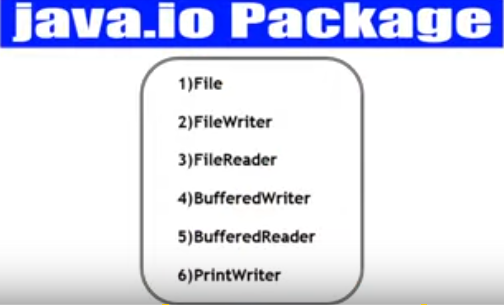
**File Handling in Java->**

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

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

We can perform **file handling in java** by Java I/O API.

Java.io. package.



Stream->

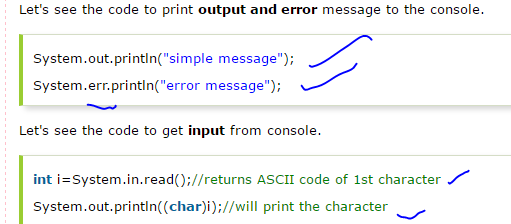
A stream is a sequence of data.In Java a stream is composed of bytes. It's called a stream because it is like a stream of water that continues to flow.

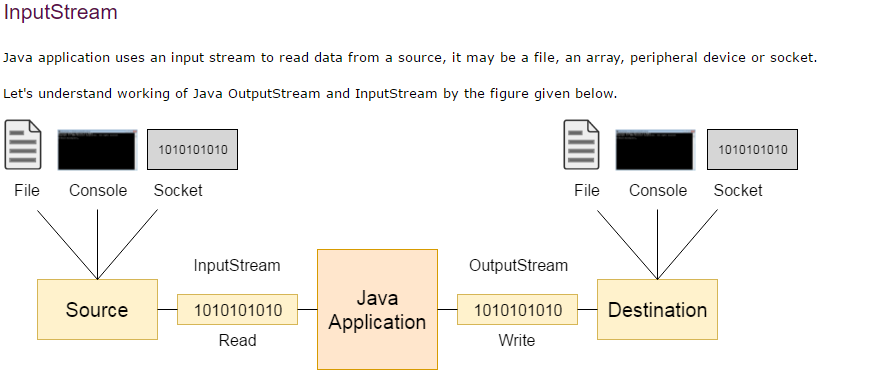
In java, 3 streams are created for us automatically. All these streams are attached with console.

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

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

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





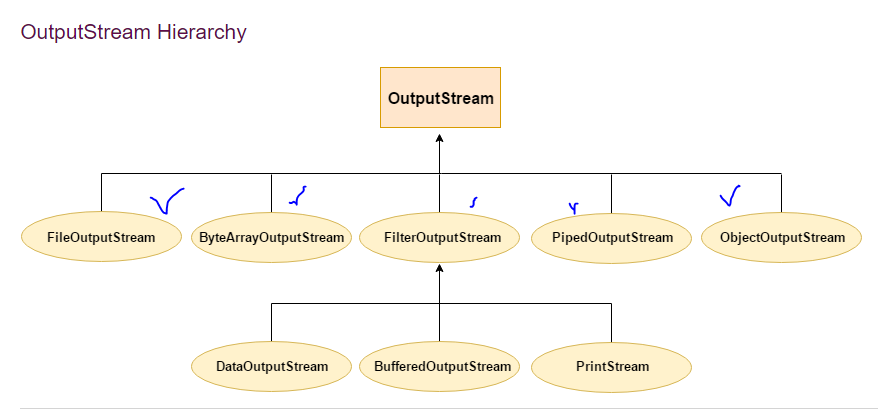
\*We can take source as text file..or cmd or network (socket..)

\*data send by the input steram.

\*ouput stream.

\*Destination can be file ,console,socket.

\*



## InputStream class->

## \* InputStream class is an abstract class.

## \* It is the super class of all classes representing an input stream of bytes.

## 

## 

# **Java FileOutputStream 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. But, for character-oriented data, it is preferred to use FileWriter than FileOutStream.

Output stream class ->

\*void write(int ) throw ioexception

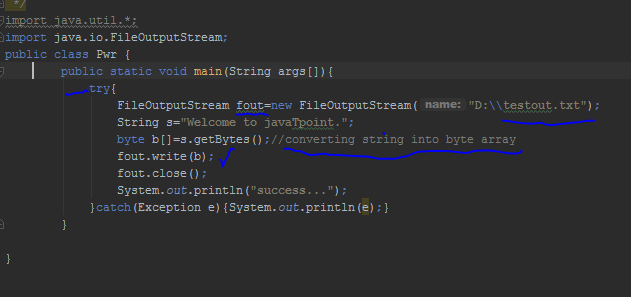
To read byte.

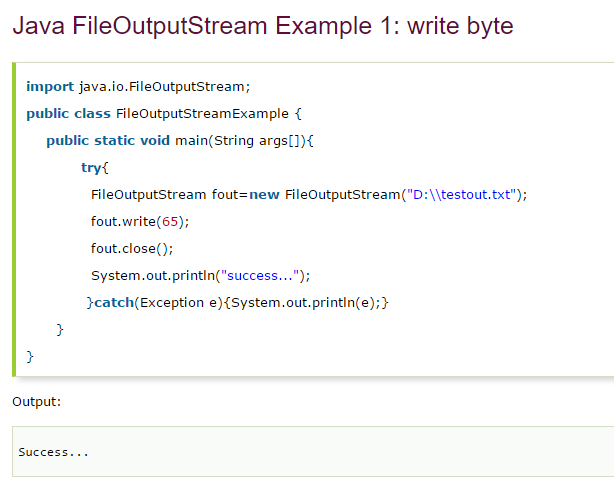
\*void write(byte[])throw io exceptio.

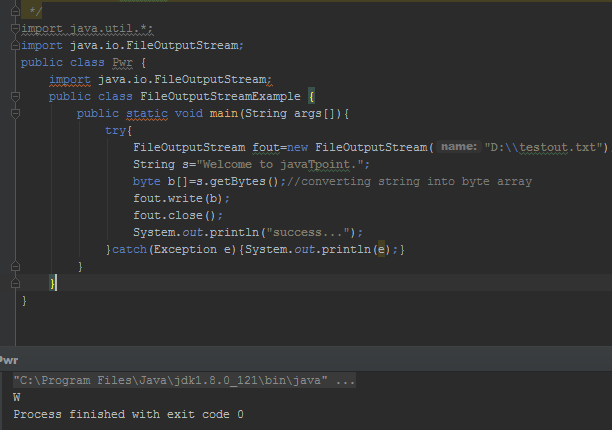
Read Array

\*void flush( )throw io execption.

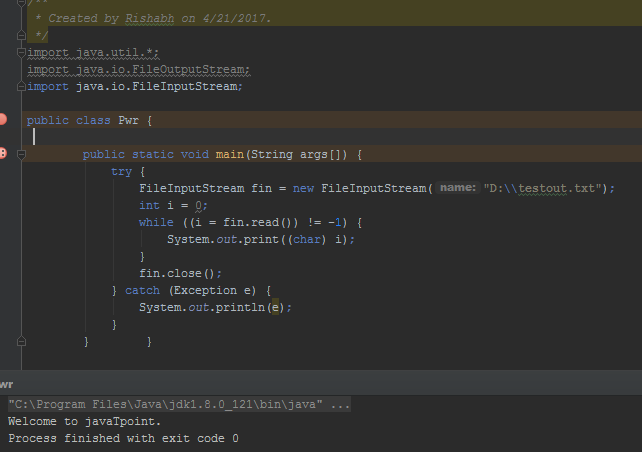
## FileOutputStream class declaration











# **Java BufferedOutputStream Class->**

Java BufferedOutputStream class is used for buffering an output stream.

It internally uses buffer to store data. It adds more efficiency than to write data directly into a stream. So, it makes the performance fast.

For adding the buffer in an OutputStream, use the BufferedOutputStream class. Let's see the syntax for adding the buffer in an OutputStream:



