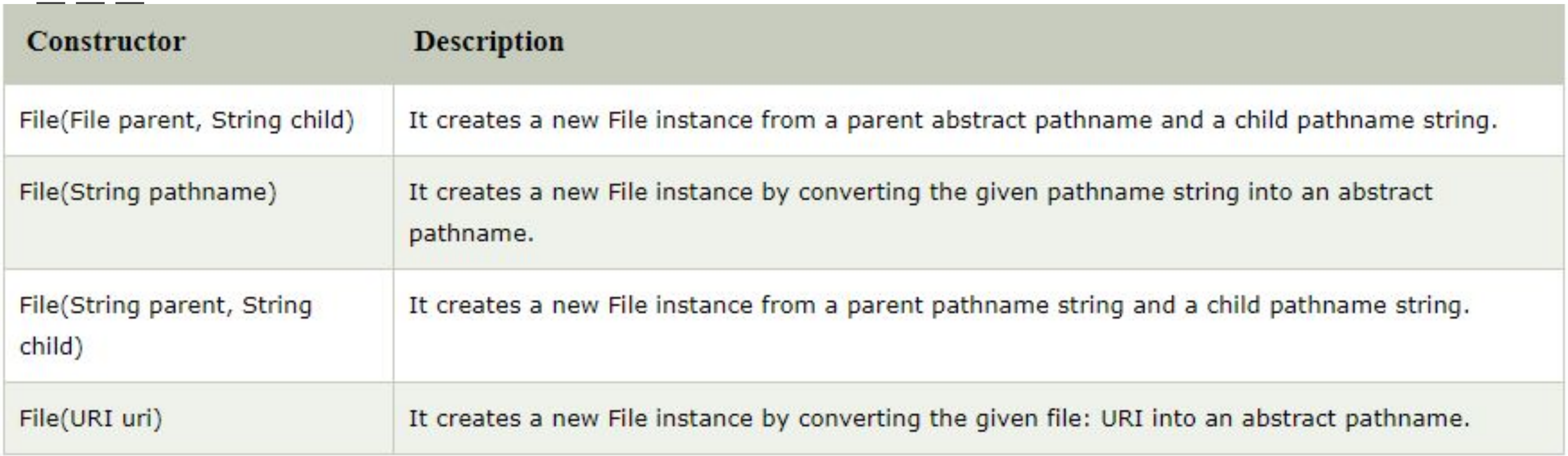
File Handling In Java

Deshan Koshala Sumanathilaka

File Class

- **Java File class represents the files and directory pathnames in an abstract manner.**
- **used for creation of files and directories, file searching, file deletion, etc.**



Constructor	Description
File(File parent, String child)	It creates a new File instance from a parent abstract pathname and a child pathname string.
File(String pathname)	It creates a new File instance by converting the given pathname string into an abstract pathname.
File(String parent, String child)	It creates a new File instance from a parent pathname string and a child pathname string.
File(URI uri)	It creates a new File instance by converting the given file: URI into an abstract pathname.

public String getName()

Returns the name of the file or directory denoted by this abstract pathname.

— — —

public String getPath()

Converts this abstract pathname into a pathname string.

public String getAbsolutePath()

Returns the absolute pathname string of this abstract pathname.

public String getParent()

Returns the pathname string of this abstract pathname's parent, or null if this pathname does not name a parent directory.

public boolean exists()

Tests whether the file or directory denoted by this abstract pathname exists. Returns true if and only if the file or directory denoted by this abstract pathname exists; false otherwise.

public boolean canWrite()

Tests whether the application can modify to the file denoted by this abstract pathname. Returns true if and only if the file system actually contains a file denoted by this abstract pathname and the application is allowed to write to the file; false otherwise.

public boolean canRead()

Tests whether the application can read the file denoted by this abstract pathname. Returns true if and only if the file specified by this abstract pathname exists and can be read by the application; false otherwise.

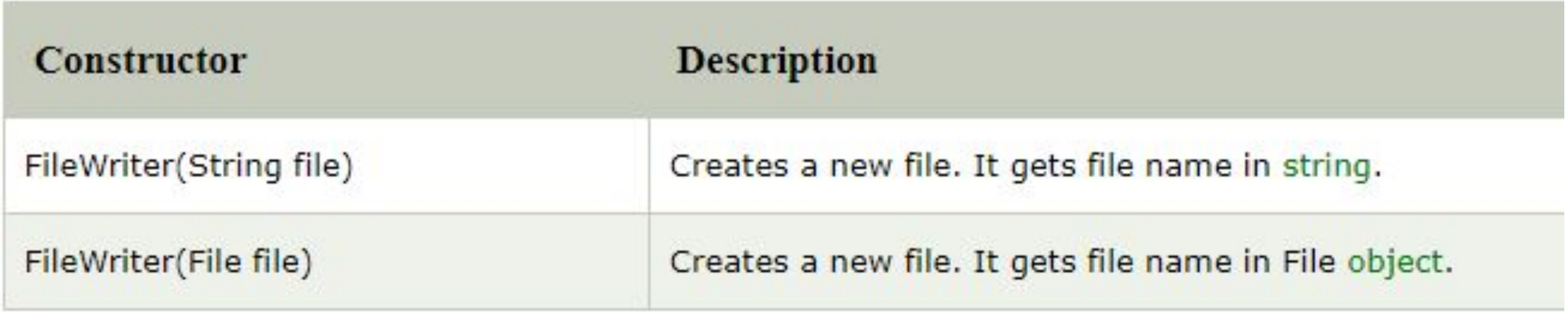
File Writer and File Reader

File Writer Class

Java FileWriter class is used to write character-oriented data to a file.

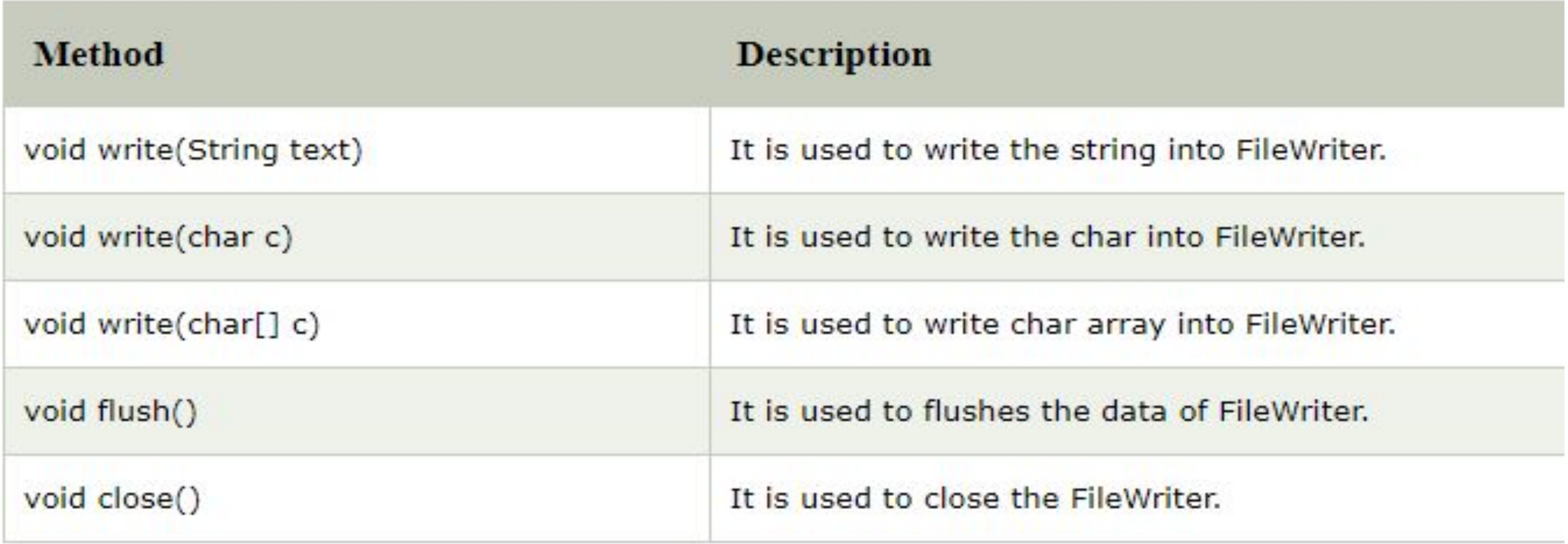
File Writer will create a file if the file does not exist.

Constructor



Constructor	Description
FileWriter(String file)	Creates a new file. It gets file name in string.
FileWriter(File file)	Creates a new file. It gets file name in File object.

Methods In File Writer



Method	Description
void write(String text)	It is used to write the string into FileWriter.
void write(char c)	It is used to write the char into FileWriter.
void write(char[] c)	It is used to write char array into FileWriter.
void flush()	It is used to flushes the data of FileWriter.
void close()	It is used to close the FileWriter.

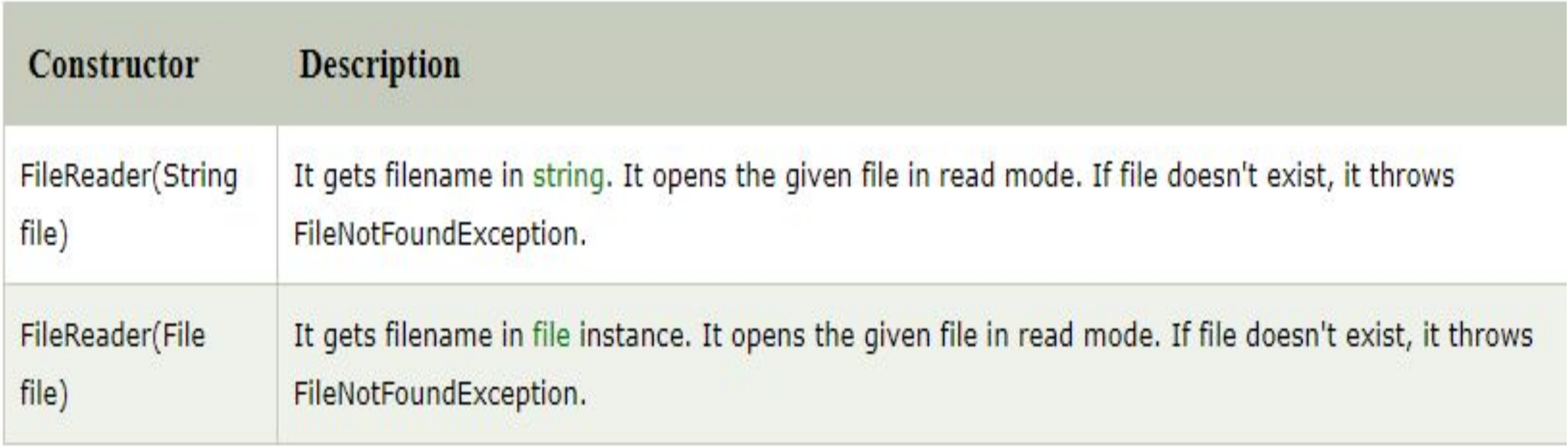
File Reader

File Reader Class

Java FileReader class is used to read data from the file.

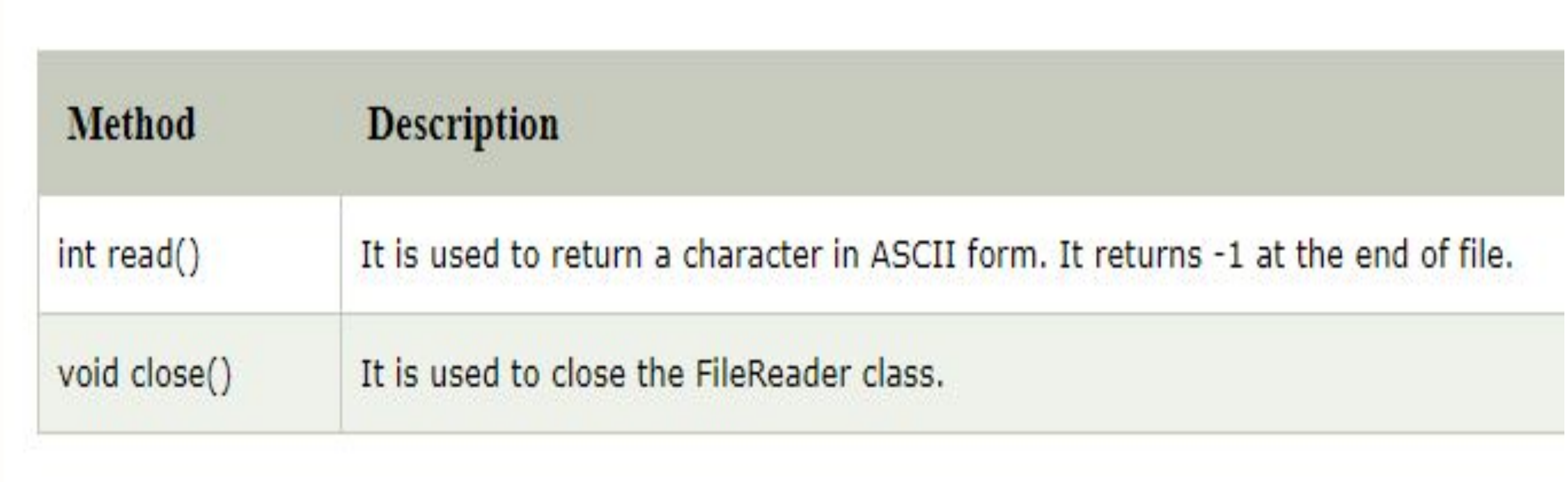
It is character-oriented class which is used for file handling in java.

Constructor



Constructor	Description
FileReader(String file)	It gets filename in string. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.
FileReader(File file)	It gets filename in file instance. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.

Methods



Method	Description
int read()	It is used to return a character in ASCII form. It returns -1 at the end of file.
void close()	It is used to close the FileReader class.

Java I/O

File Handling in JAVA

Java I/O (Input and Output) is used *to process the input* and *produce the output*.

Java uses the concept of a stream to make I/O operation fast. The java.io package contains all the classes required for input and output operations.

Stream

What is Stream?

- A stream is a sequence of data
- In Java, a stream is composed of bytes.
- It's called a stream because it is like a stream of water that continues to flow.

In Java, 3 streams are created for us automatically. All these streams are attached with the console.

1) **System.out:** standard output stream

2) **System.in:** standard input stream

3) **System.err:** standard error stream

Output Stream

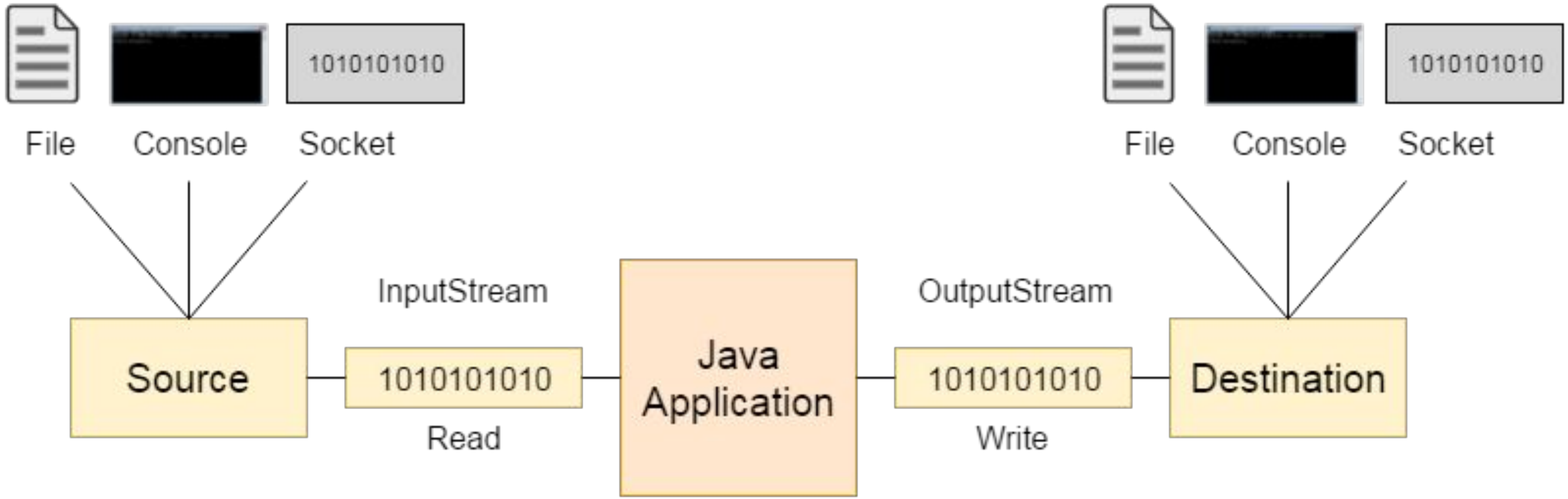
Java application uses an output stream to write data to a destination.

it may be a file, an array, peripheral device or socket.

InputStream

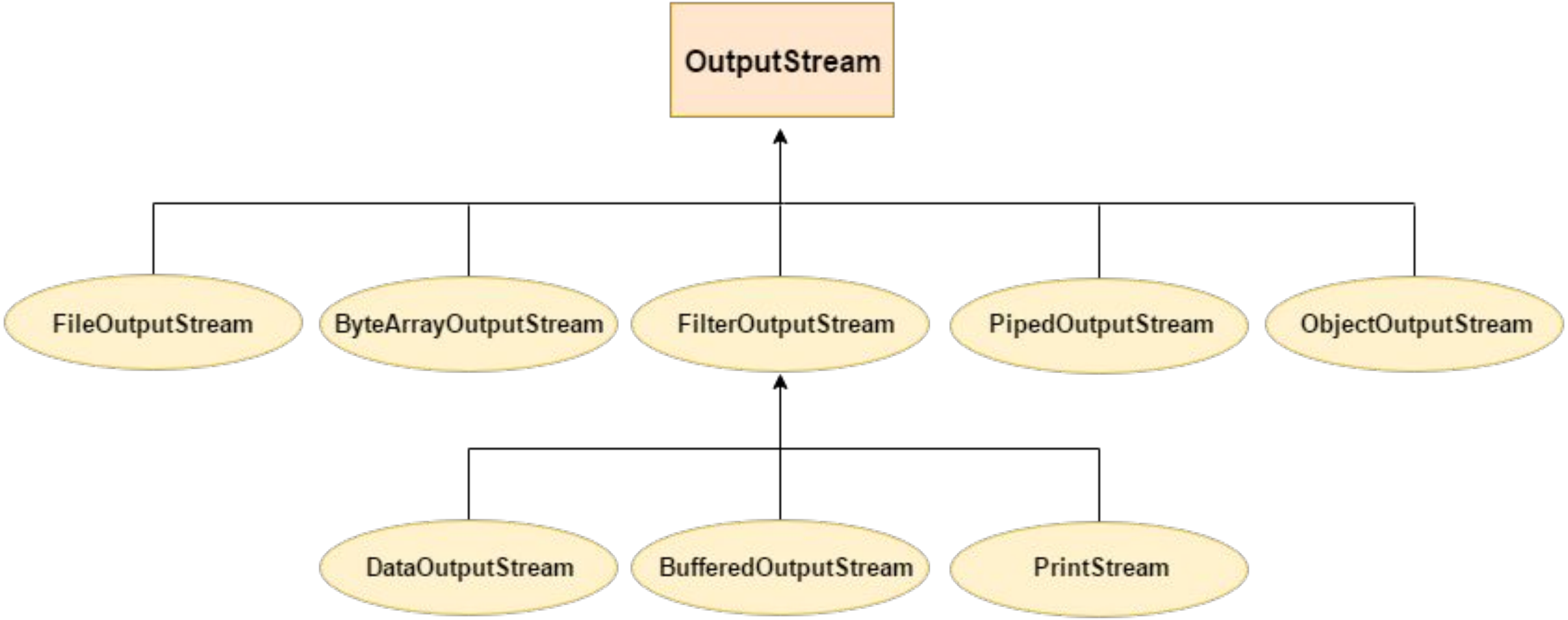
Java application uses an input stream to read data from a source

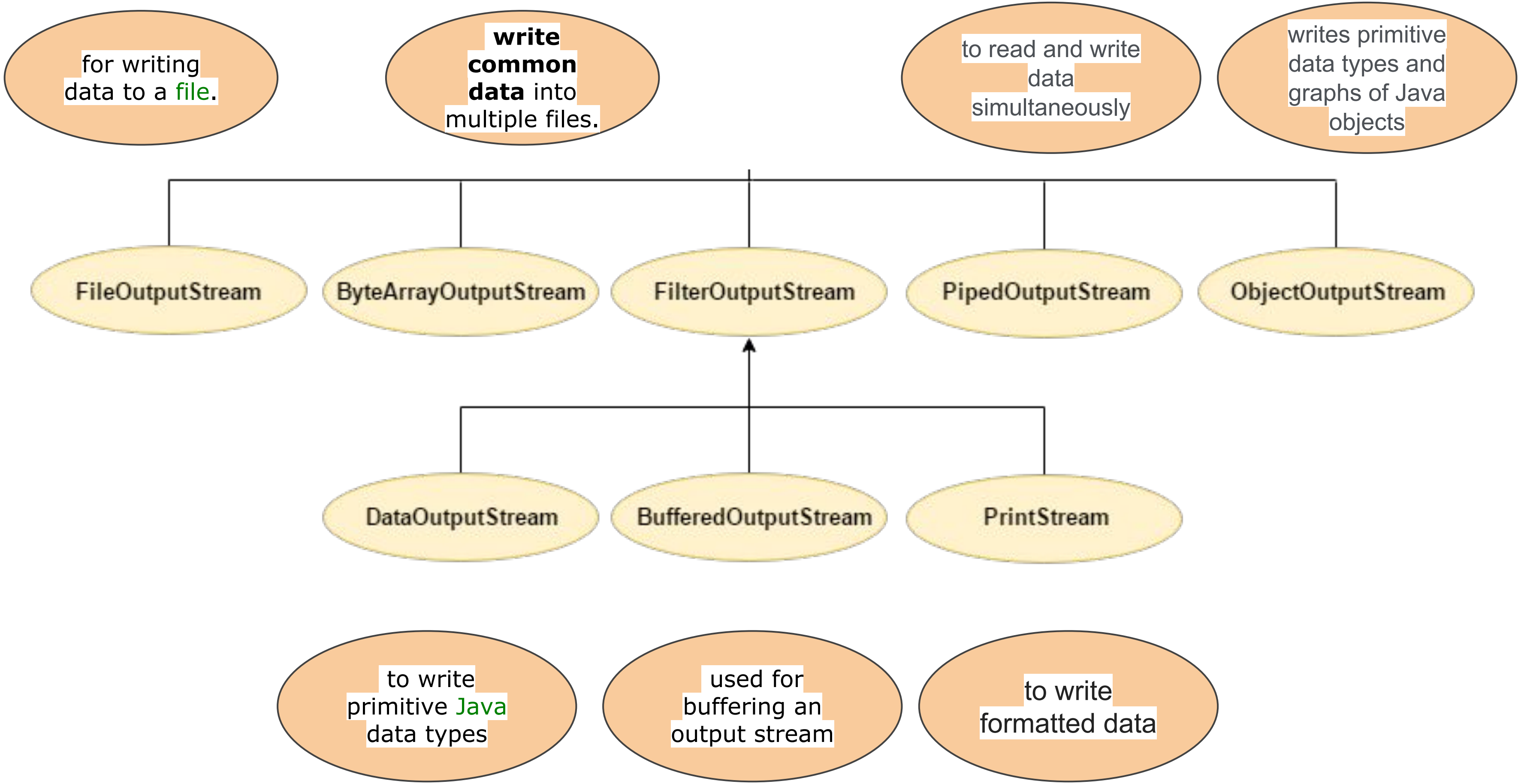
It may be a file, an array, peripheral device or socket.


1010101010
File
Console
Socket
1010101010
File
Console
Socket
InputStream
OutputStream
Source
1010101010
Read
Java Application
1010101010
Write
Destination

Output Stream Class

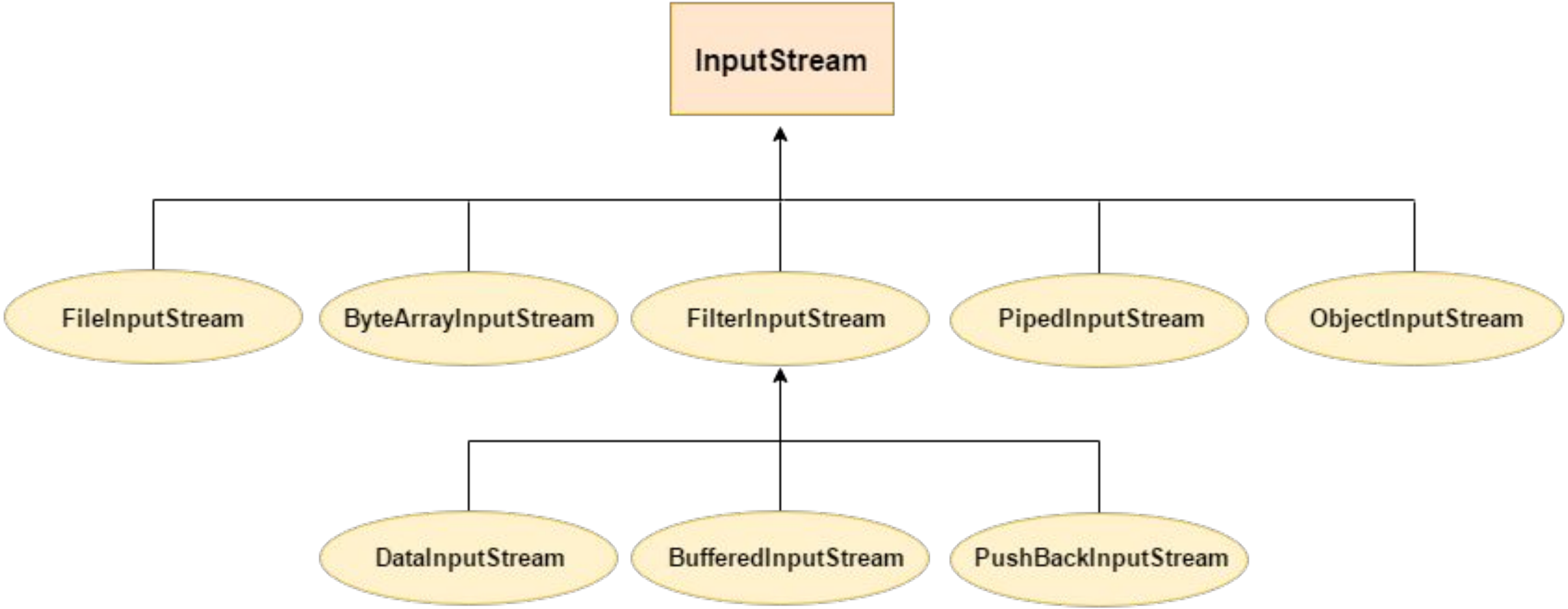
- OutputStream class is an abstract class.
- It is the superclass of all classes representing an output stream of bytes.




for writing data to a file.
write common data into multiple files.
to read and write data simultaneously
writes primitive data types and graphs of Java objects
FileOutputStream
ByteArrayOutputStream
FilterOutputStream
PipedOutputStream
ObjectOutputStream
DataOutputStream
BufferedOutputStream
PrintStream
to write primitive Java data types
used for buffering an output stream
to write formatted data

InputStream Class

- InputStream class is an abstract class.
- It is the superclass of all classes representing an input stream of bytes



File Output Stream Class

- Java FileOutputStream is an output stream used for writing data to a file.
- If you have to write primitive values into a file, use FileOutputStream class. You can write byte-oriented as well as character-oriented data through FileOutputStream class

Example >>>>

File Input Stream Class

- FileInputStream class obtains input bytes from a file.
- used for reading byte-oriented data (streams of raw bytes) such as image data, audio, video etc.
- read character-stream data.

Example >>>>

Serializable in Java

serialization where an object can be represented as a sequence of bytes

Classes **ObjectInputStream** and **ObjectOutputStream** are high-level streams that contain the methods for serializing and deserializing an object.

Example >>>>