# **Java I/O**

Java I/O is used to process input and produce output. An input stream reads the file and stores the data in process. The output stream reads from the process and writes to the destination file.

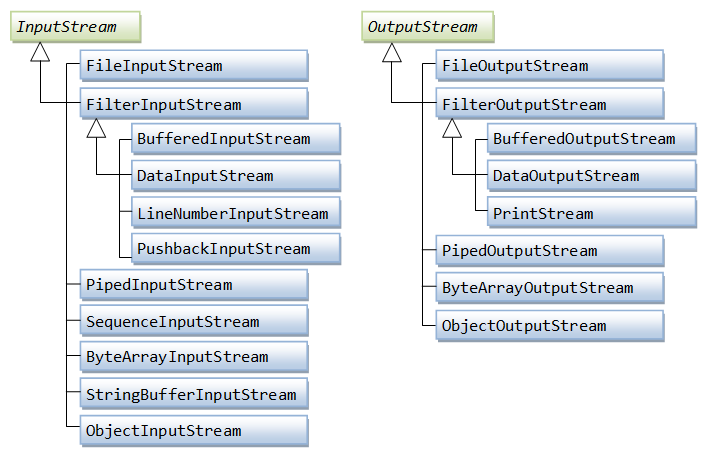
There are two types of streams – Byte Streams and Character Streams

[Byte streams](http://way2java.com/io/byte-streams-vs-character-streams/) were introduced with JDK 1.0 and operate on the files containing ASCII characters. We know Java supports other language characters also known as Unicode characters. To read the files containing Unicode characters, the designers introduced character streams with JDK 1.1. As ASCII is a subset of Unicode, for the files of English characters, we can go with either byte streams or [character streams](http://way2java.com/io/byte-streams-and-character-streams/).

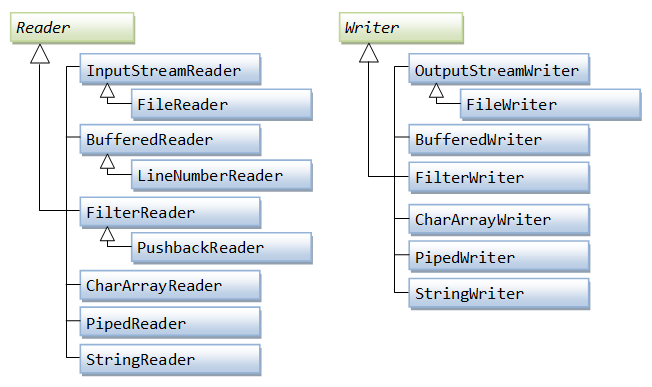
All Byte streams can be divided into two types – InputStream and OutputStream

All Character streams can be divided into two types – Reader and Writer.

Byte I/O Streams:



Character I/O Streams:



**File/Directory Operations in Java:**

File/Directory Operations are operations other than reading or writing to files like checking if a file exists in directory, creating a new directory, creating a new file etc.

These operations are provided by java.io.File class.

These methods can be divided into following.

1. Methods that query the file system
2. Methods that modify the file system

Some of the methods that query the file system are

1. canRead()
2. canWrite()
3. exists()
4. isDirectory()
5. isFile()
6. isHidden()
7. getAbsolutePath()
8. lastModified()
9. length()
10. listFiles()
11. listRoots()

Some of the methods that modify the file system are

1. createNewFile()
2. mkdir()
3. renameTo()
4. delete()
5. deletOnExit()
6. setReadOnly()
7. setLastModified()