

10

문자 스트림

최 문 환



1. Reader 클래스와 Writer 클래스

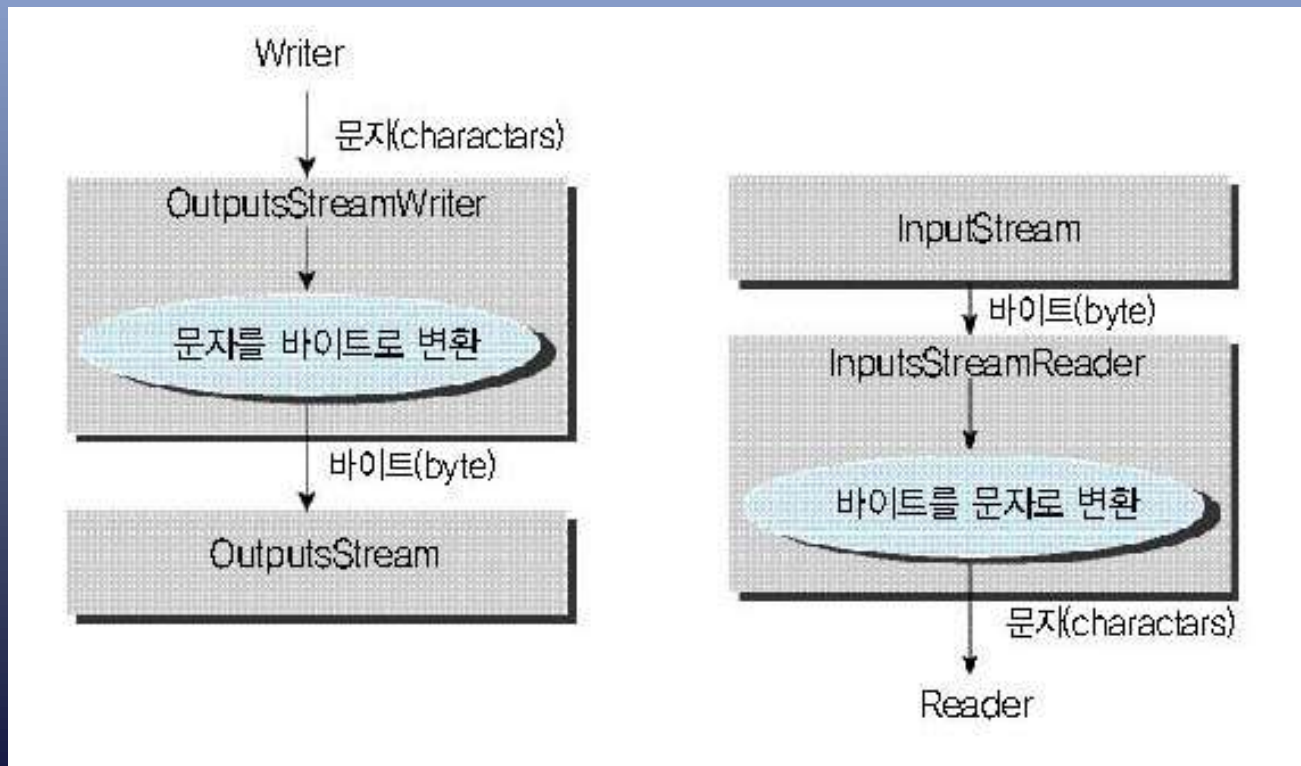
1. Reader 클래스는 문자 입력 스트림의 최상위 추상클래스 이다.

2. Writer 클래스는 문자 출력 스트림의 최상위 추상클래스 이다.

2. InputStreamReader 클래스와 OutputStreamWriter 클래스

InputStreamReader(InputStream instreams)
InputStreamReader(InputStream instreams, String encoding)

OutputStreamWriter(OutputStream outstreams)
OutputStreamWriter(OutputStream outstreams, String encoding)



3. FileReader 클래스와 FileWriter 클래스

생성자	설명
FileWriter (File file)	주어진 File 객체를 이용하여 FileWriter 객체 생성
FileWriter (String fileName)	주어진 파일을 열어 FileWriter 객체 생성
FileWriter(String fileName, boolean append)	주어진 파일을 append 값에 따라 읽기/추가 모드로 열어 FileWriter 객체 생성

생성자	설명
FileReader(File file)	주어진 File 객체를 이용하여 FileReader 객체 생성
FileReader(String fileName)	주어진 파일을 열어 FileReader 객체 생성

4. BufferedReader 클래스

생성자	
BufferedReader(Reader in)	입력 스트림에 대한 디폴트 크기의 내부 버퍼를 갖는 객체를 생성
BufferedReader(Reader in, int sz)	주어진 크기의 내부 버퍼를 갖는 객체를 생성

1. BufferedReader : Reader로 부터 상속된 클래스로 문자, 배열, 행을 버퍼링 하는 것에 의해, 문자형 입력 스트림로부터 텍스트를 효율 좋게 읽어들이는다.

<예제> 한줄 단위로 문자열 입력받아 처리하기

```
001:import java.io.*;
002:class ReaderWriterTest01{
003:    public static void main(String[] args) {
004:        String fileName;
005:        //InputStreamReader 객체 생성, InputStream을 InputStreamReader로 바꿈
006:        InputStreamReader myIn=new InputStreamReader(System.in);
007:        BufferedReader keyBr =new BufferedReader(myIn);
008:        try{
009:            System.out.print("파일 이름을 입력하세요->");
010:
011:            fileName= keyBr.readLine( );
012:            System.out.println(fileName );
013:            keyBr.close();
014:            myIn.close();
015:        }catch(IOException e){
016:            System.out.print(e);
017:        }
018:    }
019: }
```

Reader 클래스를 이용해서 파일 읽어오는 순서

1. 파일 리더 객체 생성

```
FileReader fr = new FileReader("파일이름");
```

2. 버퍼 리더 객체 생성

```
BufferedReader fileBr=new BufferedReader(fr);
```

3. 파일에서 데이터를 읽어와 표준 출력장치인 모니터에 출력한다.

```
while((buf = fileBr.readLine()) != null) {  
    System.out.println(buf);  
}
```

<예제> 파일명 입력받아 파일 내용을 화면에 출력하기

```
001:import java.io.*;
002:class FileType02{
003:    public static void main(String[] args) {
004:        String fileName;
005:        String buf;//키보드에서 라인단위로 읽어온 데이터를 저장할 변수
006:        //InputStreamReader 객체 생성, InputStream을 InputStreamReader로
007:        Reader myIn=new InputStreamReader(System.in);
008:        //버퍼리더 객체 생성 버퍼리더는 리더형 인자만 받음
009:        BufferedReader keyBr =new BufferedReader(myIn);
010:        try{
011:            System.out.print("파일 이름을 입력하세요->");
012:
013:            fileName=keyBr.readLine( ); //키보드에서 파일이름 입력 받음
014:
```


<예제> 파일명 입력받아 파일 내용을 화면에 출력하기

```
015:      FileReader fr=new FileReader(fileName);
016:      BufferedReader fileBr=new BufferedReader(fr);
017:      while((buf = fileBr.readLine()) != null) {
018:          System.out.println(buf);
019:      }
020:      fileBr.close();
021:      keyBr.close();
022:      myIn.close();
023:  }catch(IOException e){
024:      System.out.print(e);
025:  }
026: }
027: }
```

5. BufferedWriter 클래스

생성자	
BufferedWriter(Writer out)	주어진 문자 출력 스트림에 대한 디폴트 크기의 내부 버퍼를 갖는 객체 생성
BufferedWriter(Writer out, int sz)	주어진 문자 출력 스트림에 대한 주어진 크기의 내부 버퍼를 갖는 객체 생성

<예제> 키보드에서 입력받은 내용을 파일에 기록하기

```
001:import java.io.*;
002:class ReaderWriterTest03{
003: public static void main(String[] args) {
004:   String fileName; //파일 명을 저장할 변수
005:   String buf;       //데이터를 읽어와 저장할 변수
006:   Reader myIn=new InputStreamReader(System.in);
007:   BufferedReader keyBr =new BufferedReader(myIn);
```

<예제> 키보드에서 입력받은 내용을 파일에 기록하기

```
008:     try{
009:         System.out.print("파일 이름을 입력하세요->");
010:         fileName=keyBr.readLine( );           //키보드에서 파일이름 입력 받음
011:         FileWriter fw=new FileWriter(fileName); //BufferedWriter 객체 생성
012:         BufferedWriter fileBw=new BufferedWriter(fw); //버퍼리더 생성자에 리더 객체 전달
013:         System.out.println("파일에 저장할 내용을 입력하세요.");
014:         while((buf = keyBr.readLine()) != null) { //키보드에서 라인 단위로 읽어 와서
015:             fileBw.write(buf);                    //파일에 출력
016:             fileBw.newLine();
017:         }
018:         fileBw.close();                          //완료된 후에 각 파일 입력 스트림을 닫음
019:         keyBr.close();
020:         myIn.close();
021:     }catch(IOException e){
022:         System.out.print(e);
023:     }
024: }
025: }
```