



JAVA 프로그래밍

Chapter 17 입출력 프로그래밍

목차

- ❖ JDK라이브러리의 파일관련 클래스들
- ❖ 파일 관리에 사용되는 File 클래스
- ❖ 파일의 내용을 읽고 쓰는 클래스들
- ❖ 입출력 기능과 성능을 향상시키는 클래스들
- ❖ 데이터를 포맷해서 출력하는 클래스들

파일 관리에 사용되는 File 클래스

- ❖ 시스템에 있는 파일이나 디렉토리를 추상화한 클래스이다.
- ❖ 파일의 생성, 삭제, 크기, 읽기 쓰기 모드 등과 같이 파일 자체를 관리하는 클래스.

[표 14-1] File 클래스의 주요 생성자

생성자	설명
File(String pathname)	문자열 pathname을 가지고 경로를 생성하여 File 객체를 생성한다.
File(String parent, String child)	Parent와 child 문자열을 연결한 문자열로 경로를 생성하여 File 객체를 생성한다.
File(File parent, String child)	Parent의 파일 객체와 child 문자열로 경로를 생성하여 File 객체를 생성한다.

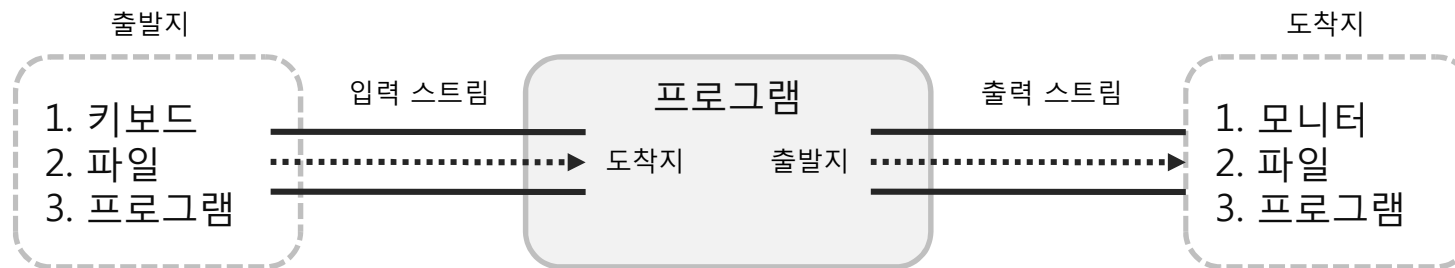
File 클래스 주요 메소드

[표 14-2] File 클래스의 주요 메서드

반환형	메서드	설명
boolean	canRead()	파일을 읽을 수 있으면 true, 그렇지 않으면 false다.
	canWrite()	파일을 쓸 수 있으면 true, 그렇지 않으면 false다.
	createNewFile()	파일을 새로 생성하면 true, 그렇지 않으면 false다.
	delete()	파일을 지우면 true, 그렇지 않으면 false다.
	exists()	파일이나 디렉토리가 존재하면 true, 그렇지 않으면 false다.
String	getAbsolutePath()	파일의 절대 경로를 반환한다.
	getCanonicalPath()	파일의 정규 경로를 반환한다.
	getName()	파일명을 반환한다.
boolean	isDirectory()	디렉토리면 true, 그렇지 않으면 false다.
	isFile()	파일이면 true, 그렇지 않으면 false다.
long	lastModified()	1970년 1월 1일부터 현재까지의 시간을 밀리세컨드 초로 반환한다.
	length()	파일의 크기를 바이트로 반환한다.
String[]	list()	특정 디렉토리의 모든 파일과 자식 디렉토리를 스트링 배열로 반환한다.
boolean	mkdir()	디렉토리를 생성하면 true, 디렉토리가 있어서 생성하지 못하면 false다.
	renameTo(File dest)	dest 파일 객체로 이름을 바꾸면 true, 그렇지 않으면 false다.

자바 입출력

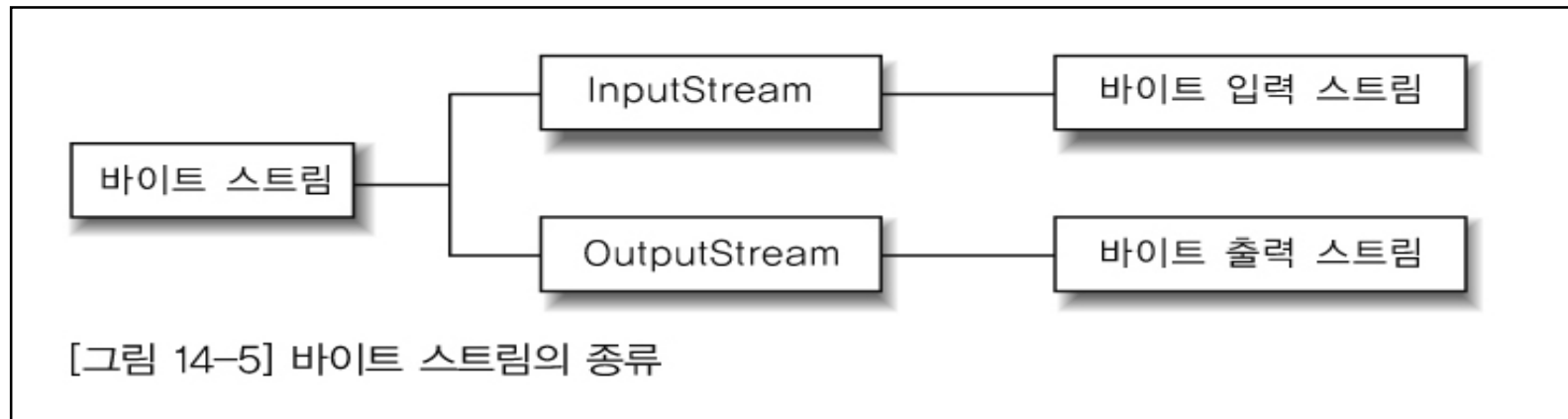
- ❖ 자바에서는 입 출력시 데이터를 읽고 쓸 때 스트림 형태로 읽고 쓴다.
- ❖ 스트림(Stream)
 - 일차원적인 데이터의 흐름.
 - 데이터를 목적지로 입출력 하기 위한 방법이다.
 - 스트림에 데이터를 쓸 수 있고, 스트림에서 데이터를 읽을 수 있다.
 - 스트림에 데이터를 쓸 경우, 이러한 스트림을 출력 스트림(output stream)이라고 한다.
 - 스트림에서 데이터를 읽을 경우, 이러한 스트림을 입력 스트림(input stream)이라고 한다



스트림의 종류

❖ 바이트 스트림

- 바이트 스트림은 1 byte를 입출력 할 수 있는 스트림이다.
- 일반적으로 바이트로 구성된 파일, 즉 동영상 파일, 이미지 파일, 음악 파일을 처리하기에 적합한 스트림이다.
- InputStream는 입력용, OutputStream는 출력용 바이트 스트림이다.
- 'InputStream'나 'OutputStream'이라는 단어가 붙어 있다면 바이트 스트림이다.
- PrintStream은 출력용만 존재하기 때문에 약간 특이하다.



스트림의 종류 - InputStream

- ❖ InputStream은 바이트 입력을 수행하는 데 필요한 메서드를 정의하는 추상 클래스이다.
- ❖ 자바 프로그램은 객체를 생성하고 생성된 객체와 바이트 스트림과 연결함으로써 파일을 연다.
- ❖ 자바는 다른 장치들과도 바이트 스트림을 연결할 수 있고 프로그램이 시작되면 장치들과 연결된 세 개의 객체(System.in, System.out, System.err)를 생성한다.
- ❖ System.in 객체는 키보드로 바이트를 입력할 수 있는 InputStream 객체이다.

스트림의 종류 - InputStream

[표 14-4] InputStream의 주요 메서드

반환형	메서드	설명
abstract int	read()	스트림 데이터 1바이트를 읽어온다. 반환값은 0~255의 아스키코드값이기 때문에 문자로 나타내려면 char로 캐스팅해야 한다. 더 이상 읽을 수 없을 때는 -1을 반환한다.
int	read(byte b[])	스트림 데이터 1바이트를 읽어 바이트 배열에 저장하고, 읽은 수만큼 반환한다.
	read(byte b[], int start, int length)	스트림 데이터를 length만큼 읽어 바이트 배열 b의 start 위치에 저장하고, 읽을 수만큼 반환한다.
	available()	읽을 수 있는 바이트 수를 반환한다.
long	skip(long n)	읽을 수 있는 바이트에서 n만큼 건너뛴다.
void	close()	입력 스트림을 닫는다.

스트림의 종류 - OutputStream

- ❖ OutputStream은 바이트 출력을 수행하는 필요한 메서드를 정의한 추상 클래스이다.
- ❖ 프로그램이 시작 되면 장치와 연결된 두 개의 출력 스트림은 System.out, System.err를 생성한다.
- ❖ System.out 객체는 화면에 데이터를 출력 한다.
- ❖ System.err 객체는 화면에 오류 메시지를 출력하게 된다.

[표 14-8] OutputStream의 주요 메서드

반환형	메서드	설명
abstract void	write(int b)	출력 스트림으로 b의 값을 바이트로 변환하여 쓰기한다.
void	write(byte[] b)	출력 스트림으로 바이트 배열 b를 쓰기한다.
	write(byte[] b, int start, int length)	출력 스트림으로 바이트 배열 b를 start부터 length만큼 쓰기한다.
	flush()	출력 스트림을 통하여 쓰기를 할 때 일반적으로 버퍼에 가득차게 되면 한꺼번에 보내게 되는데, 이 메서드를 사용하게 되면 버퍼에 가득 차 있지 않더라도 버퍼의 내용을 바로 보내게 된다.
	close()	모든 자원을 반납한다.

스트림의 종류

❖ 문자 스트림 패턴

- 바이트 스트림에 추가하여 Reader와 Writer 클래스를 제공하는데, 이것은 2 바이트를 입출력 할 수 있는 문자 기반 스트림이다.
- 바이트 스트림은 1바이트를 입출력하기 때문에 일반적으로 영문자로 구성된 파일, 동영상 파일, 음악 파일의 입출력 등에 적합한 스트림이다.
- 문자 스트림은 2바이트를 입출력하기 때문에 세계 모든 언어로 구성된 파일을 입출력 하기
에 적합하다
- Reader나 Writer가 붙는다면 문자 스트림이다.

스트림의 종류-Reader

- ❖ Reader 클래스는 문자 입력 스트림의 최상위 추상 클래스이다.
- ❖ InputStream 클래스와 거의 같은 메서드를 제공하고 있으며, 차이점은 Reader 클래스는 2바이트를 읽을 수 있는 메서드로 구성되었다는 점이다.

[표 14-13] Reader의 주요 메서드

반환형	메서드	설명
int	read()	문자 입력 스트림에서 한 개의 문자를 읽어온다. 반환값은 0에서 65535 (0x0000-0xffff) 범위의 유니코드값을 숫자로 반환한다. 문자로 나타내기 위해서는 char로 캐스팅해준다. 더 이상 읽을 수 없을 때는 -1을 반환한다.
	read(char[] cbuf)	문자 입력 스트림에서 문자를 하나씩 읽어 char[]에 저장하고, 읽은 수만큼 반환한다.
abstract int	read(char[] b, int start, int length)	문자 입력 스트림에서 문자를 length만큼 읽어 char[]의 start 위치에 저장하고, 읽을 수만큼 반환한다.
int	skip(long n)	문자 입력 스트림에서 n만큼의 문자를 건너뛴다.
abstract void	close()	문자 입력 스트림을 닫는다.

스트림의 종류-Writer

- ❖ Writer 클래스는 문자 출력 스트림의 최상위 추상 클래스이며, OutputStream과 거의 같은 메서드를 제공한다.
- ❖ Writer 클래스는 2 바이트를 출력할 수 있는 메서드로 구성되어 있다.

[표 14-16] Writer의 주요 메서드

반환형	메서드	설명
abstract void	write(int c)	문자 출력 스트림으로 c값을 char로 변환하여 쓰기한다.
void	write(char[] cbuf)	문자 출력 스트림으로 문자 배열 b를 쓰기한다.
abstract void	write(char[] cbuf, int start, int length)	문자 출력 스트림으로 문자 배열 b를 start부터 length만큼 쓰기한다.
void	write(String str)	문자 출력 스트림으로 문자열 str를 쓰기한다.
abstract void	flush()	버퍼가 가득 차 있지 않더라도 버퍼의 내용을 바로 보내게 된다.
abstract void	close()	문자 출력 스트림을 닫는다.

파일 입출력 스트림의 종류

- ❖ 파일로 부터 데이터를 입출력하기 위한 스트림.
- ❖ 바이트기반과 문자기반 스트림
 - 바이트 단위의 파일을 핸들 할 때에는 FileInputStream과 FileOutputStream을 사용한다.
 - 문자 단위의 파일을 핸들 하고자 할 때에는 FileReader와 FileWriter를 사용한다.

FileInputStream

- ❖ FileInputStream은 시스템에 있는 모든 파일을 읽을 수 있는 기능을 제공한다.
- ❖ 파일을 읽을 때는 파일의 경로와 파일 객체를 생성자의 매개 변수로 설정할 수 있다.
- ❖ 만약, 파일이 존재하지 않으면 FileNotFoundException을 발생하게 된다.

[표 14-5] FileInputStream의 주요 생성자

생성자	설명
FileInputStream(String name)	name이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, 이것을 매개변수로 FileInputStream 객체를 생성한다.
FileInputStream(File file)	File 객체를 이용하여 FileInputStream 객체를 생성한다.

FileOutputStream

- ❖ FileOutputStream은 시스템에 있는 모든 파일에 쓸 수 있는 기능을 제공한다.
- ❖ 만약 객체를 생성할 때 , 파일이 존재하지 않으면 FileNotFoundException을 발생하게 된다.
- ❖ 객체가 생성되면 파일이 있는 경우에는 파일을 생성하지 않으면 파일이 없는 경우에는 파일을 생성하게 된다.
- ❖ FileNotFoundException의 의미는 경로가 일치하지 않을 때 발생하는 예외이다. 즉 경로는 일치하고 파일이 없는 경우에는 예외가 발생하지 않고 파일 생성하게 된다.

FileOutputStream

[표 14-9] FileOutputStream의 주요 생성자

생성자	설명
FileOutputStream (String name)	name이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, 이것을 매개변수로 FileOutputStream 객체를 생성한다.
FileOutputStream (String name, boolean append)	name이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, append가 true이면 이어쓰기의 기능을 하고, false이면 덮어쓰기를 한다. 이 두 개의 매개변수로 FileOutputStream 객체를 생성한다.
FileOutputStream(File file)	File 객체를 이용하여 FileOutputStream 객체를 생성한다.
FileOutputStream (File file, boolean append)	이 두 개의 매개변수로 FileOutputStream 객체를 생성한다. append의 의미는 두 번째 생성자와 동일하다.

FileReader

- ❖ FileReader클래스는 시스템에 있는 파일을 읽을 수 있는 기능을 제공한다.
- ❖ 파일을 읽을 때는 파일의 경로, File 객체를 생성자의 매개변수로 지정할 수 있다.
- ❖ 파일이 존재 하지 않으면 FileNotFoundException 예외를 발생한다.
- ❖ FileReader 클래스는 문자 스트림으로 한 문자를 읽기 때문에 화면에 출력하더라도 한글 깨지는 현상이 일어나지 않는다.

[표 14-14] FileReader의 주요 생성자

생성자	설명
FileReader(String filename)	filename이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, 이것을 매개변수로 FileReader 객체를 생성한다.
FileReader(File file)	File 객체를 이용하여 FileReader 객체를 생성한다.

FileWriter

- ❖ FileWriter 클래스는 문자 파일에 출력할 때 사용하는 클래스이다.
- ❖ FileWriter 클래스의 생성자는 파일의 경로, File 객체를 이용하여 객체를 생성할 수 있다.
- ❖ 만약, 경로가 실제로 존재하지 않으면 IOException를 발생하게 된다.(FileNotFoundException예외가 아님)

[표 14-17] FileWriter의 주요 생성자

생성자	설명
FileWriter(String name)	name이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, 이것을 매개변수로 FileWriter 객체를 생성한다.
FileWriter(String name, boolean append)	name이 의미하는 것은 파일 시스템의 실제 경로를 의미하고, append가 true이면 이어쓰기를 하고, false이면 덮어쓰기를 한다. 이 두 개의 매개변수로 FileWriter 객체를 생성한다.
FileWriter(File file)	File 객체를 이용하여 FileWriter 객체를 생성한다.
FileWriter(File file, boolean append)	이 두 개의 매개변수로 FileWriter 객체를 생성한다. append의 의미는 두 번째 생성자와 동일하다.

BufferedInputStream

- ❖ 버퍼링은 입출력 수행을 향상 시킨 기술이다.
- ❖ 버퍼링이란 논리적 데이터 덩어리들이 하나의 큰 물리적 입력 연산으로서 파일로 부터 읽혀서 버퍼로 입력 하는 것을 말한다.
- ❖ 버퍼링을 이용하면 데이터를 읽어서 버퍼를 꽉 채우고, 연속된 read() 메서드 호출의 경우는 단지 메모리 버퍼로 부터 데이터를 읽어 내는 것일 뿐 이므로 훨씬 효율적이다 또한 mark 기능과 reset 기능을 추가로 구현하였다.

[표 14-7] BufferedInputStream의 주요 생성자

생성자	설명
BufferedInputStream (InputStream in)	매개변수인 InputStream 객체로 BufferedInputStream 객체를 생성한다.
BufferedInputStream (InputStream in, int size)	매개변수인 InputStream 객체로 BufferedInputStream 객체를 생성하고 size는 버퍼의 용량을 정하는 부분인데, 만약 지정하지 않으면 8192byte가 정해져 있다.

BufferedOutputStream

- ❖ 이 클래스를 사용하면 버퍼가 채워질 때마다 한번에 대량으로 출력장치로의 실제 전송이 수행된다.
- ❖ OutputStream은 출력 속도의 향상을 위해서 flush() 메서드를 정의하고 있다. 하지만 실제로는 구현되지 않았다.
- ❖ 플러쉬란 버퍼가 다 차지 않더라도 버퍼를 비워주는 기능을 말한다.
- ❖ 플러쉬 기능을 구현한 클래스가 바로 BufferedOutputStream 클래스가 된다.

[표 14-11] BufferedOutputStream의 주요 생성자

생성자	설명
BufferedOutputStream (OutputStream out)	매개변수인 OutputStream 객체로 BufferedOutputStream 객체를 생성한다.
BufferedOutputStream (OutputStream out, int size)	매개변수인 OutputStream 객체로 BufferedOutputStream 객체를 생성하고 size는 버퍼의 용량을 정하는 부분인데, 만약 지정하지 않으면 8192byte로 정해진다.

BufferedReader

- ❖ 바이트 입력 스트림의 BufferedInputStream과 동일한 기능을 제공하는 BufferedReader 클래스는 문자 출력 스트림의 효율적인 버퍼링을 가능하게 한다.
- ❖ BufferedReader클래스에는 readLine() 메서드가 추가 되었는데, 이 메서드는 한 줄 단위로 읽는 메서드이다.
- ❖ 이 때, 한 줄의 끝을 '\r','\n' 중의 하나가 올 경우 또는 '\r\n'이 오는 경우를 한 줄의 끝으로 간주한다.

[표 14-15] BufferedReader의 생성자

생성자	설명
BufferedReader(Reader in)	매개변수인 Reader 객체로 BufferedReader 객체를 생성한다.
BufferedReader(Reader in, int size)	매개변수인 Reader 객체로 BufferedReader 객체를 생성하고 size는 버퍼의 크기를 정하는 부분인데, 만약 지정하지 않으면 8192문자를 정할 수 있는 버퍼가 생성된다.

BufferedWriter

- ❖ 바이트 출력 스트림의 BufferedOutputStream과 동일한 기능을 제공하며, 문자 출력 스트림의 효율적인 버퍼링을 가능하게 한다.
- ❖ 객체를 생성할 때는 예외처리를 하지 않아도 된다.
- ❖ 이 클래스에서는 한 줄 내려쓰기를 할 수 있는 newLine() 메서드가 추가되었다.

[표 14-18] BufferedWriter의 주요 생성자

생성자	설명
BufferedWriter(Writer out)	매개변수인 Reader 객체로 BufferedWriter 객체를 생성한다.
BufferedWriter (Writer out, int size)	매개변수인 Reader 객체로 BufferedWriter 객체를 생성하고 size는 버퍼의 용량을 정하는 부분인데, 만약 지정하지 않으면 8192문자를 정할 수 있는 버퍼가 생성된다.

DataInputStream

- ❖ DataInput 인터페이스는 입력 스트림으로 부터 기본형 데이터를 읽기 위한 메서드를 정의한다.
- ❖ DataInput 인터페이스는 기본 자료형을 읽을 수 있는 각종 메서드와 문자를 읽을 수 있는 메서드를 정의 하고 있다.
- ❖ DataInputStream 클래스의 생성자는 한 개로 구성되어 있으며, 어떠한 예외 처리도 되어 있지 않다.

[표 14-6] DataInputStream의 생성자

생성자	설명
DataInputStream(InputStream in)	매개변수인 InputStream 객체로 DataInputStream 객체를 생성한다.

DataOutputStream

- ❖ DataOutput 인터페이스는 출력 스트림으로 부터 기본형 데이터를 쓰기 위한 메서드를 정의한다.
- ❖ DataOutput 인터페이스는 기본 자료형을 쓸 수 있는 각종 메서드와 문자를 쓸 수 있는 메서드를 정의 하고 있다.
- ❖ DataOutputStream 클래스의 생성자는 한 개로 구성되어 있으며, 어떠한 예외 처리도 되어 있지 않다.

[표 14-10] DataOutputStream의 주요 생성자

생성자	설명
DataOutputStream(OutputStream out)	매개변수인 OutputStream 객체로 DataOutputStream 객체를 생성한다.

객체의 직렬화

❖ 객체의 직렬화

- 객체의 직렬화란 객체를 스트림으로 파일에 저장하는 방법이다.
- 객체를 직렬화 하기위한 두 가지 방법으로 Serializable과 Externalizable 인터페이스 구현하면 된다.

❖ Serializable 인터페이스

- Serializable 인터페이스를 구현한 클래스를 작성하면 해당 클래스의 모든 멤버 변수가 직렬화 대상이 된다.
- 객체가 스트림을 통해 직렬화 될 때는 객체에 있는 멤버변수가 직렬화 되는 것이다
- 객체의 멤버변수 중에 직렬화 대상에 제외하고 싶다면 transient 키워드를 사용하면 된다.

객체의 직렬화-ObjectOutputStream

❖ 객체의 직렬화

- 객체를 직렬화 하고 다시 역직렬화 시킬 수 있는 클래스가 `ObjectInputStream` 과 `ObjectOutputStream` 이다.

❖ ObjectOutputStream

- `ObjectOutput` 인터페이스를 구현한 클래스로 객체를 파일에 기록 가능한 클래스이다.
- `ObjectOutput` 인터페이스는 `writeObject(Object obj)` 메서드를 포함하는데 이 메서드가 객체의 데이터를 직렬화 시켜주는 메서드(직렬화 메서드)이다.
- 만약 `obj`가 `Serializable` 인터페이스로 구현되어 있지 않다면 `NotSerializableException` 예외가 발생한다.
- 보통 직렬화된 데이터를 저장할 파일 확장자는 "ser"로 정하는 것이 관례이다.

객체의 직렬화-ObjectInputStream

❖ ObjectInputStream

- ObjectInput 인터페이스를 구현한 클래스로 직렬화 된 객체를 읽어올 수 있는 클래스이다.
- ObjectInput 인터페이스는 readObject() 메서드를 포함하는데 이 메서드는 객체의 데이터를 복원 시켜주는 메서드(역 직렬화 메서드)이다.

InputStreamReader

❖ InputStreamReader

- 바이트 입력 스트림 => 문자 입력 스트림

[표 14-20] InputStreamReader의 주요 생성자

생성자	설명
<code>InputStreamReader(InputStream in)</code>	기본 문자셋을 이용하여 <code>InputStreamReader</code> 객체를 생성한다.
<code>InputStreamReader(InputStream in, String charsetName)</code>	<code>charName</code> 으로 문자셋을 지정하여 <code>InputStreamReader</code> 객체를 생성한다.

OutputStreamWriter

❖ OutputStreamWriter

- 바이트 출력 스트림 = > 문자 출력 스트림

[표 14-21] OutputStreamWriter의 주요 생성자

생성자	설명
OutputStreamWriter(OutputStream out)	기본 문자셋을 이용하여 OutputStreamWriter 객체를 생성한다.
OutputStreamWriter(OutputStream out, String charsetName)	charName으로 문자셋을 지정하여 OutputStreamWrite 객체를 생성한다.

PrintStream

- ❖ PrintStream은 모든 자료형을 출력할 수 있는 print(), println() 메서드가 오버로딩 되어 있다.
- ❖ 프로그램이 시작되면 장치와 연결된 출력스트림인 System.out, System.err 객체가 PrintStream 객체이다.
- ❖ 자바 5.0에서는 PrintStream의 format() 메서드와 printf() 메서드가 추가되어 있기 때문에 이전의 System.out.printf() 나 System.out.format()을 이용해서 출력문을 작성할 수 있었다.
- ❖ PrintStream은 두 가지 중용한 특징을 가진다.

첫번째, 다른 스트림과는 다르게 플러쉬 기능을 자동으로 처리할 수 있는 생성자를 가지고 있다.

두번째, 모든 메서드의 예외처리를 하지 않았다는 점이다.

PrintWriter

- ❖ PrintWriter 클래스는 다른 스트림과 다르게 바이트 출력 스트림과 문자 출력 스트림을 가지고 객체를 생성할 수 있는 클래스이다.
- ❖ 자동 플러쉬 기능을 가지고 있다.
- ❖ PrintWriter 클래스의 생성자에는 FileNotFoundException 예외를 발생하기 때문에 반드시 예외처리를 해야 한다.

[표 14-19] PrintWriter의 주요 생성자

생성자	설명
PrintWriter(OutputStream out)	자동 플러쉬 없이 OutputStream 객체로 PrintWriter 객체를 생성한다.
PrintWriter(OutputStream out, boolean autoFlush)	자동 플러쉬를 할 수 있는 PrintWriter 객체를 생성한다.
PrintWriter(Writer out)	자동 플러쉬 없이 Writer 객체로 PrintWriter 객체를 생성한다.
PrintWriter(Writer out, boolean autoFlush)	행 플러쉬를 할 수 있는 PrintWriter 객체를 생성한다.

JDK 입출력관련 클래스들 요약

문자 스트림 클래스	바이트 스트림 클래스	설명
Reader	InputStream	문자/바이트 입력 스트림을 위한 추상클래스
BufferedReader	BufferedInputStream	문자/바이트 버퍼 입력, 라인 해석
LineNumberReader	LineNumberInputStream	문자/바이트 입력 시, 라인 번호를 유지
CharArrayReader	ByteArrayInputStream	문자/바이트 배열에서 읽어 들임
InputStreamReader	(none)	바이트 스트림을 문자 스트림으로 변환
FileReader	FileInputStream	파일에서 바이트를 읽어 들어 문자/바이트 스트림으로 변환
FilterReader	FilterInputStream	필터적용(filtered) 문자/바이트 입력을 위한 추상클래스
PushbackReader	PushbackInputStream	읽어 들인 문자/바이트를 되돌림(pushback)
PipedReader	PipedInputStream	PipedWriter/PipedOutputStream에서 읽어 들임
StringReader	StringBufferInputStream	문자열에서 읽어 들임
Writer	OutputStream	문자 출력 스트림을 위한 추상클래스
BufferedWriter	BufferedOutputStream	문자/바이트 스트림에 버퍼 출력, BufferedWriter는 플랫폼에서 사용하는 라인 구분자(line separator) 사용
CharArrayWriter	ByteArrayOutputStream	문자/바이트 스트림에 문자/바이트 배열 출력
FilterWriter	FilterOutputStream	필터적용(filtered) 문자/바이트 출력을 위한 추상클래스
OutputStreamWriter	(none)	문자 스트림을 바이트 스트림으로 변환
FileWriter	FileOutputStream	문자/바이트 스트림을 바이트 파일로 변환
PrintWriter	PrintStream	Writer/Stream에 값과 객체를 프린트
PipedWriter	PipedOutputStream	PipedReader/PipedOutputStream에 출력
StringWriter	(none)	문자열 출력