Java provides us with a variety of Input and Output Streams. In the case of the FileInputStream and FileOutputStream class, we are referring to the file input source and the output destination. Our output destination may be a file, i/o device or another program (<https://www.tutorialspoint.com/what-is-the-use-of-fileinputstream-and-fileoutputstream-in-classes-in-java>).

This class is most commonly used to read the contents of file with raw bytes and this may include files with images.

The way this class is instantiated is the same as many classes and two of the most common ways are:

FileInputStream inputStream = new FileInputStream("file\_path");

or,

File file = new File("file\_path");

FileInputStream inputStream = new FileInputStream(file);

It is common to use the read method with this class to properly read the data from the file. This read method can take no parameters, as well as bytes and even bytes arrays.

In order to properly display the output we must use the FileOutputStream class as well. Much like our FileInputStream class we instantiate it the same way. An example could be:

FileOutputStream outputStream = new FileOutputStream("file\_path");

or,

File file = new File("file\_path");

FileOutputStream outputStream = new FileOutputStream (file);

We can also use the FileDescriptor class to help us write the file.

FileDescriptor descriptor = new FileDescriptor();

FileOutputStream outputStream = new FileOutputStream(descriptor);

Much like the FIleInputStream we have some methods available to us as well. Most commonly we use the write method. This method works the same as our read method and can take in no parameters, it can take in bytes and bytes arrays.