

입출력(IO)





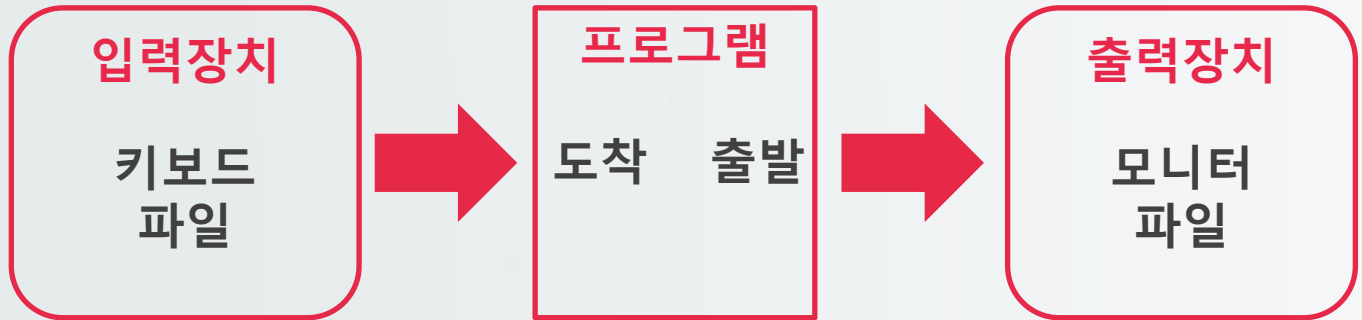
입출력(IO)



입출력

입출력

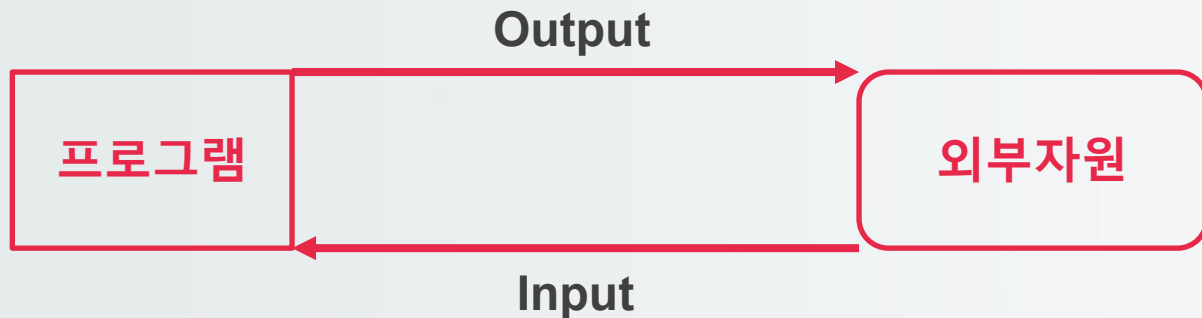
Input과 Output의 약자로, 컴퓨터 내부 또는 외부 장치와 프로그램 간의 데이터를 주고 받는 것



입출력

스트림(stream)

- 데이터를 입출력 하기 위한 통로
- 외부 자원과 입출력을 하기 위해서는 시스템으로부터 스트림을 얻어와 사용하고 난 다음 반드시 시스템에 반환
- java.io 패키지의 클래스들로 지원함
- 스트림은 **단방향**임
 - 입력과 출력이 동시에 일어나야 하는 경우 입력용 스트림과 출력용 스트림 2개가 필요함



입출력

입출력 절차

1. 외부 자원과 스트림 연결
 - 외부 자원과의 데이터 이동 통로를 생성
 - 스트림 클래스에 대한 객체를 생성
2. 데이터를 읽고 쓰기
 - 레퍼런스.읽기 메소드(); 또는 레퍼런스.쓰기 메소드() 호출
3. 데이터 읽고 쓰기가 끝나면 OS에 스트림 반납



입출력

스트림의 분류

1. 전송 방향에 따른 분류

- 입력 스트림 : 디바이스로 부터 데이터를 읽어오는 스트림
→ **InputStream, Reader**
- 출력 스트림 : 디바이스로 데이터를 출력하는 스트림
→ **OutputStream, Writer**

2. 전송 단위에 따른 분류

- 바이트 스트림 : 1byte 단위로 입력/출력하는 스트림
→ **InputStream, OutputStream**
- 문자 스트림 : 한문자(2byte)단위로 입력/출력하는 스트림
→ **Reader, Writer**

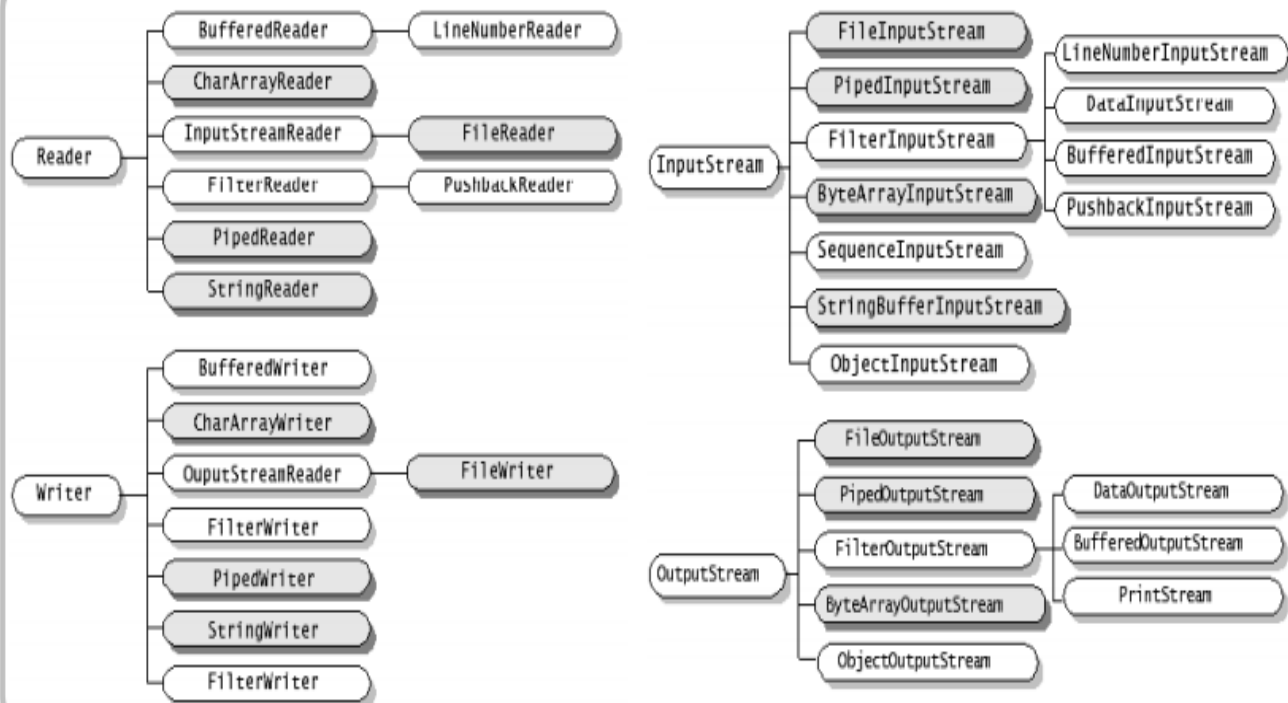
3. 보조 스트림

- 스트림의 기능을 향상시키거나 새로운 기능을 추가 시킴
- 보조 스트림만 단독적으로 사용할 수 없음



입출력

스트림의 종류





바이트 스트림



바이트 스트림

바이트 스트림

- 바이트 단위로 데이터를 전송
- 바이트 단위로 구성된 파일(동영상, 이미지, 음악 등)을 처리하기에 적합한 스트림
- 바이트 스트림의 최상위 클래스는 `InputStream`과 `OutputStream`
- 바이트 기반의 주스트림과 보조스트림이 있다.



바이트 스트림

InputStream

- 바이트 기반의 입력 스트림의 최상위 클래스로 추상클래스이다.

리턴타입	메소드	설명
int	read()	입력스트림으로부터 1바이트 읽고 읽은 바이트 리턴
int	read(byte[] b)	입력스트림으로부터 읽은 바이트들을 매 개변수로 주어진 배열에 저장하고 읽은 바이트 수 리턴
int	read(byte[] b, int off, int len)	입력스트림으로부터 len개 바이트를 읽어 b[off]부터 len개 저장하고 읽은 바이트 수 만큼 리턴
void	close()	사용한 시스템자원 반납하고 스트림 종료



바이트 스트림

OutputStream

- 바이트 기반의 출력 스트림의 최상위 클래스로 추상클래스이다.

리턴타입	메소드	설명
void	write(int b)	출력스트림으로 1바이트 전송
void	write(byte[] b)	출력 스트림에 매개변수로 주어진 배열의 모든 바이트를 전송
void	write(byte[] b, int off, int len)	출력 스트림에 매개변수로 주어진 바이트 배열의 b[off]부터 len개 바이트 전송
void	flush()	버퍼에 잔류하는 모든 바이트 출력
void	close()	사용한 시스템자원 반납하고 스트림 종료





문자 스트림



문자 스트림

문자 스트림

- 문자단위(2byte)로 데이터 전송
- 영어 이외의 문자에 대한 처리와 인코딩을 내부에서 처리
- 문자 스트림의 최상위 클래스는 Reader와 Writer
- 문자 기반의 주스트림과 보조스트림이 있다.



문자 스트림

Writer

- 문자 기반의 입력 스트림의 최상위 클래스로 추상클래스이다.

리턴타입	메소드	설명
int	read()	입력스트림으로부터 1개의 문자를 읽고 리턴
int	read(char[] b)	입력스트림으로부터 읽은 문자들을 매개 변수로 주어진 배열에 저장하고 읽은 문자 수 리턴
int	read(char[] b, int off, int len)	입력스트림으로부터 len개 문자를 읽어 b[off]부터 len개 저장하고 읽은 문자 수 만큼 리턴
void	close()	사용한 시스템자원 반납하고 스트림 종료



문자 스트림

Writer

- 문자 기반의 출력 스트림의 최상위 클래스로 추상클래스이다.

리턴타입	메소드	설명
void	write(int b)	출력스트림으로 문자 1개 전송
void	write(char[] b)	출력 스트림에 매개변수로 주어진 배열의 모든 문자 전송
void	write(char[] b, int off, int len)	출력 스트림에 매개변수로 주어진 문자 배열의 b[off]부터 len개 바이트 전송
void	write(String str)	출력스트림에 매개변수로 주어진 모든 문자열 전송
void	write(String str, int off, int len)	출력스트림에 매개변수로 주어진 모든 문자열을 off부터 len개 전송
void	flush()	버퍼에 잔류하는 모든 바이트 출력
void	close()	사용한 시스템자원 반납하고 스트림 종료



보조스트림



보조 스트림

보조 스트림

- 스트림의 기능을 향상시키거나 새로운 기능을 추가하기 위해 사용
- 보조스트림은 실제 데이터를 주고받는 스트림이 아니기 때문에 입출력을 직접 처리할 수 없고, 스트림을 먼저 생성하고 생성된 스트림을 이용해서 보조스트림을 생성



보조 스트림

보조 스트림 종류

목적	보조스트림
문자변환	InputStreamReader / OutputStreamReader
입출력 성능 향상	BufferedInputStream / BufferedOutputStream
기본데이터 타입 입출력	DataInputStream / DataOutputStream
객체 입출력	ObjectInputStream / ObjectOutputStream

사용 예)

```
FileInputStream fis = new FileInputStream("test.txt");//주스트림 생성  
BufferedInputStream bis = new BufferedInputStream(fis);//보조스트림 생성  
bis.read();//보조스트림을 통해 데이터 읽어옴
```

※ 이외에도 다양한 보조스트림이 존재함





File Class



File Class

File Class

파일 시스템의 파일을 표현하는 클래스로 파일크기, 파일속성, 파일이름 등의 정보를 제공하며 파일 생성 및 삭제기능도 제공

파일 및 디렉토리 생성 및 삭제 메소드

리턴타입	메소드	설명
boolean	createNewFile()	새로운 파일 생성
boolean	mkdir()	새로운 디렉토리 생성
boolean	mkdirs()	경로상에 없는 모든 디렉토리 생성
boolean	delete	파일 또는 디렉토리 삭제



File Class

파일 및 디렉토리 정보 리턴 메소드 - 1

리턴타입	메소드	설명
boolean	canExcute()	실행 할 수 있는 파일인지 여부
boolean	canRead()	읽을 수 있는 파일인지 여부
boolean	canWrite()	수정 및 저장할 수 있는 파일인지 여부
String	getName()	파일 이름 리턴
String	getParent()	부모 디렉토리 리턴
File	getParentFile()	부모 디렉토리를 파일객체형태로 리턴
String	getPath()	전체경로 리턴
boolean	isDirectory()	디렉토리인지 여부
boolean	isFile()	파일인지 여부
boolean	isHidden()	숨김파일인지 여부



File Class

파일 및 디렉토리 정보 리턴 메소드 - 2

리턴타입	메소드	설명
long	lastModified()	마지막 수정 날짜 및 시간 리턴
long	length()	파일의 크기 리턴
String[]	list()	디렉토리에 포함된 파일 및 서브디렉토리에 있는 파일 이름을 모두 배열로 리턴
String[]	list(FilenameFilter filter)	디렉토리에 포함된 파일 및 서브디렉토리 목록 중에 FilenameFilter에 맞는 것만 배열로 리턴
File[]	listFile()	디렉토리에 포함된 파일 및 서브 디렉토리 목록 전부를 File배열로 리턴
File[]	listFile(FilenameFilter filter)	디렉토리에 포함된 파일 및 서브 디렉토리 목록 중에 FilenameFilter에 맞는 것만 File배열로 리턴





직렬화



직렬화

직렬화(Serializable)

- 객체를 스트림으로 전송하기 위해서 진행해야 하는 작업
- 객체는 큰 덩어리 이므로 바이트단위로 잘라주어야 함
- `java.io.Serializable`을 implements하여 구현

serialVersionUID

- 직렬화시에 사용되는 객체의 고유번호
- 명시하지 않아도 `Serializable` 인터페이스를 implements하면 JVM이 임의의 번호를 붙여 생성하지만 직접 생성하는 것을 권장

ex)

```
private static final long serialVersionUID=392839429392L;
```



직렬화

transient

- 객체 직렬화 시 제외할 필드 앞에 붙여 해당필드의 직렬화를 예외 하는 키워드

역직렬화

- 직렬화의 반대 과정으로 직렬화 된 데이터를 다시 객체로 합치는 과정
- 클래스의 이름이 같더라도 클래스 내용이 변경된 경우 실패

