

4.1: INPUT / OUTPUT BASICS

Java 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.
- ✓ These streams support all the types of objects, data-types, characters, files etc to fully execute the I/O operations.
- ✓ **The java.io package** contains all the classes required for input and output operations.



➤ Java Input

There are several ways to get input from the user in Java. To get input by using Scanner object, import Scanner class using:

```
import java.util.Scanner;
```

Then, we will create an object of Scanner class which will be used to get input from the user.

```
Scanner input = new Scanner (System.in);  
int number = input.nextInt();
```

Example : Get Integer Input From the User

```
import java.util.Scanner;
class Input{
public static void main(String[] args){
Scanner input =newScanner (System.in);
System.out.print ("Enter an integer: ");
int number =input.nextInt ();
System.out.println ("You entered"+ number);
}
}
```

Output

Enter an integer: 23

You entered 23

➤ Java Output

Simply use `System.out.println()`, `System.out.print()` or `System.out.printf()` to send output to standard output (screen). `System` is a class and `out` is a public static field which accepts output data.

Example to output a line:

```
class Test
{
    public static void main(String[] args)
    {
        System.out.println("Java programming is interesting.");
    }
}
```

Output:

Java programming is interesting.

What's the difference between println (), print () and printf ()?

- print () - prints string inside the quotes.
- println () - prints string inside the quotes similar like print() method. Then the cursor moves to the beginning of the next line.
- printf () - it provides string formatting.

4.1.1: STREAMS

A **Stream** is a sequence of data or it is an abstraction that either produces or consumes information. In other simple words it is a flow of data from which you can read or write data to it. It's called a stream because it's like a stream of water that continues to flow.

➤ **PREDEFINED STREAMS:**

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

1) System.in: This is used to feed the data to user's program and usually a keyboard is used as standard input stream and represented as **System.in**. - It is an object of type InputStream.

2) System.out: This is used to output the data produced by the user's program and usually a computer screen is used to standard output stream and represented as **System.out**. - It is an object of type PrintStream

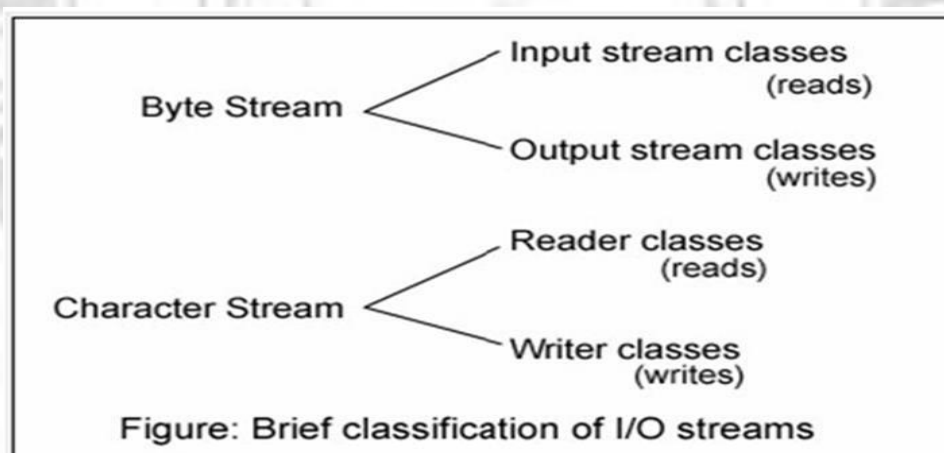
3) System.err: This is used to output the error data produced by the user's program and usually a computer screen is used to standard error stream and represented as **System.err**. - It is an object of type PrintStream



➤ **TYPES OF STREAMS:**

1. Byte Stream – Byte Streams provide a convenient means of handling input and output in terms of bytes. Byte streams are used when reading or writing binary data.

2. Character Stream – Character streams provide a convenient means of handling input or output in terms of characters. In some cases, character streams are more efficient than byte streams.



Some important Byte stream classes:

Stream class	Description
BufferedInputStream	Used for Buffered Input Stream.
BufferedOutputStream	Used for Buffered Output Stream.
DataInputStream	Contains method for reading java standard datatype
DataOutputStream	An output stream that contain method for writing java standard data type
FileInputStream	Input stream that reads from a file
FileOutputStream	Output stream that write to a file.
InputStream	Abstract class that describe stream input.
OutputStream	Abstract class that describe stream output.
PrintStream	Output Stream that contain print() and println() method

Some important Charcter stream classes.

Stream class	Description
BufferedReader	Handles buffered input stream.
BufferedWriter	Handles buffered output stream.
FileReader	Input stream that reads from file.
FileWriter	Output stream that writes to file.
InputStreamReader	Input stream that translate byte to character
OutputStreamReader	Output stream that translate character to byte.
PrintWriter	Output Stream that contain print() and println() method.
Reader	Abstract class that define character stream input
Writer	Abstract class that define character stream output