

[JAVA]입출력 IO & NIO

kyo kyo 2021. 1. 20. 20:51 수정 삭제

입출력이란?

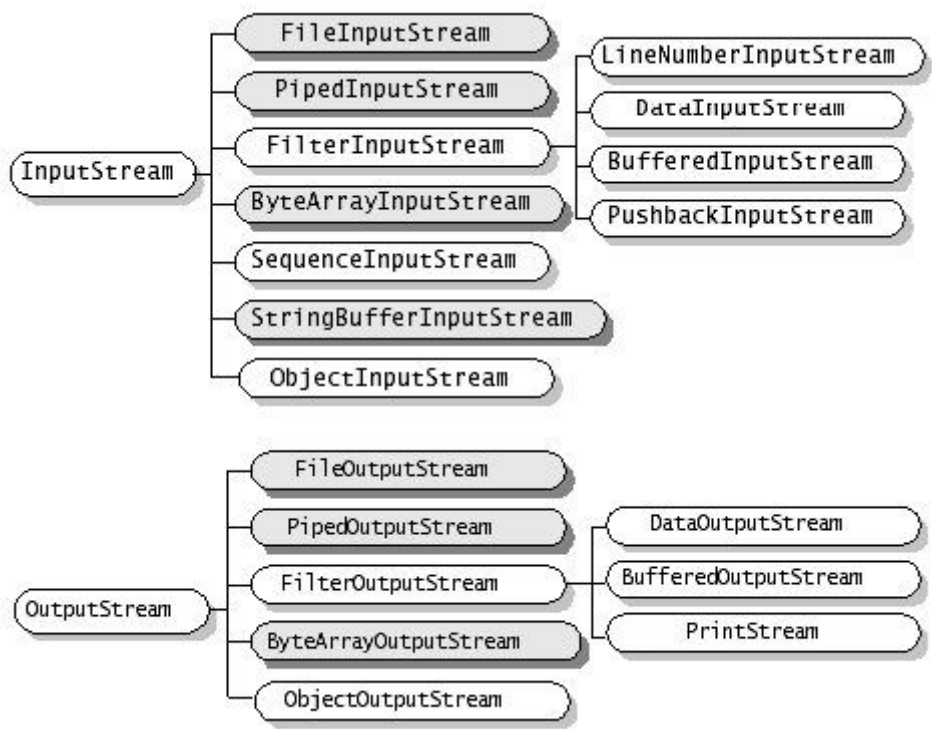
I/O란 입력과 출력, 컴퓨터 내부 또는 외부의 장치와 프로그램간의 데이터를 주고 받는 것을 말한다.

스트림(stream)

스트림이란 데이터를 운반하는데 사용되는 연결통로이다. 스트림은 단방향통신만 가능하기 때문에 하나의 스트림으로 입력과 출력을 동시에 처리할수 없다. 그래서 입력과 출력을 동시에 수행하려면 입력을 위한 입력스트림과 출력을 위한 출력 스트림 모두 2개의 스트림이 필요하다. 스트림은 먼저 보낸 데이터를 먼저 받게 되어 있으며 중간에 건너뛴 없이 연속적으로 데이터를 주고받는다. 큐와 같은 FIFO(First In First Out) 구조로 되어 있다고생각하면 이해하기 쉬울 것이다.

바이트 기반 스트림 (InputStream, OutputStream)

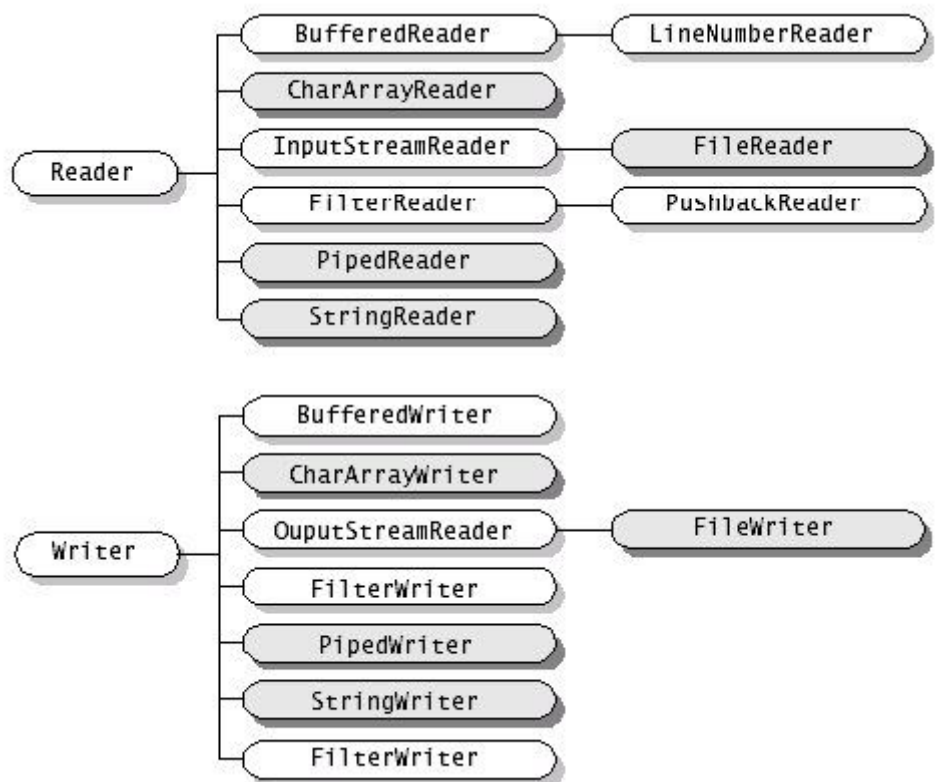
- 입출력 되는 데이터를 단순 바이트의 스트림으로 처리 1byte 단위.
- InputStream, OutputStream 클래스(추상 클래스)와 그 하위 클래스를 이용



바이트 기반 스트림 상속도

문자 기반 스트림 (Reader, Writer)

- 16비트(2Byte)로 표현되는 유니코드(nuicode), 문자 단위의 입출력을 다루는 스트림
- 한글은 2byte여서 무조건 문자 입출력 스트림만 사용 가능
- Reader, Writer 클래스(추상 클래스)와 그 하위 클래스를 이용



문자 기반 스트림 상속도

IO와 NIO의 차이점

IO

- 스트림 기반으로 입력스트림과 출력스트림이 구분되어 있어 데이터를 읽기위한 입력스트림과 데이터 출력을 위한 출력 스트림을 생성해야 한다.
- 넌버퍼로 IO는 버퍼를 제공해주는 보조 스트림인 `BufferedInputStream`, `BufferedOutputStream` 을 연결해서 사용 하기도 한다.
- 스트림에서 읽은 데이터를 즉시 처리하기 때문에 입력된 전체 데이터를 별도로 저장하지 않으면 입력된 데이터의 위치를 자유롭게 이용할 수 없다.
- 블로킹(blocking) 이 된다. 입력스트림의 `read()` 출력스트림의 `write()` 메소드를 호출하면 블로킹 된다. IO 스레드가 블 로킹되면 다른일을 할수 없고 인터럽트도 불가하다. 스트림을 닫는 방법이 블로킹을 빠져나올 수 있다.

NIO(New IO)

- 채널 기반으로 채널을 양방향으로 입출력이 가능하여 입력과 출력을 위한 별도의 채널을 만들 필요가 없다.
- 기본적으로 버퍼를 사용해서 입출력을 하므로 IO보다 성능이 좋다. 채널은 버퍼에 저장된 데이터를 출력하고 입력된 데이 터를 버퍼에 저장한다.
- 읽은 데이터를 무조건 버퍼에 저장하기 때문에 버퍼 내에서 데이터의 위치를 이동해가면서 필요한 부분만 읽고 쓸 수 있 다.
- 블로킹과 넌블로킹 특징을 모두 가지고 있다. IO 블로킹과의 차이점은 NIO 블로킹은 스레드를 인터럽트 함으로써 빠져나 올 수가 있다는 것이다. 넌블로킹은 입출력 작업 준비가 완료된 채널만 선택해서 작업 스레드가 처리하기 때문에 작업 스레 드가 블로킹 되지 않는다.

공감