JAC444 / BTP400 Course Object-Oriented Software Development II - Java

Basics Input / Output

Segment 1

Objectives

Upon completion of this lecture, you should be able to:

- Understand Input / Output classes in Java
- Create and Use I/O Streams in Java
- Distinguish Byte, Character, and Buffered Stream
- Design and Develop File I/O programs

Input / Output

In this lesson you will be learning about:

- Types of Input / Output Stream
- Typical use of IO Streams
- Character and Byte Streams.

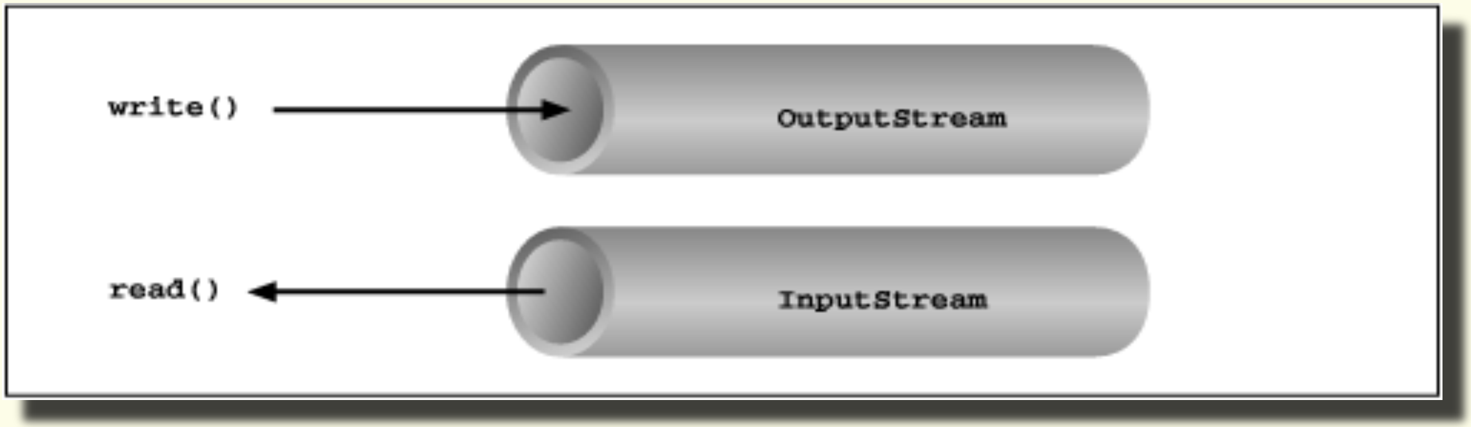
Reading / Writing Data

Reading

```
Open a stream
while more information
        read information
close the stream
```

Writing

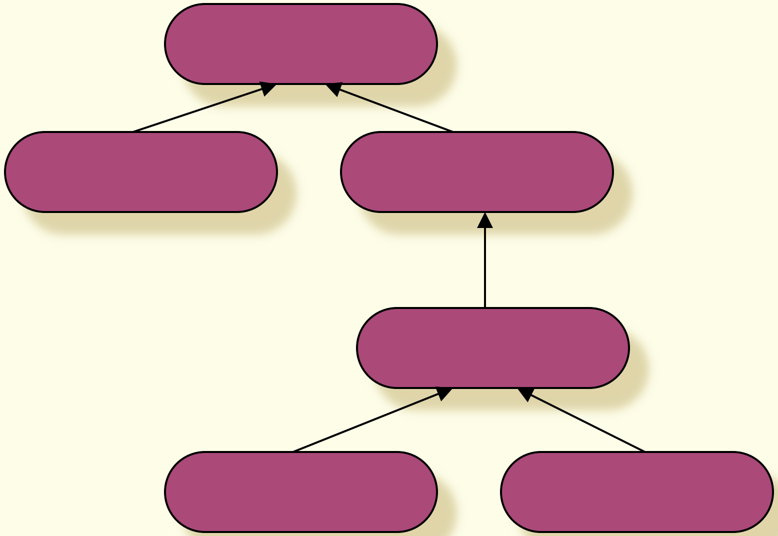
```
open a stream
while more information
        write information
close the stream
```



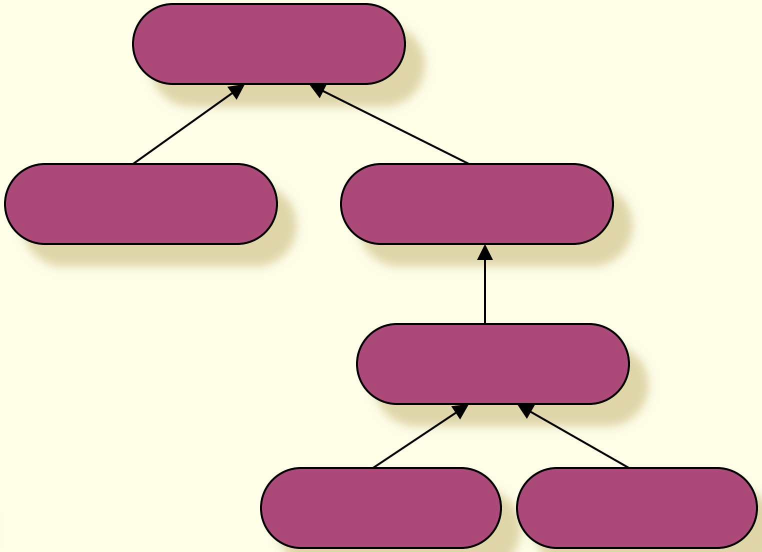
I/O Package

1. Contains a collection of classes that support I/O algorithms.
2. These classes are divided into two class hierarchies based on the data type: either characters or bytes on which they operate.

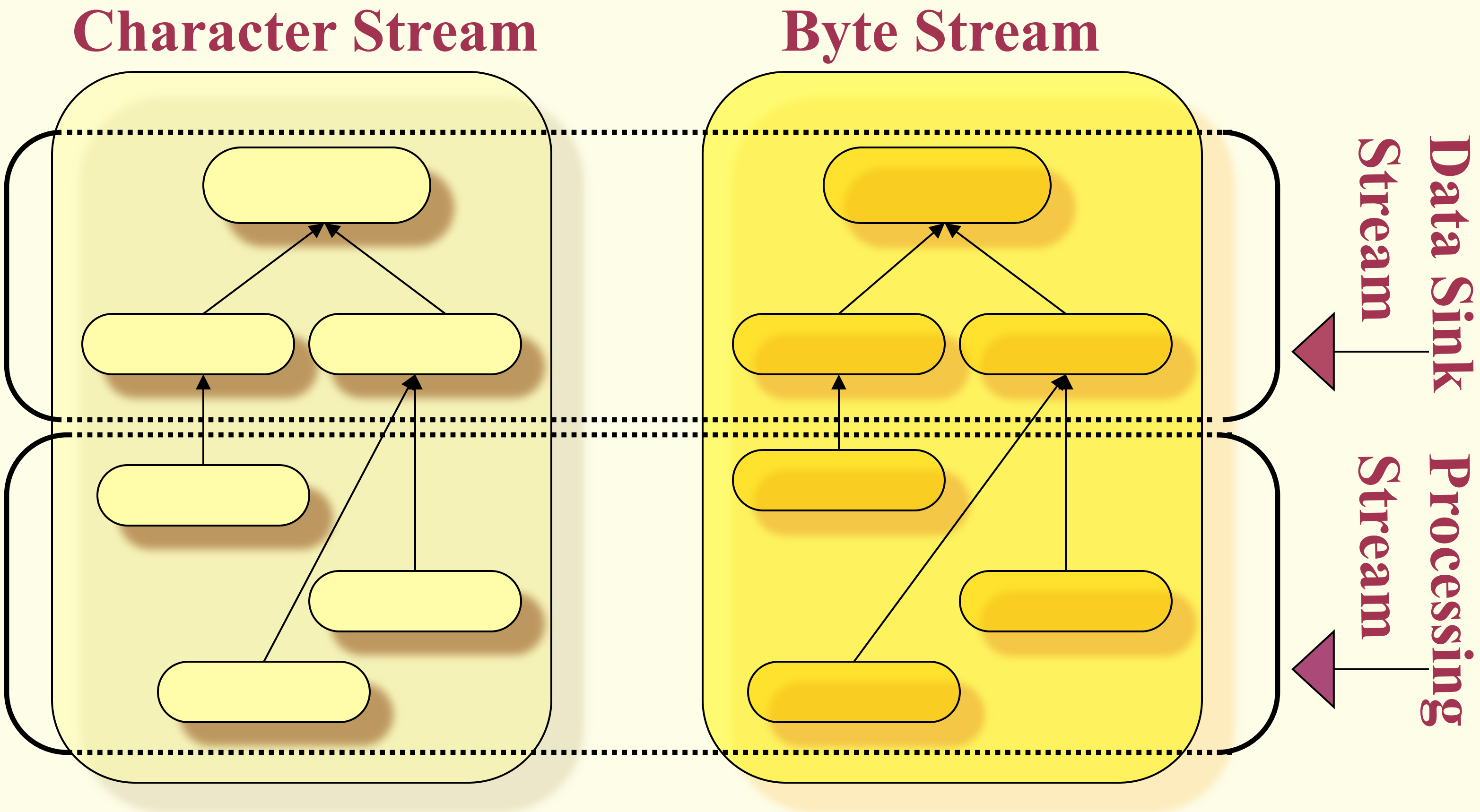
CHARS Reader/Writer



BYTES Input/Output Streams



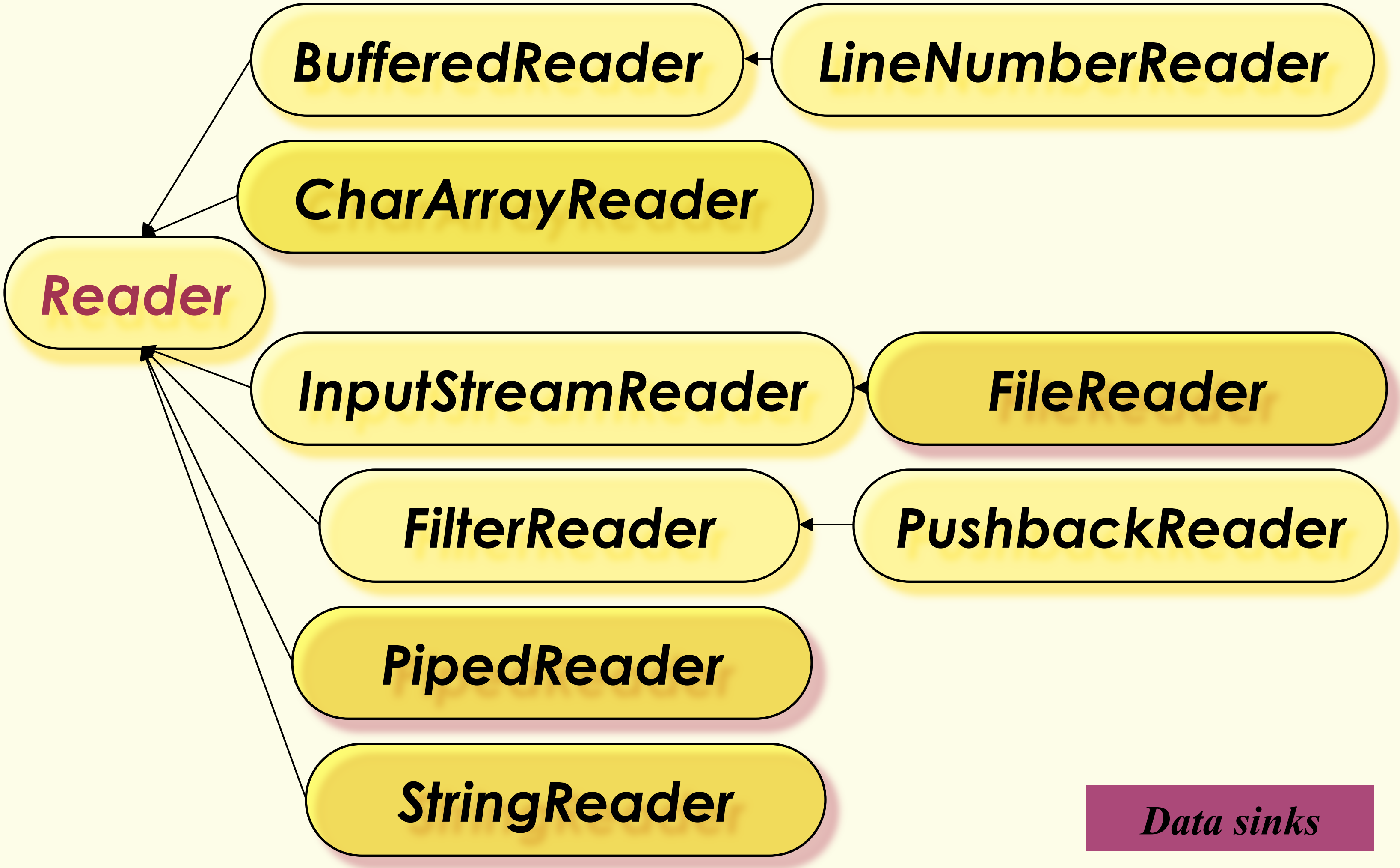
Logical Group of IO Classes



Character Stream

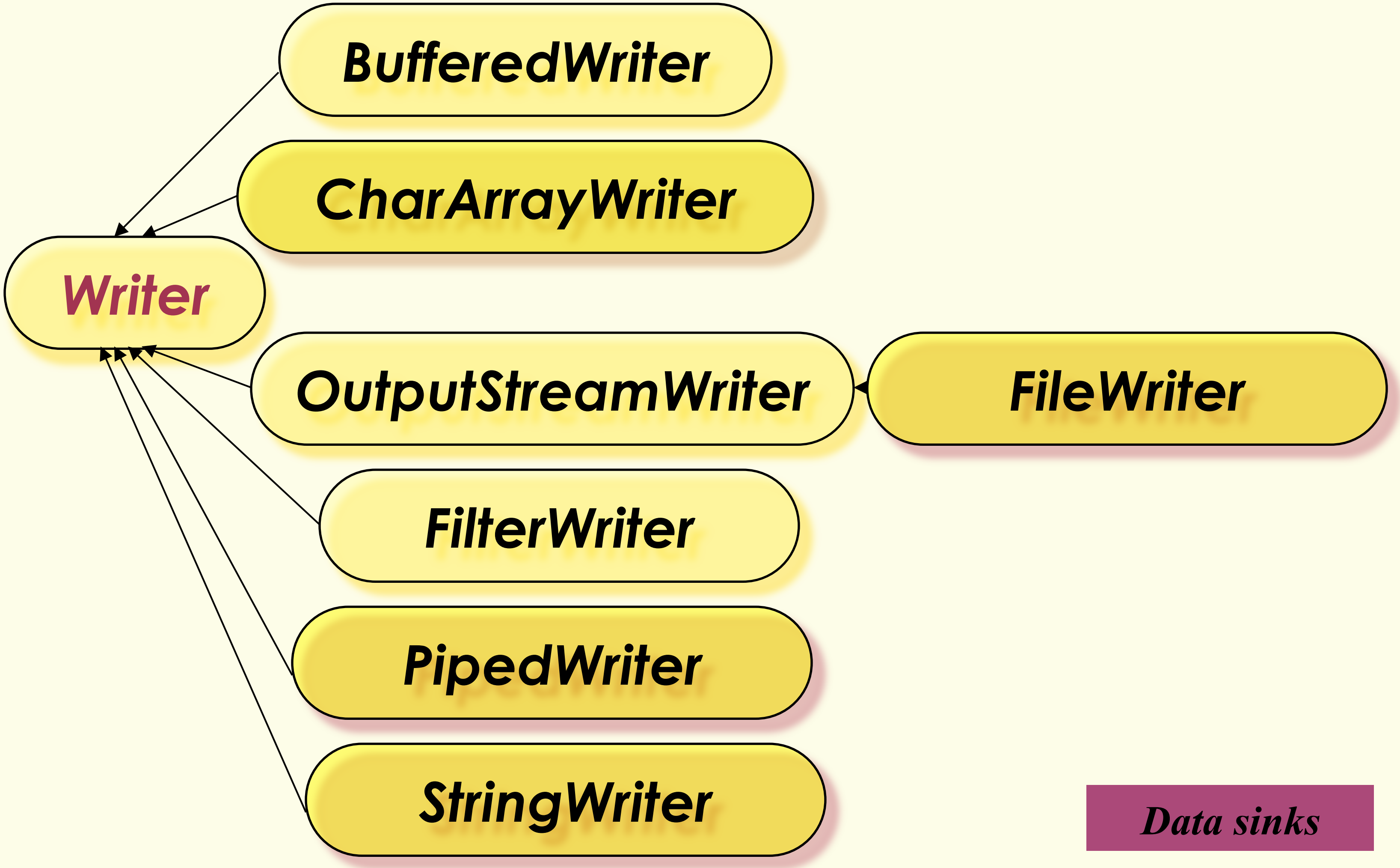
- `Reader/Writer` are abstract superclasses for character streams in `java.io` package.
- `Reader` Abstract class for reading character streams and provides methods for reading 16-bit characters.
- The only methods that a subclass must implement are `read(char[],int, int)` and `close()`.
- `Writer` Abstract class for writing to character streams and provides methods for writing 16-bit characters.
- The only methods that a subclass must implement are `write(char[], int, int)`, `flush()`, and `close()`.

Reader



Data sinks

Writer

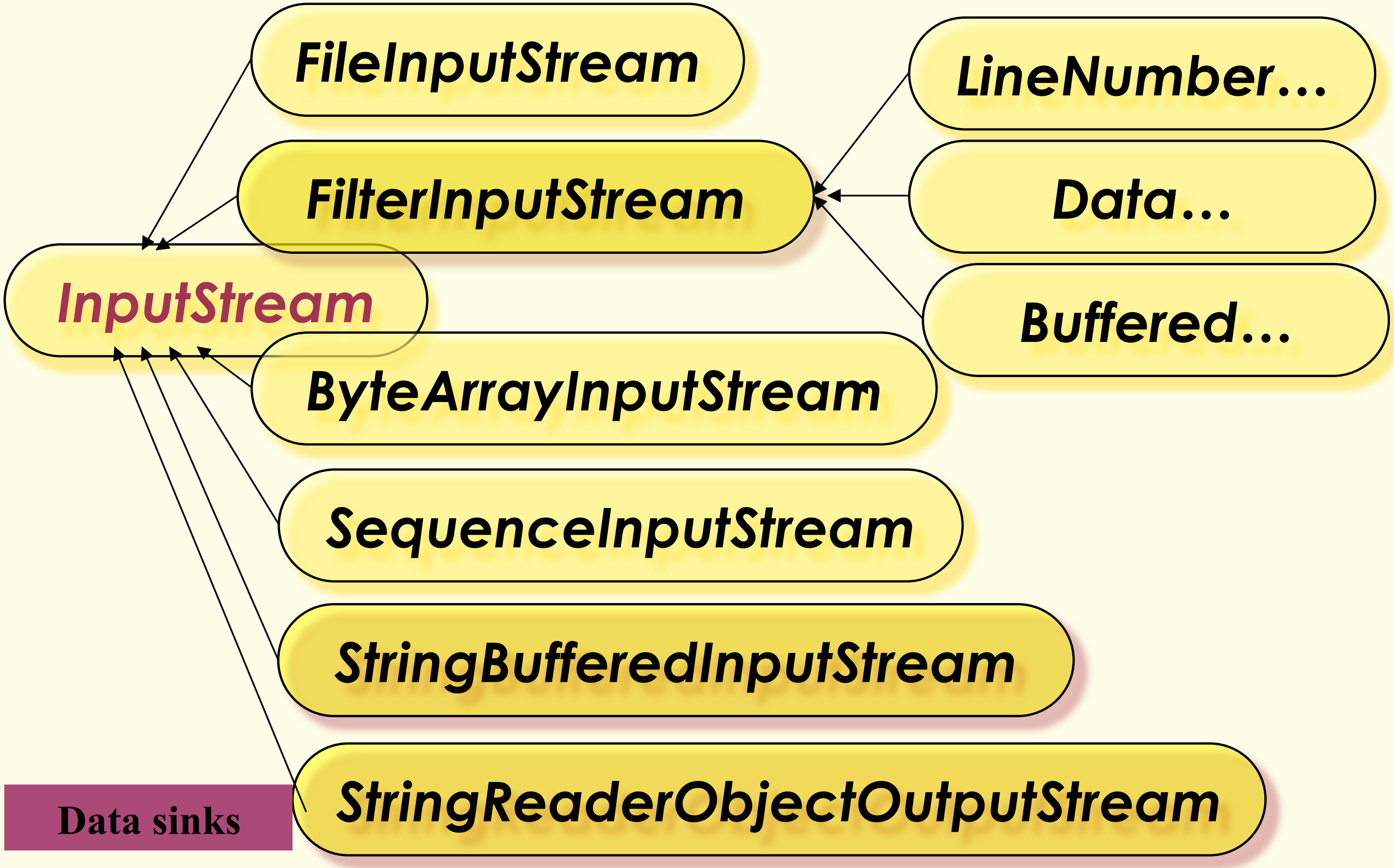


Data sinks

Byte Streams

- `InputStream` and `OutputStream` provide the methods and some implementation for read and write 8bit bytes.
- Streams are typically used to read and write binary data such images and sounds
- Input/Output stream subclasses provide specialized IO that falls into two categories:
  - Data Sink Stream
  - Processing Stream

Input Stream



FileInputStream

FilterInputStream

LineNumber…

Data…

Buffered…

InputStream

ByteArrayInputStream

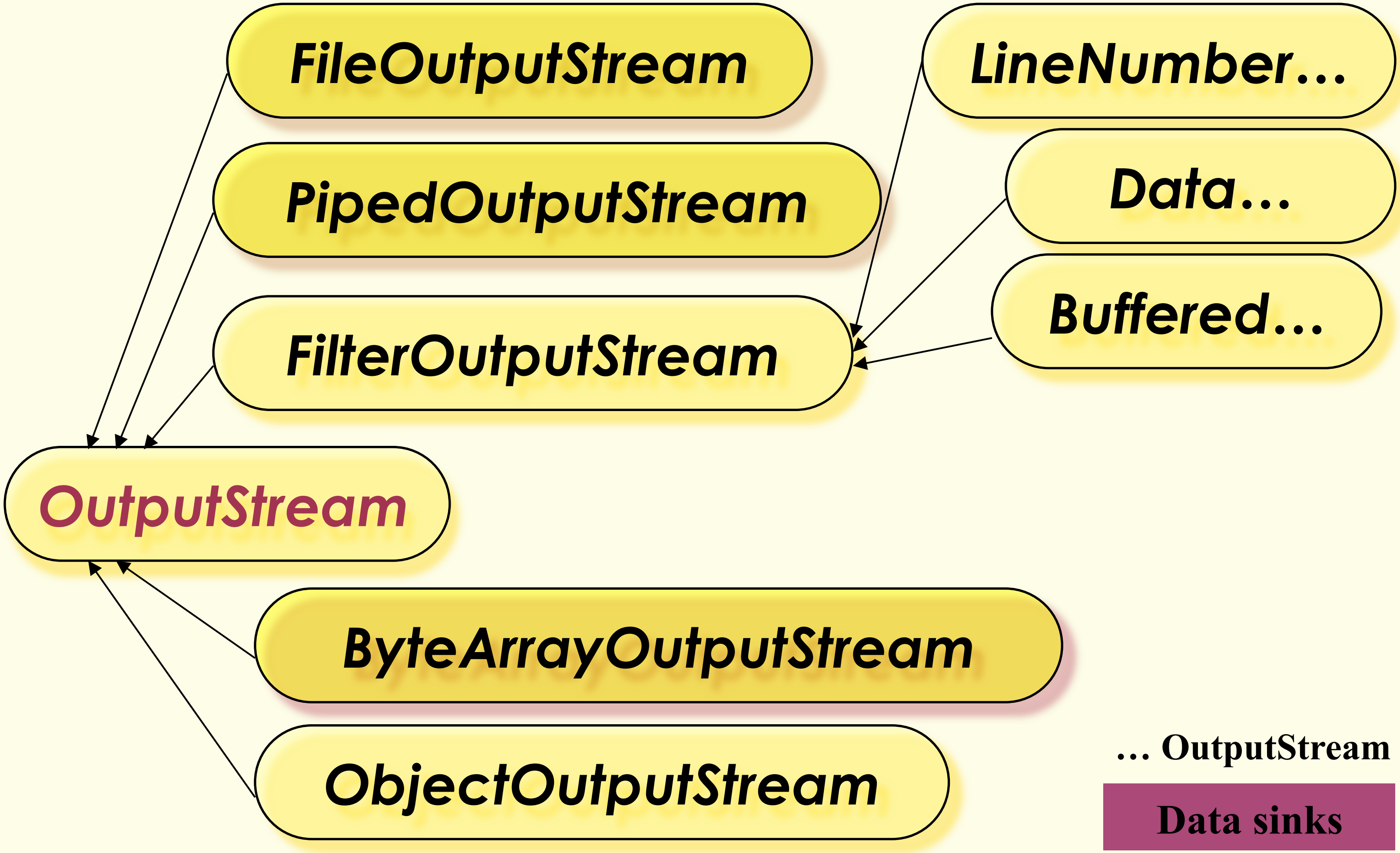
SequenceInputStream

StringBufferedInputStream

StringReaderObjectOutputStream

Data sinks

Output Stream



… OutputStream

Data sinks

Byte Streams Question

- `InputStream` and `OutputStream` provide the methods and some implementation for read and write 8bit bytes.

- Streams are typically used to read and write binary data such images and sounds

- `int read() throws IOException`

Returns: the next byte of data, or -1 if the end of the stream is reached

Question: The value of a byte b is -128 <= b <= 127

If the value of the b = -1, how do we know when we invoke ***`read`****() method that a byte is returned with the value -1 or the end of the stream is reached ?*