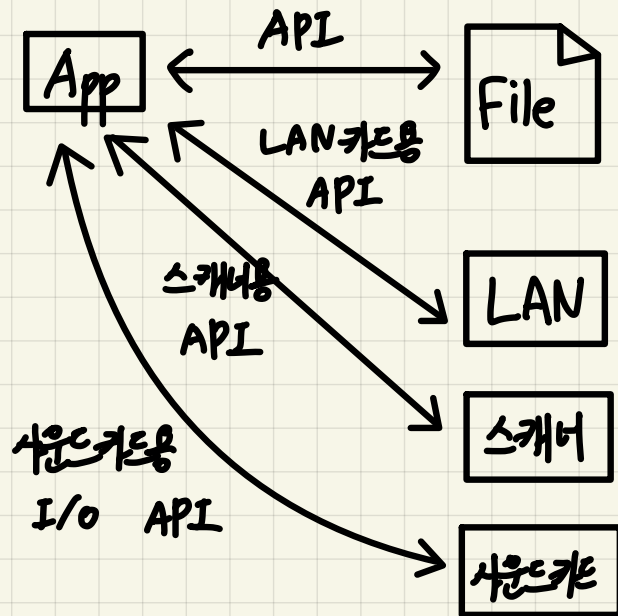


2021.9.7 (5/일) 10:40

* Streaming API 유래

↳ Data 입출력 관련 함수

① 초기



각 디바이스 (장비)에 여러
데이터 입출력을 위한 함수가 다르다.

↓↓ 문제점!

개발자는 각 장비별로
데이터 입출력 API 사용법을 학습해야 한다.

2021.9.7 (5/일) 10:40

* Streaming API 유래

(C언어에서)
등록

파일을 통해 데이터를
읽고 쓸 수 있는 함수를 정의

장비에 상관없이
데이터 입출력의 문약함을 생성

↓
장비다 연결 (mount)

(HDD I/O API)

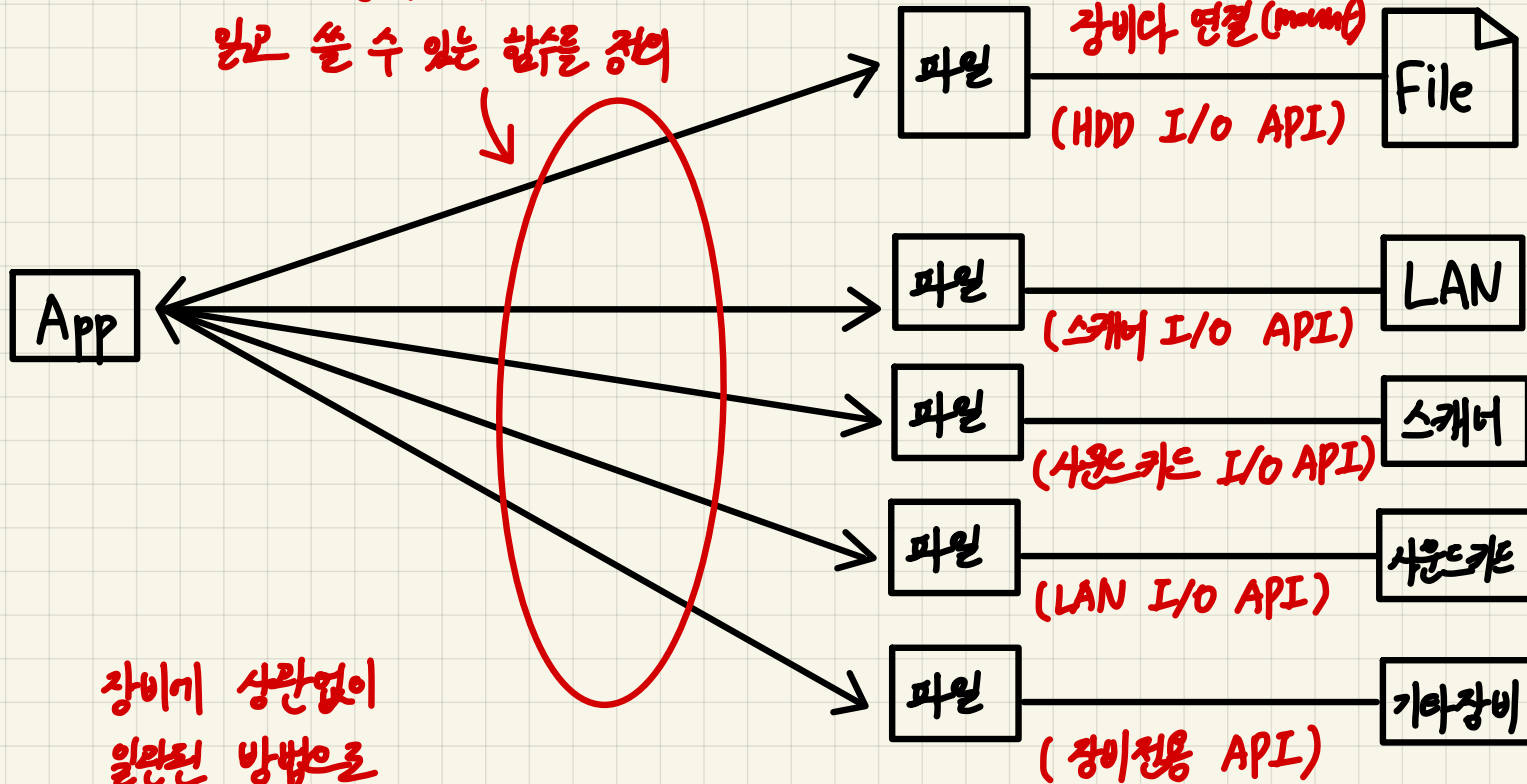
(스캐너 I/O API)

(사운드카드 I/O API)

(LAN I/O API)

(장치전용 API)

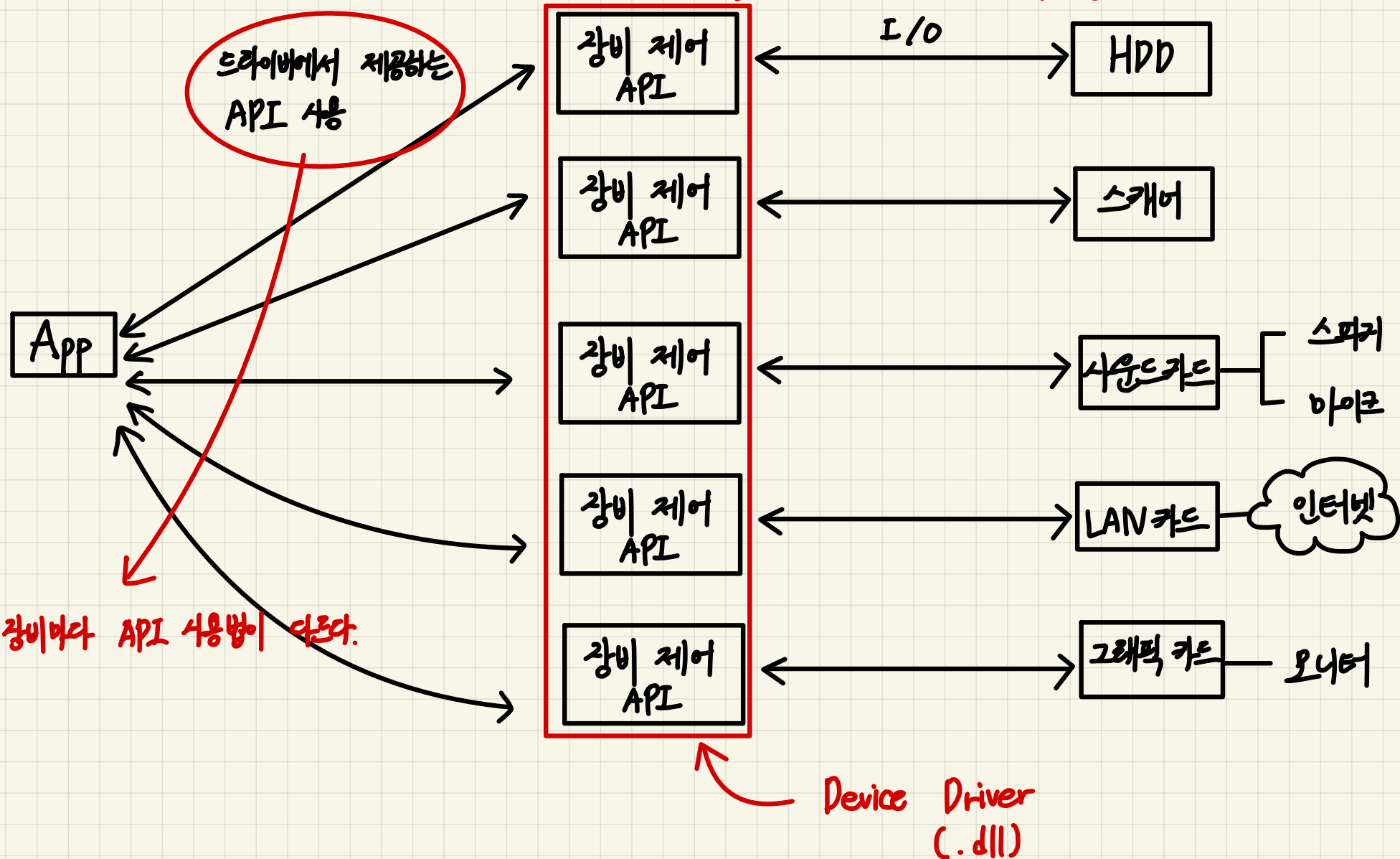
장비에 상관없이
일관된 방법으로
데이터를 읽고 쓸 수 있게 되었다.



* Streaming API 유래

장비를 제거할 때
충충한 함수가
들어있다.

H/W 제24기서 제공
↳ Device Driver



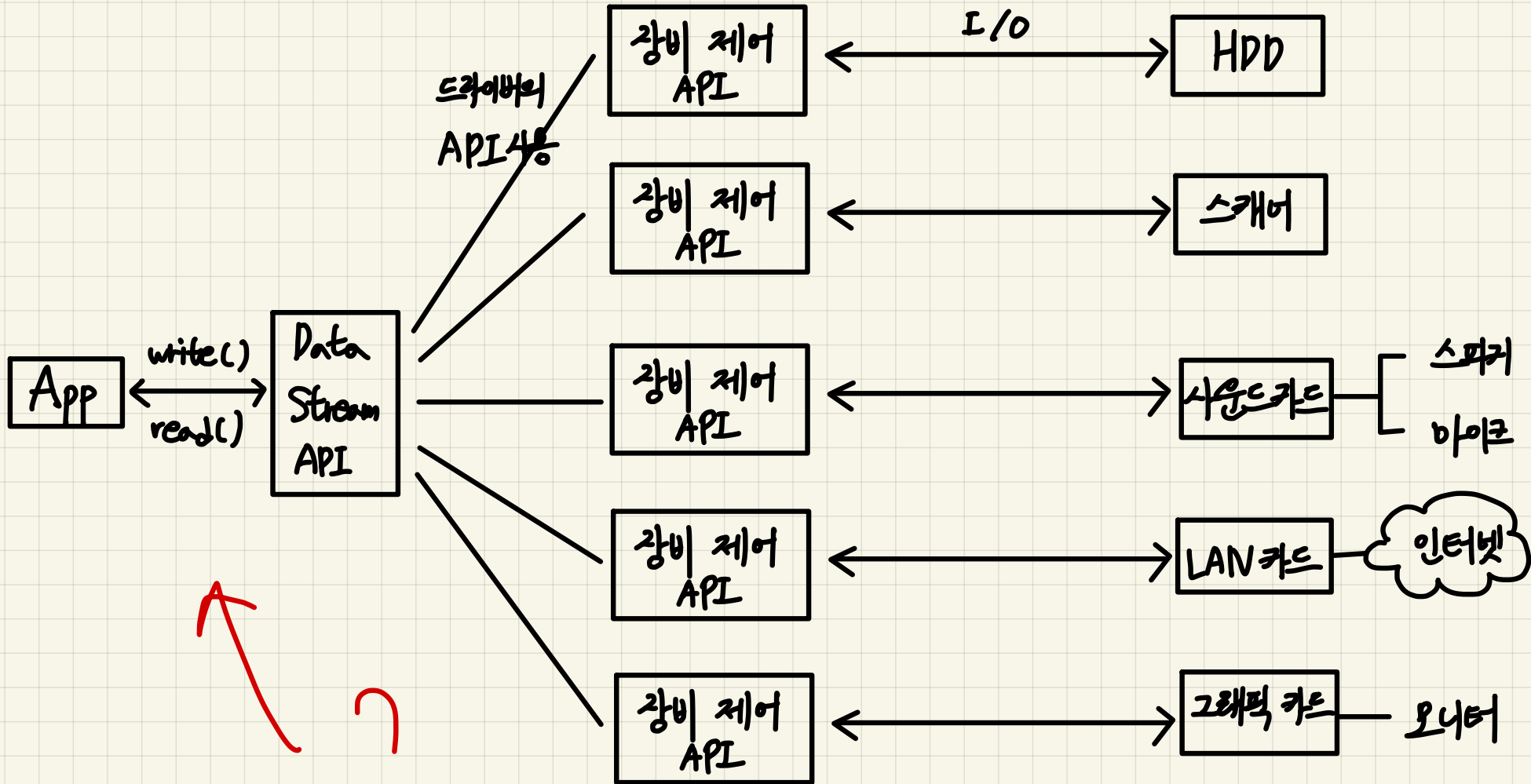
2021.9.7 (5/일) 11:10

* Streaming API 유래

↳ Data 입출력 관련 함수

장비를 제어할 때
호출한 함수가
들어있다.

H/W 제어를 제공
↳ Device Driver

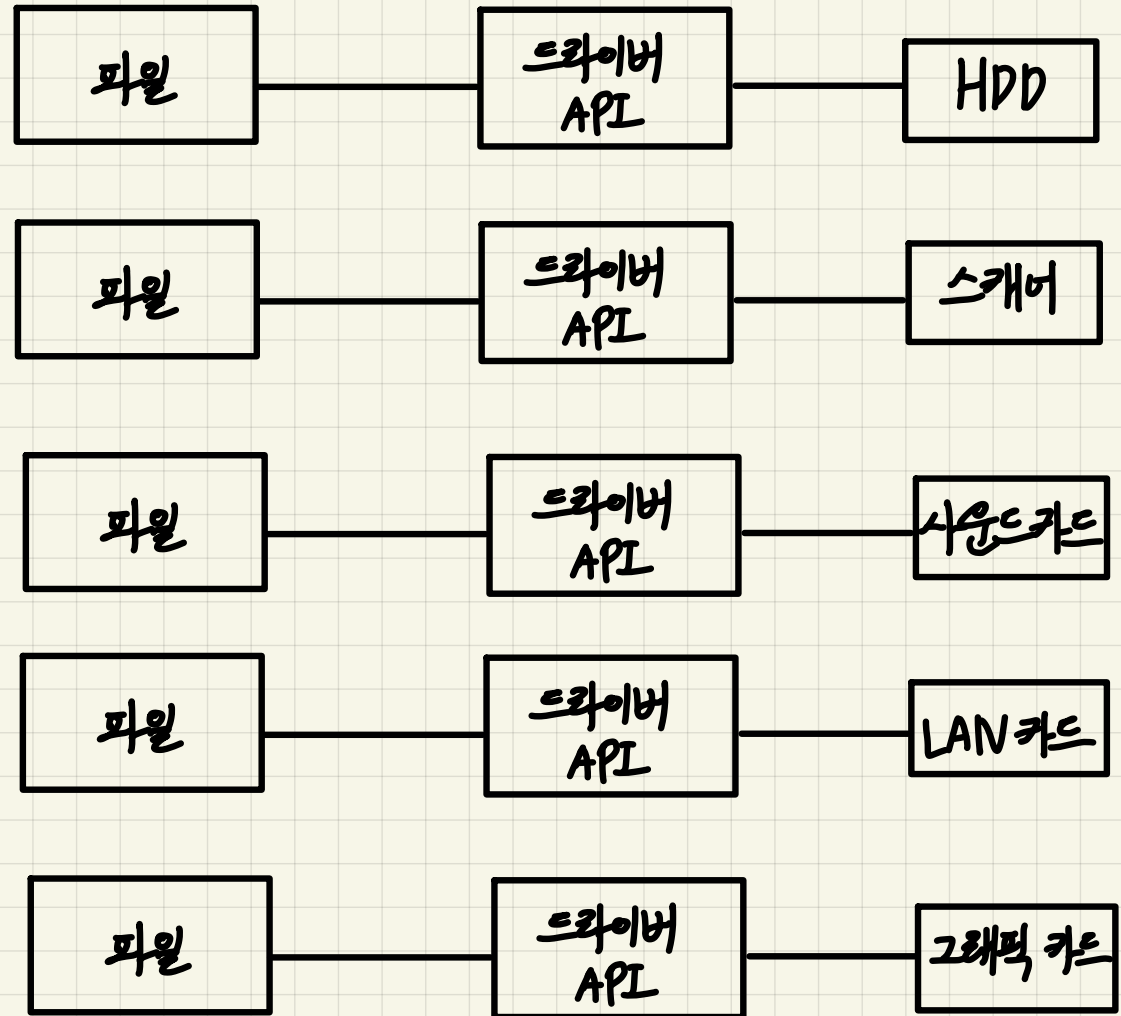
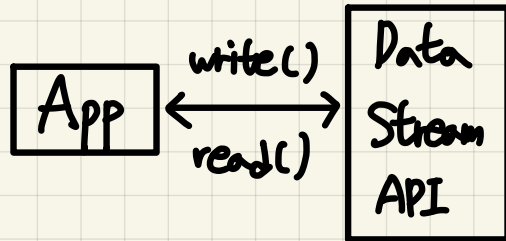


Device Driver
(.dll)

2021.9.7 (5/일) 11:20

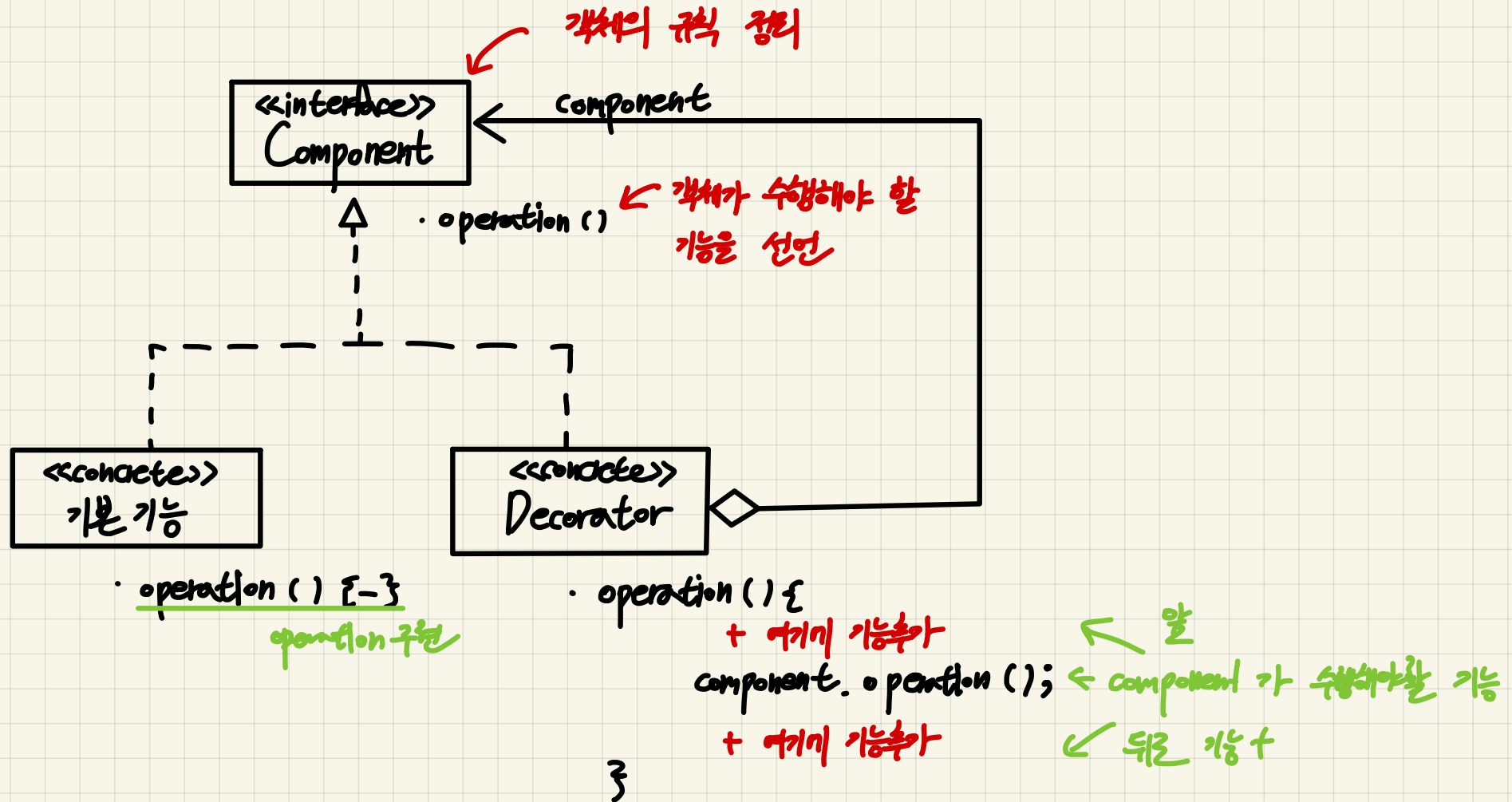
* Streaming API 유래

↳ Data 입출력 관련 함수

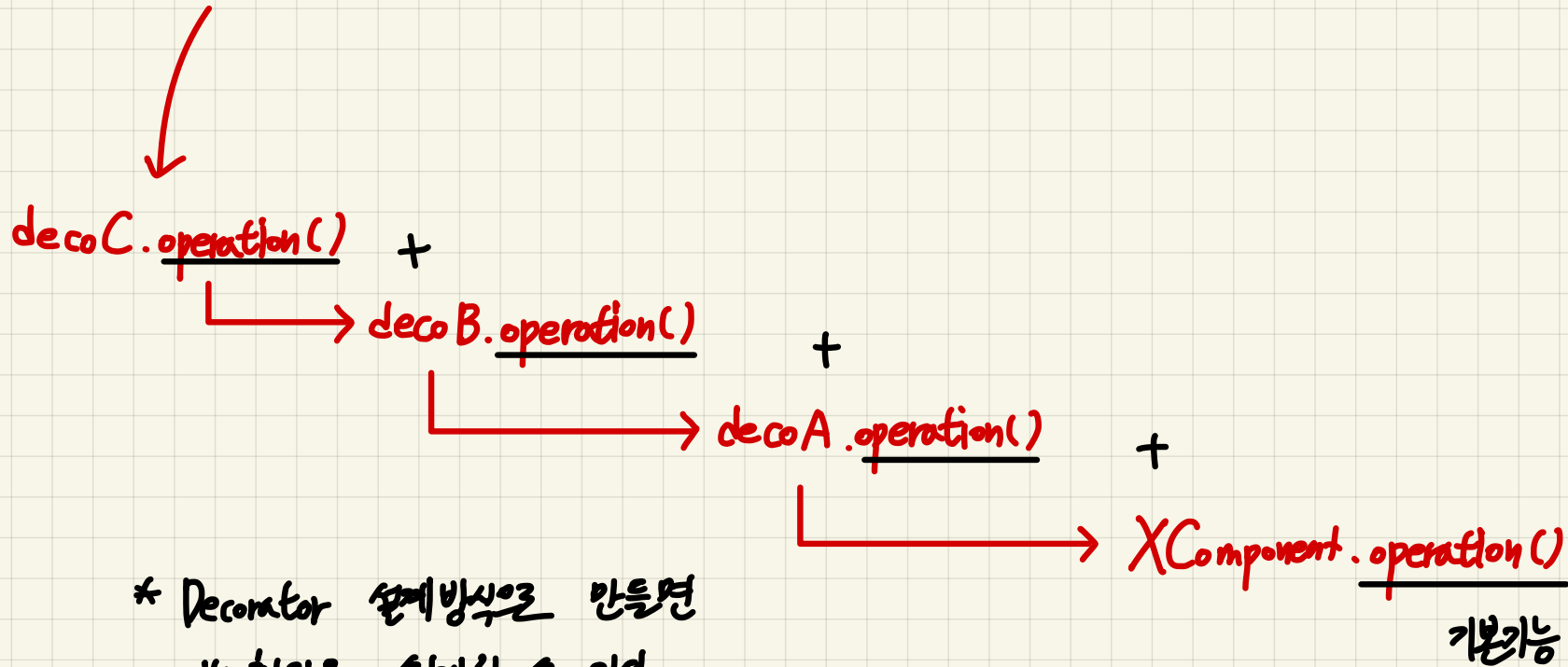


2021.9.7 (5/월) 2:50 * 데코레이터

↳ 기본 기능에 다른 기능을 덧붙이는 역할을 하는 객체



new DecoC (new DecoB (new DecoA (new X Component ())))



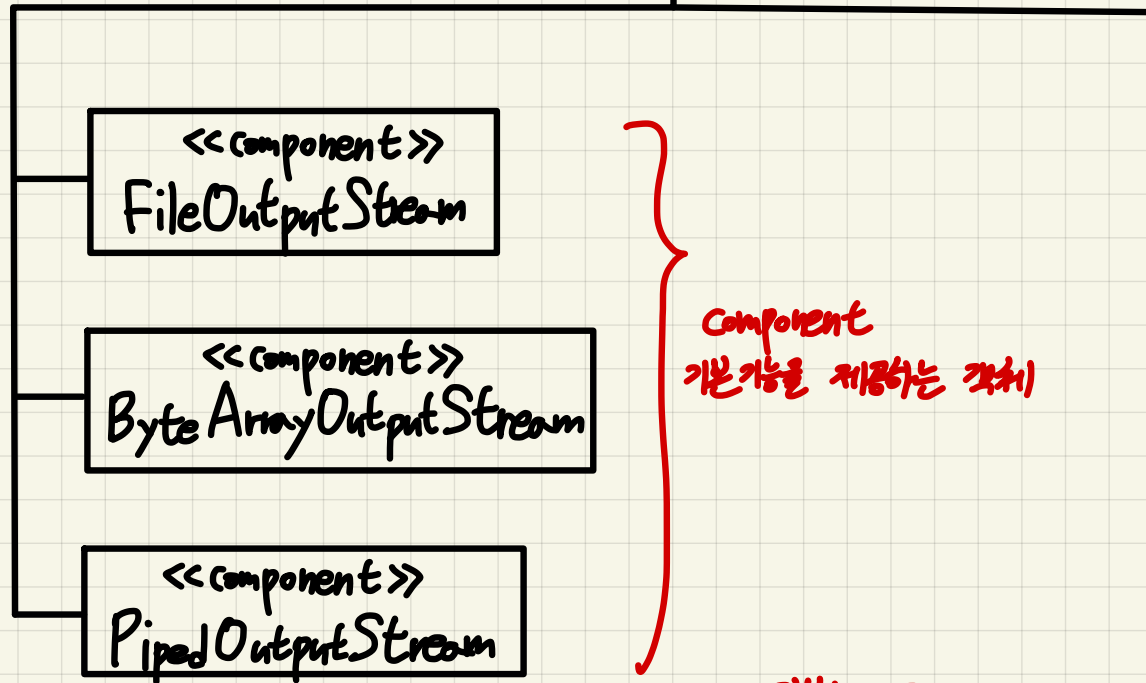
* Decorator 설계방식으로 만들면
기능 확장을 쉽게 할 수 있다.
↓

기본 기능에 새 기능을 덧붙여도 쉽다. ⇒ 각자의 Stream API는
배기도 Decorator 설계기법이 따라 만들어져 있다.

데이터 출력 일을 하는 객체가
반드시 갖춰야 하는 기능을 선언

<<abstract>>
OutputStream ← out

- write (int) ← 1byte 출력
- write (byte[]) ← 바이트 배열 출력
- write (byte[], start, end)

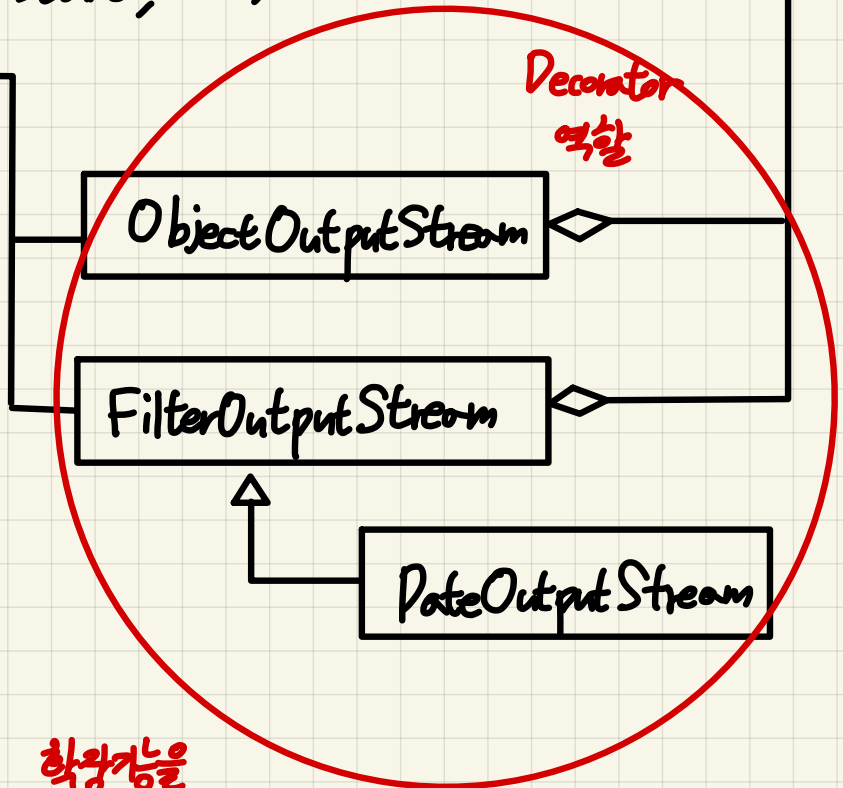


Component
기본기능을 제공하는 객체

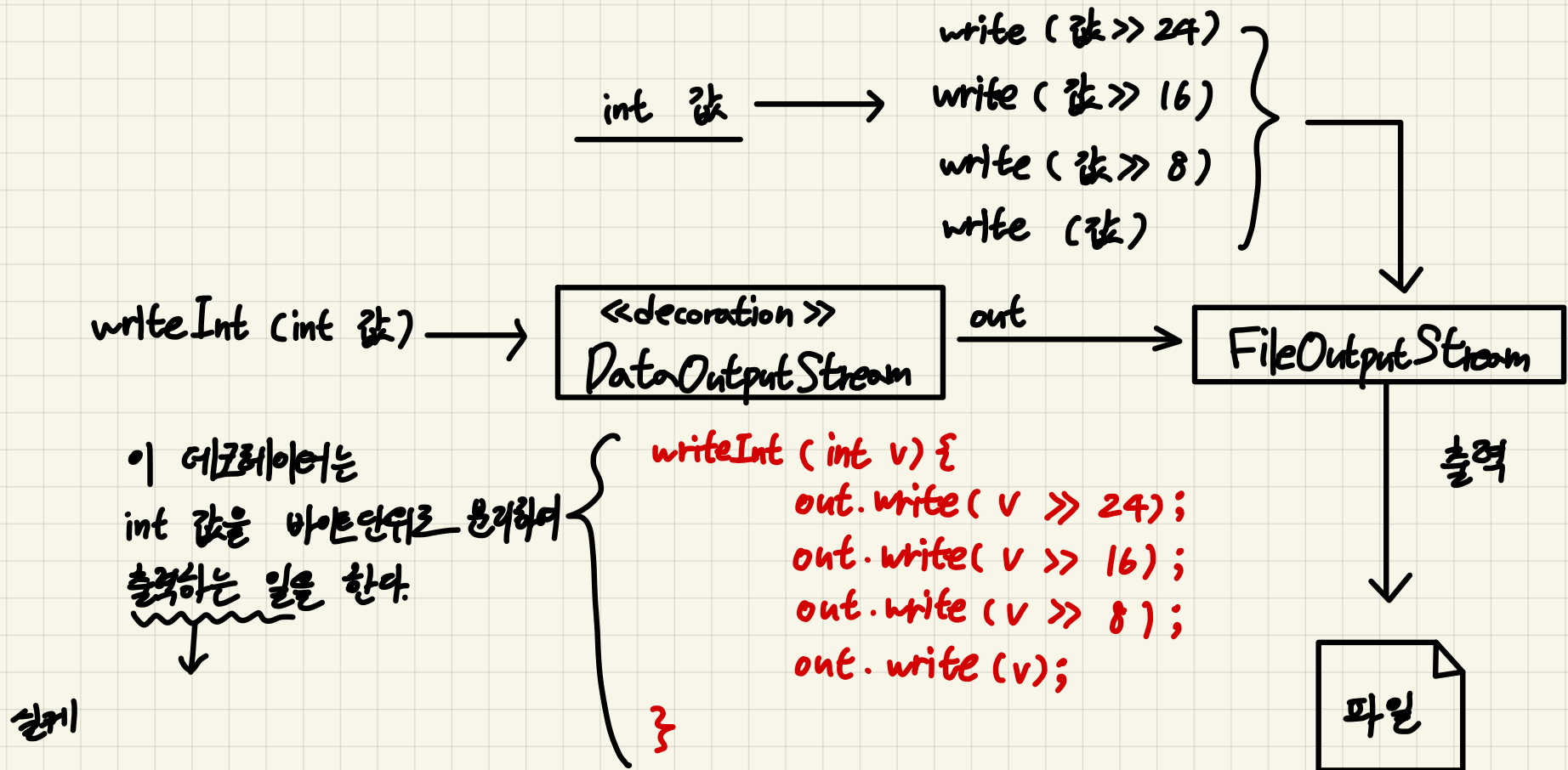
객체 스스로
기능 수행 불가!

=

확장기능을
제공하는 객체

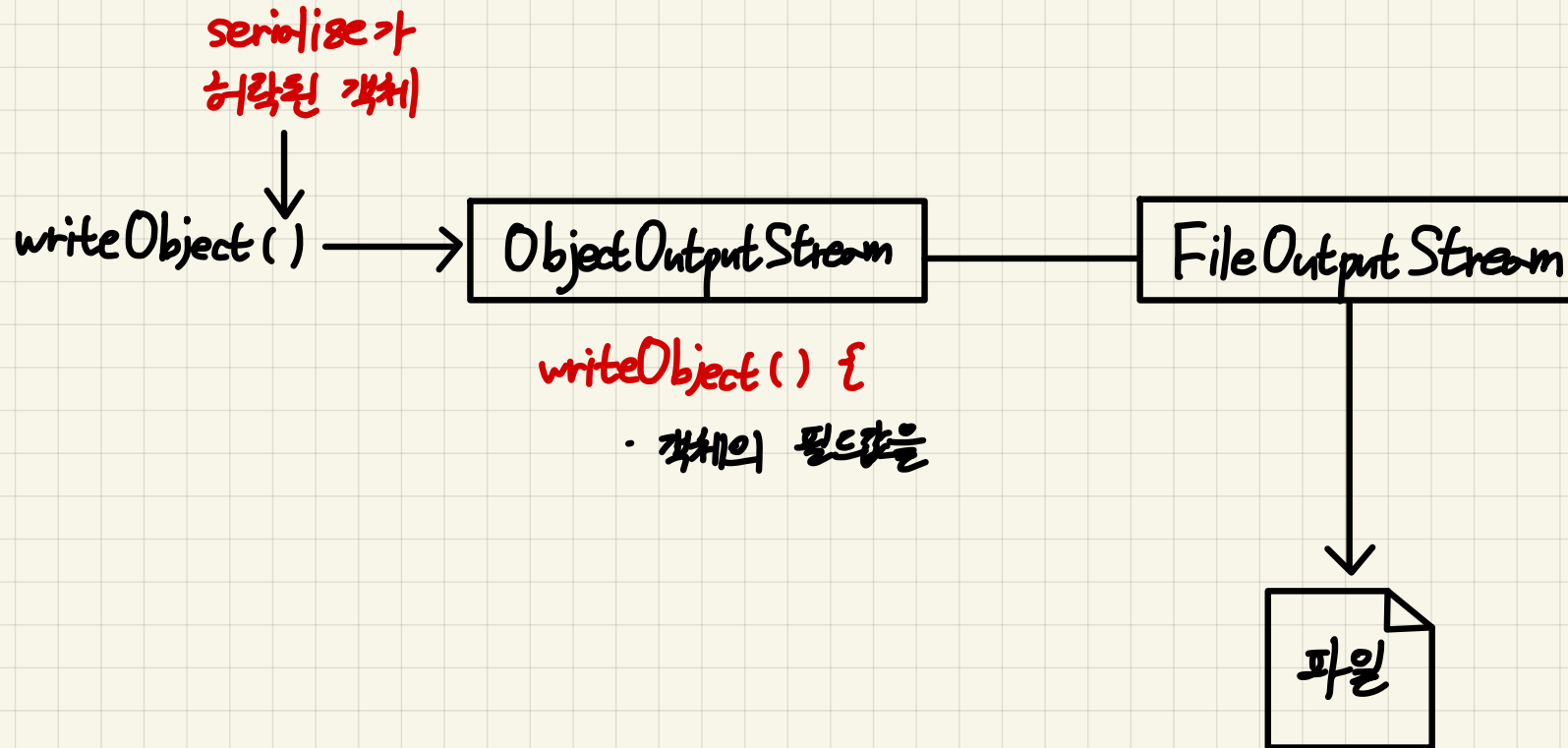


Decorator
역할



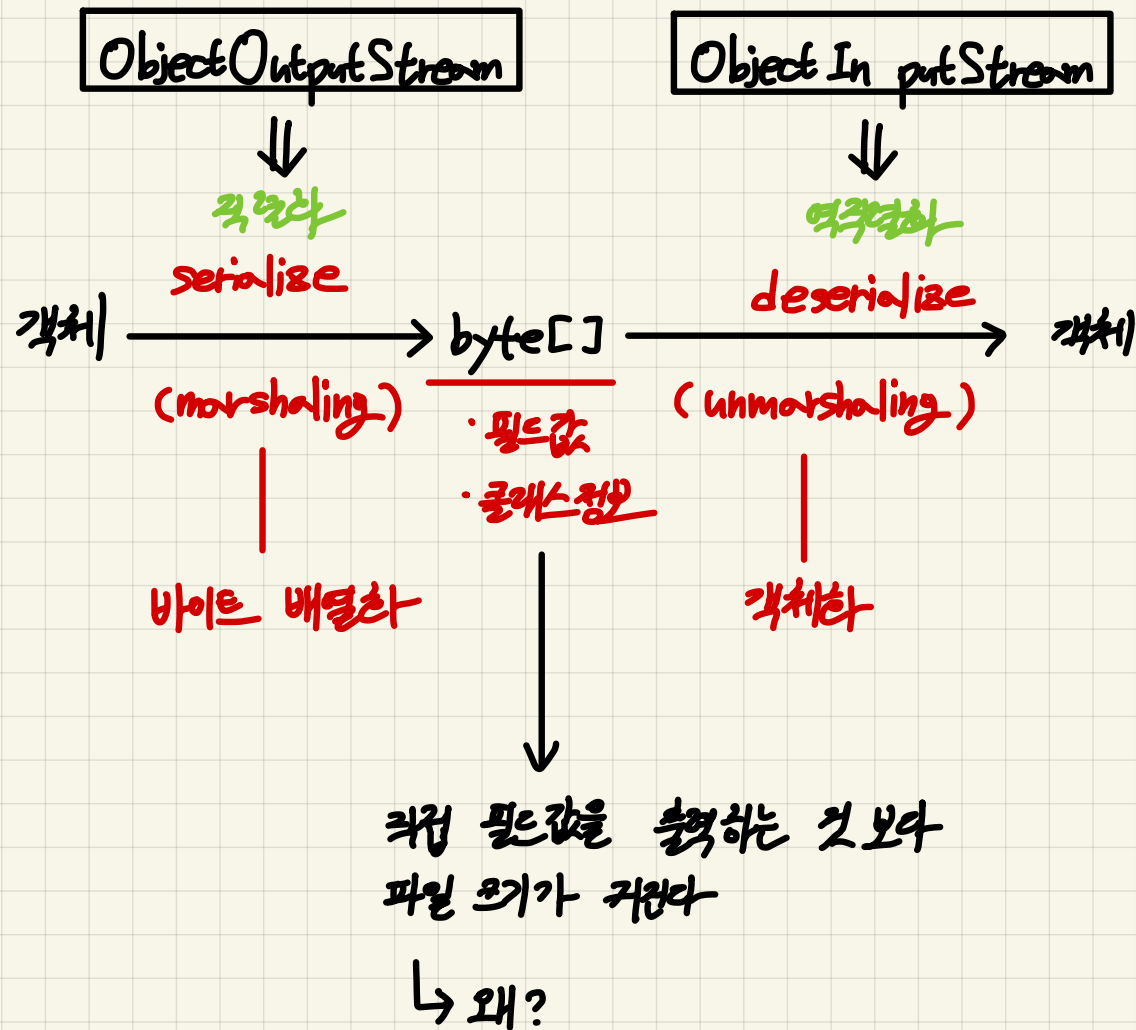
2021.9.7 (5/월) 4:25

* 객체 직렬화 시코레이터



2021.9.7 (5/일) 4:30

* 객체다 복원



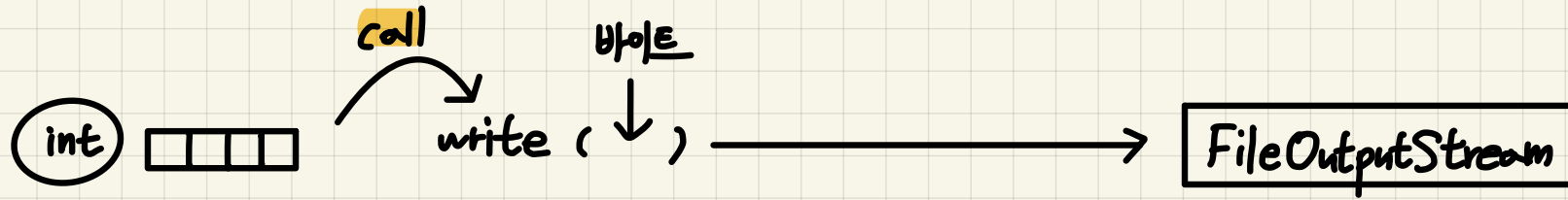
* Serialize 객체

```
class Board implements java.io.Serializable {  
    _____  
    _____  
    _____  
}
```

↑
객체의 직렬화를
허락하는 표시

4:50

*



직접화가 하려한 객체

