Java I/O

I/O in Java.

Input & Output are one of the most fundamental operations.

java.io is the package that contains class to perform I/O in Java.

java.io also supports file I/O.

What I/O have we used as of now?

For input : Scanner.

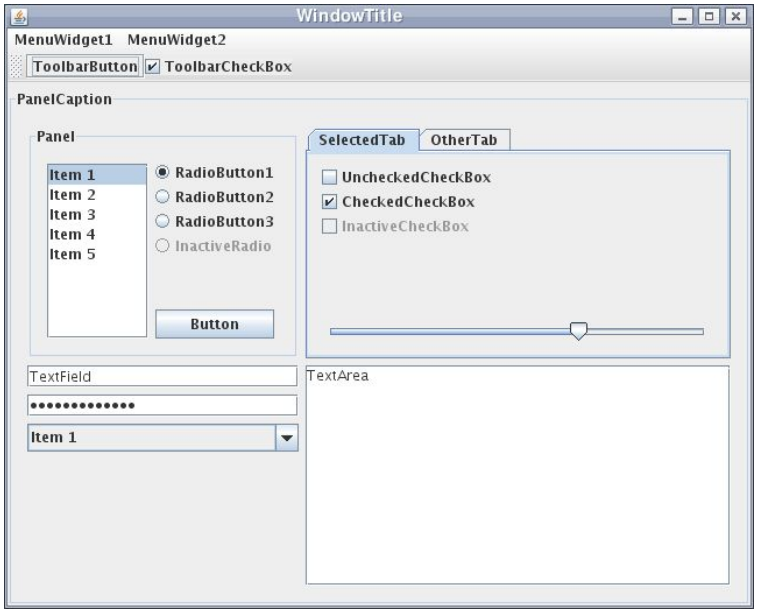
For output: println().

Above methods are not suitable for real-world applications, because most applications are not text/console based.

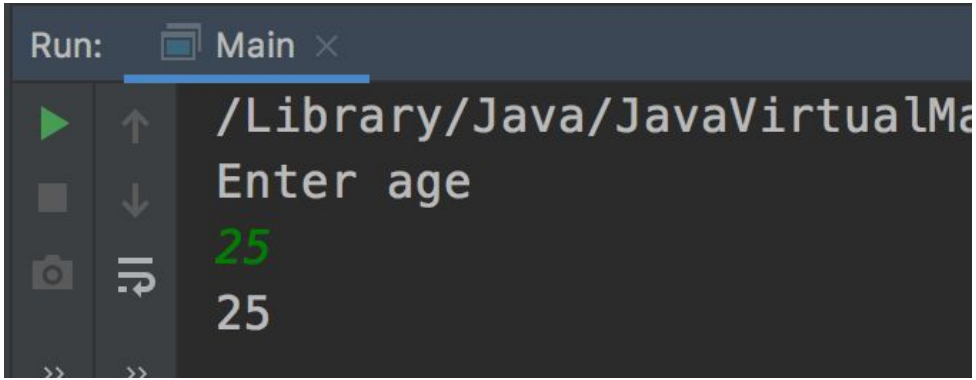
Real world applications are graphics based and have a (GUI) Graphical User Interface.

Graphical User Interface V/S console I/O

I/O With Graphical User Interface



Console based I/O



How Java I/O works

Java uses streams to perform I/O.

A stream is a sequence of data, imagine it like a stream of water that flows.

Stream in Java is like data flowing through it.

Where does that data actually flow?

To and from physical devices, like keyboard, router etc.

Types of streams

Java has two types of streams: byte stream & character stream.

Byte stream: Used for handling input and output of bytes, reading binary data.

Character stream: Used for handling input and output of characters, they use Unicode.

Both byte stream & character stream have a set of classes which help us perform I/O.

Byte stream classes

Byte stream classes can be further classified into InputStream and OutputStream.

These classes further have subclasses that can handle I/O.

The most important methods implemented by these classes are read() and write().

Read and write are used to perform reading and writing of bytes of data.

Character stream classes

They can be classified as Reader & Writer classes.

These two classes handle Unicode character streams.

The most important methods implemented by these classes are read() and write().

They perform reading and writing of characters of data.

System class

Present in the java.lang package which is automatically imported.

This package contains the system class.

System class has methods and variables.

It contains three predefined stream variables, in,our,err.

System.out: Refers to standard output stream. i.e the console

System.in: Refers to standard input stream. i.e they keyboard.

System.err: Refers to standard error stream which is also the console.

System.in: Object of type InputStream, System.out: Object of OutputStream.