I/O Streams

- Reading: Savitch, Chapter 9

Objectives

- To learn the *java.io* package.

Streams

- A stream is a sequence of data.
- Data might be characters or bytes.
- If data flow into the program, the stream is an input stream. If data flow out of the program, the stream is an output stream.

Standard I/O

- Three streams are called the standard I/O streams.

System.in	Standard input stream
System.out	Standard output stream
System.err	Standard error stream (output for error messages)

- By default, the standard input stream is from the keyboard; the standard output and error streams go to the monitor.

Standard Input Stream

- **The System class provides a stream for reading text from the standard input stream.**
  - **System.in is a static variable of System class which is a data type of InputStream**

public class InputStreamReader extends Reader

An InputStreamReader is a bridge from byte streams to character streams:

- **It reads bytes and decodes them into characters using a specified charset.**
- **The charset that it uses may be specified by name or may be given explicitly, or the platform's default charset may be accepted.**

Standard Input Stream

Reading from the Standard Input Stream

- **The System.in.read() method reads a single character and returns either the integer (0-255) representing the character that was read or, if there are no more characters to be read, -1.**

1. **Turn the InputStream object System.in into a Reader object.**
   - **The InputStreamReader class can do the job;**
   - **But , it can read one character only**

```
InputStreamReader reader = new
      InputStreamReader(System.in)
```

Standard Input Stream

2. **The BufferReader class can read entire lines at a time. More efficient**

```
BufferedReader keyboard = new
                          BufferedReader (reader )
```

Get primitive data from Keyboard

- **There is no simple way to get primitive data from the standard input – keyboard (see slide 22).**
- **The following class KeyboardInput provides a simple routine for primitive values.**
- **System.in object read bytes from keyboard and convert to Long or Double data type.**

You need to import classes from java.io package first.

```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;
```

```
class KeyBoardInput extends Object {
   private static BufferedReader keyboard = new
   BufferedReader(new InputStreamReader(System.in));
   protected static long getLong() throws IOException
   {
      long localLong = 0;   // long primitive data type
      String tempString = new String();
      try {
         tempString = keyboard.readLine();
                  // the readLine() is supplied by the BufferedReader
         localLong = Long.parseLong(tempString);
        /* static method parseLong returns a long primitive data type */
      } catch (java.lang.Exception exception)
      {
         throw new java.io.IOException();
      } // End try/catch.
      System.out.println("The localLong value you input
   is"+localLong);

  return localLong;
    }        // End getLong
 // other methods getFloat etc..
}
```

Example

```
BufferedReader br = new BufferedReader
                    (new InputStreamReader (System.in));

    System.out.println("Enter a positive integer from the
    keyboard");
    int k = Integer.parseInt(br.readLine());
    if(k <= 0) {
          System.err.println("An incorrect input.");
          System.exit(0);
    }
```

Standard I/O Reassignment

- setIn(), setOut() and setErr() are three methods defined in *java.lang.System* to reassign the "standard" input, output and error streams.

public static void **setIn**(InputStream in)
public static void **setOut**(PrintStream out)
public static void **setErr**(PrintStream err)

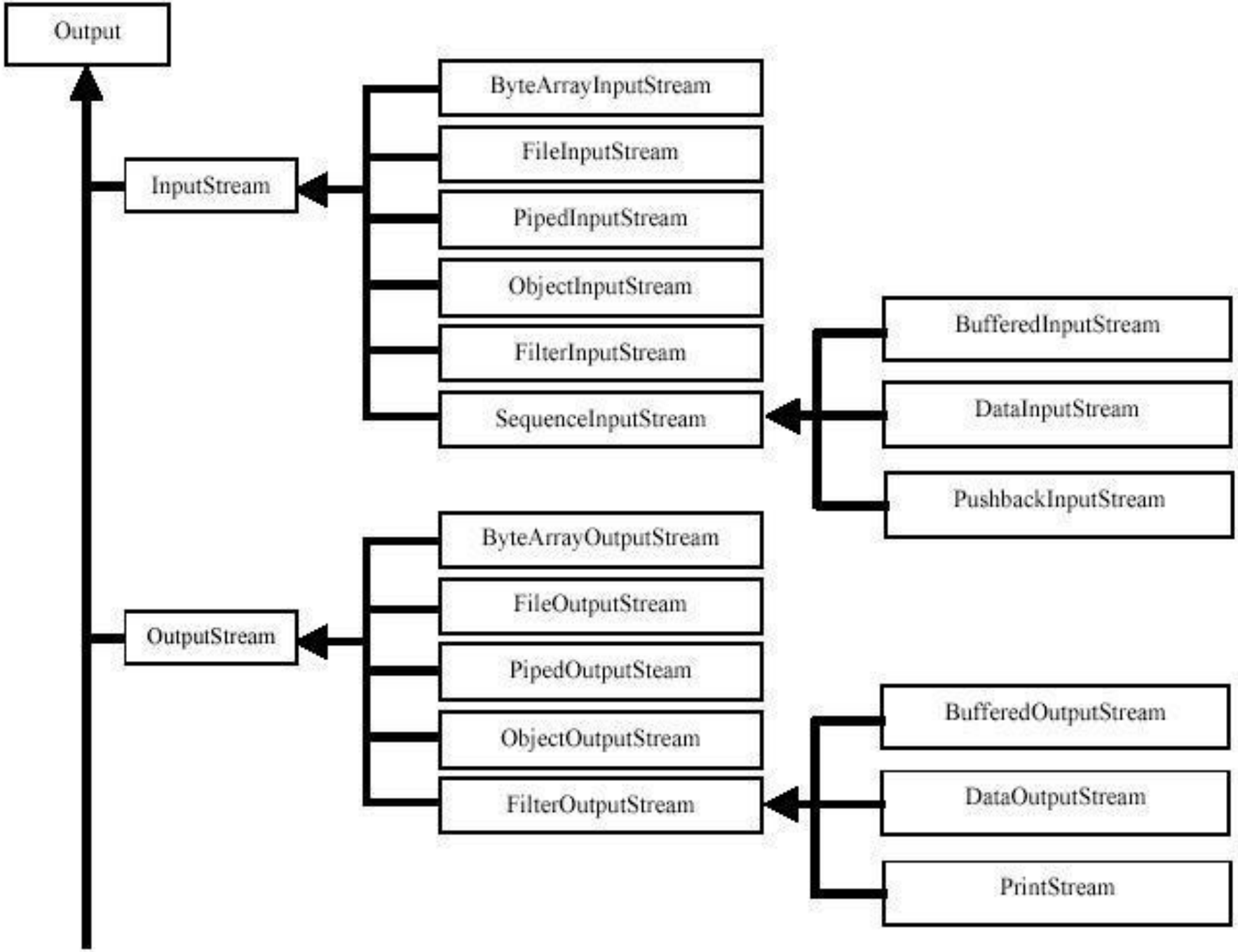
Example

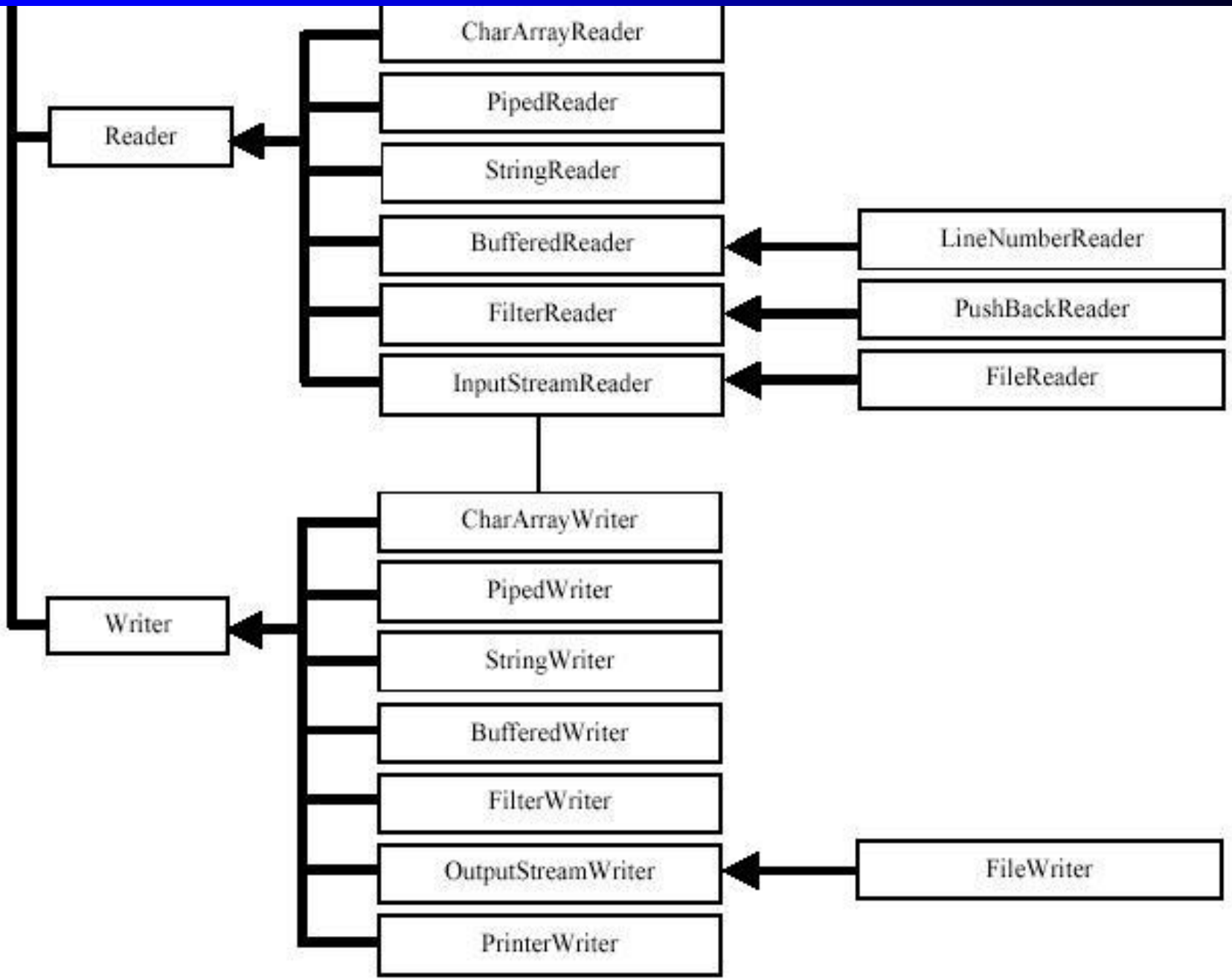
```
PrintStream out = new PrintStream (new
                    FileOutputStream ("myOutputFile"));
System.setOut(out);
System.out.println(“This line will be output to
                    myOutputFile, instead of the monitor.”);
```

The I/O Class Hierarchies

- All classes defined in *java.io* are for input/output purposes.


JAVAIOClassesnote20.pdf


Output
InputStream
ByteArrayInputStream
FileInputStream
PipedInputStream
ObjectInputStream
FilterInputStream
SequenceInputStream
BufferedInputStream
DataInputStream
PushbackInputStream
OutputStream
ByteArrayOutputStream
FileOutputStream
PipedOutputSteam
ObjectOutputStream
FilterOutputStream
BufferedOutputStream
DataOutputStream
PrintStream


CharArrayReader
PipedReader
Reader
StringReader
BufferedReader
LineNumberReader
FilterReader
PushBackReader
InputStreamReader
FileReader
CharArrayWriter
PipedWriter
Writer
StringWriter
BufferedWriter
FilterWriter
OutputStreamWriter
FileWriter
PrinterWriter

- InputSteam, OutputStream and their subclasses are used for byte stream reading/writing

- Reader, Writer and their subclasses are used for char stream reading/writing

Class Exercise

- Write a program which sets reading from a file (rather than from the keyboard) as the standard input.

- **setIn**(InputStream in)

Example

```
//StandardInOutReAssign.java
//reads 10 integers from myInputFile and
//outputs the average into myOutputFile.

import java.io.*;
class StandardInOutReAssign {
    public static void main (String[] args)
                                throws IOException  {
    String inString;
    int i, num, sum = 0;
```

```
DataInputStream in = new DataInputStream (new
FileInputStream ("myInputFile"));
System.setIn(in);  //reset System.in to a file
BufferedReader stdin = new BufferedReader
      (new InputStreamReader (System.in));
                  //System.in refers myInputFile now
for ( i = 1 ; i <= 10 ; i++)  {
      num = Integer.parseInt (stdin.readLine());
      sum += num;
      if(num < 0) {
            System.err.println("incorrect input");
            System.exit(0);
      }
}
```

```
            FileOutputStream outTemp = new
            FileOutputStream ("myOutputFile");
        PrintStream out = new PrintStream (outTemp);
        System.setOut(out);    //out refers myOutPut now

            System.out.println("The average is: " +
                                        (double)sum / 10);
    }
}
```

Scanner Class

- Java 1.5 introduced the Scanner class to simplify reading primitive data types from the keyboard.
- To use the Scanner class, add the line
  import java.util.*;
  to your program.