

JAVA PROGRAMMING COURSE

DATA INPUT AND OUTPUT IN JAVA



By the expert: Ubaldo Acosta



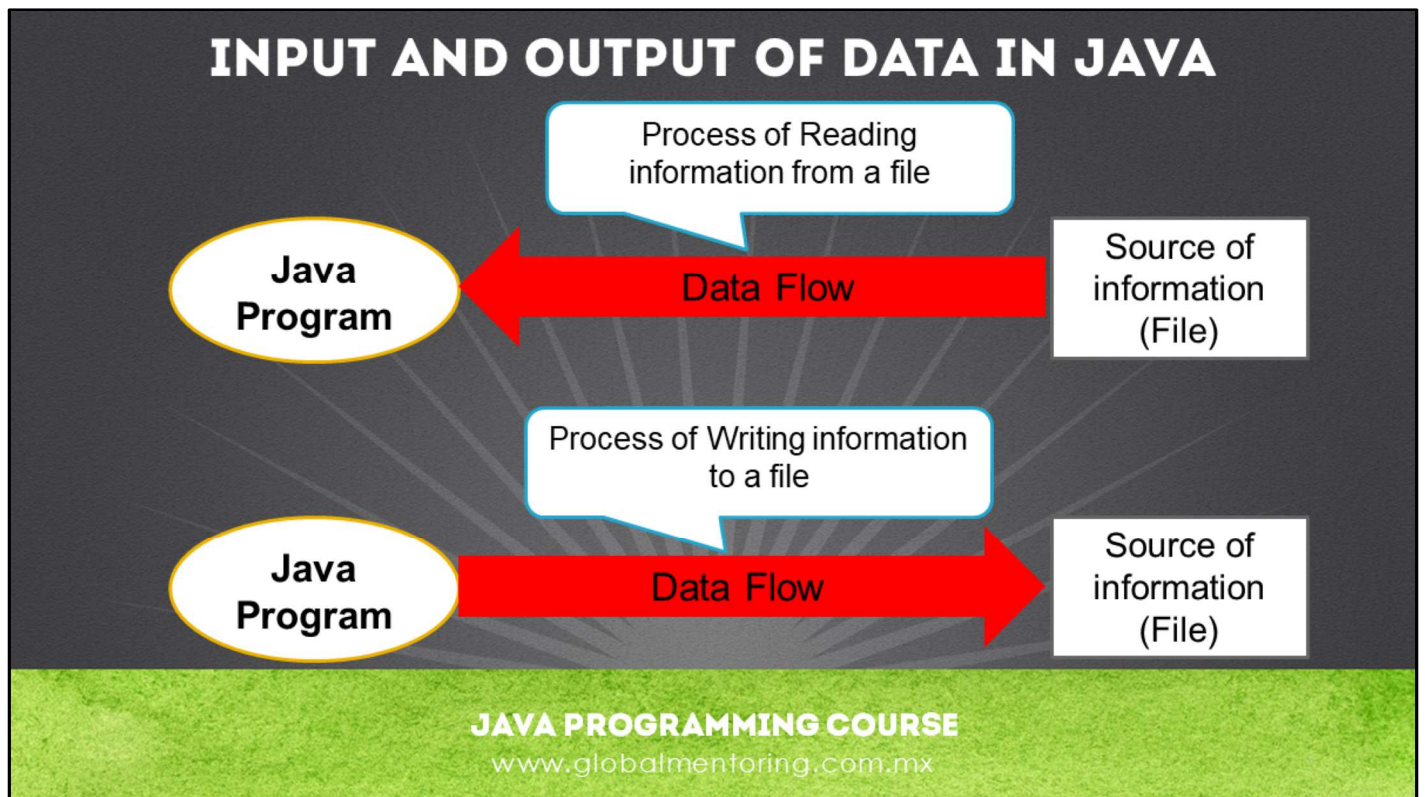
JAVA PROGRAMMING COURSE

www.globalmentoring.com.mx

Hello, Ubaldo Acosta greets you again. I hope you're ready to start with this lesson ..

We are going to study the topic of input and output of data in Java.

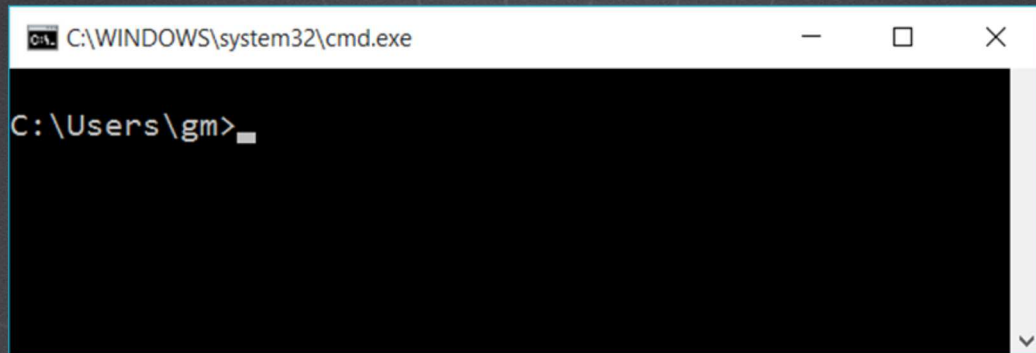
Are you ready? Come on!



In this lesson we are going to study the topic of input and output of data. In Java, we have a set of classes to read and write information, both standard output (console), and files, flows, and other types of information inputs and outputs.

The figure shows us the reading and writing processes in Java. In this lesson we will review some of the classes that will allow us to write and read console information, also known as standard data output or input.

CONSOLE: DATA INPUT AND OUTPUT



JAVA PROGRAMMING COURSE
www.globalmentoring.com.mx

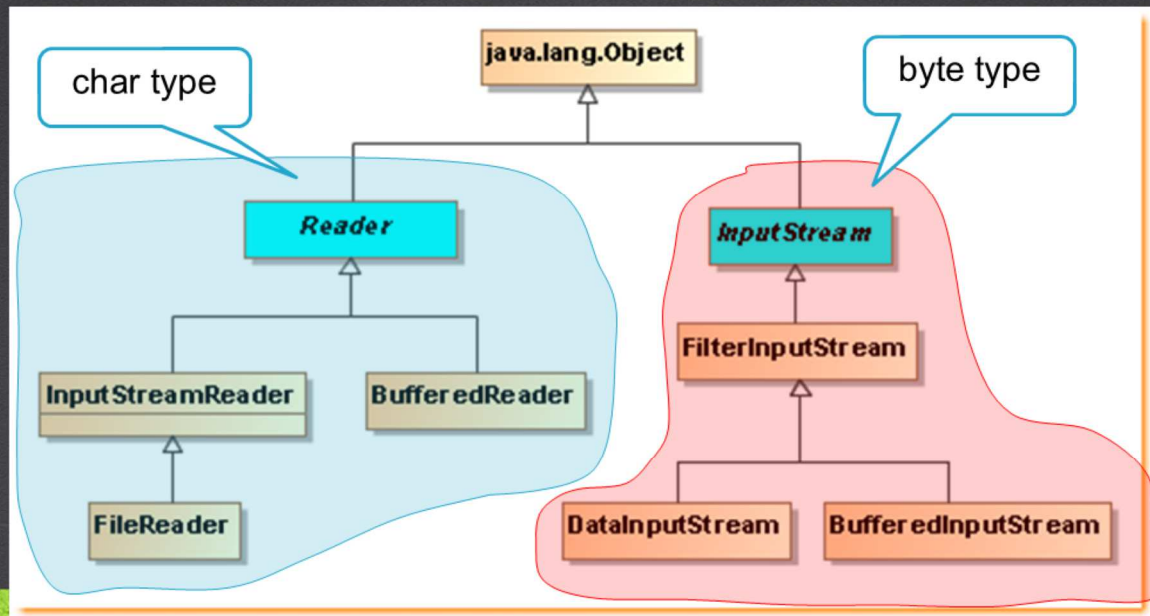
There are 3 variables to interact with the operating system console, also known as standard input / output flows.

System.in: Read from the standard input, and it is of the `InputStream` type (eg read keyboard data).

System.out: Write to the standard output, and it is of the `PrintStream` type (eg it shows data on the screen).

System.err: Write to the standard error output, and it is of the `PrintStream` type (eg it shows error data on the screen).

HOW TO READ A FILE IN JAVA



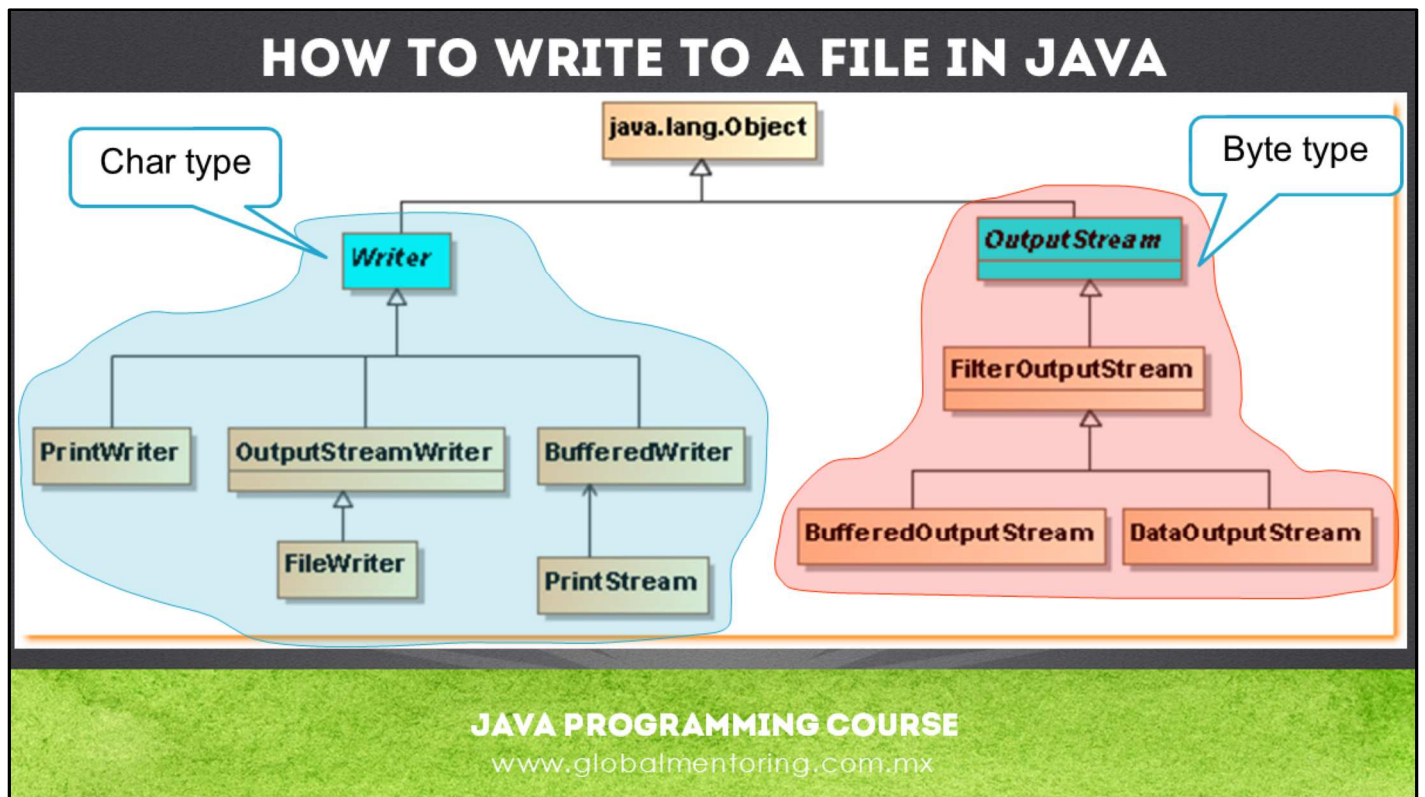
JAVA PROGRAMMING COURSE

www.globalmentoring.com.mx

In the `java.io` package there are two large classifications, classes that allow to handle char types and others of type byte. The figure shows in a summarized way the classes involved in the reading process.

The classes on the left side are for flows of char type and the classes on the right side are for flows of byte type.

We will see later examples of some of these classes to read data in Java.



Now we will see the classes of the Output API in Java. The figure shows in a summarized way the classes involved in the writing process.

The classes on the left side are for flows of char type and the classes on the right side are for flows of byte type.

We will see later examples of some of these classes to write data in Java.

DIFFERENCE BETWEEN BYTE AND CHAR FLOWS

Conversions of flow types



JAVA PROGRAMMING COURSE
www.globalmentoring.com.mx

Let's see now the difference between using data types byte or char in data flows, which has to do directly with the type of data we are working on.

Streams are used to process one byte at a time, they are recommended to process binary files, such as Word, Excel files, or any format that is more complex than a simple text file.

Reader's / Writer's type classes are used to process one character (char) at a time, they are recommended to process text based on some character set such as ASCII or Unicode files.

The classes that have a buffer are used to process more than one character or byte at a time.

READING STANDARD ENTRY IN JAVA

Using the Java Data Entry API:

```
public class DataEntry {  
  
    public static void main(String args[]) {  
        String capture;  
        InputStreamReader input = new InputStreamReader(System.in);  
        BufferedReader bInput = new BufferedReader(input);  
        try {  
            System.out.println("Enter a data: ");  
            capture = bInput.readLine();  
            while (capture != null) {  
                System.out.println("Data entered: " + capture);  
                capture = bInput.readLine();  
            }  
        } catch (IOException e) {  
            e.printStackTrace(System.out);  
        }  
    }  
}
```

www.globalmentoring.com.mx

Let's see now how to read data from the standard input, this we will achieve by means of the variable `System.in`, which reads data from the standard input.

The `InputStreamReader` class accesses the standard console, but can only read one character at a time. Therefore we need another class so that we can read more data than just one character at a time.

The `BufferedReader` class allows us to capture a complete line of data, instead of just one character from the standard input and with it we can more easily access more information from our standard input.

Let's create an example of reading data from standard input.

SCANNER CLASS TO READ DATA

Using the Scanner class in Java:

```
public class DataEntryScanner {  
  
    public static void main(String[] args) {  
        String capture = null;  
        Scanner scan = new Scanner(System.in);  
        System.out.println("Enter a data: ");  
        capture = scan.nextLine();  
        while (capture != null) {  
            System.out.println("Data entered: " + capture);  
            capture = scan.nextLine();  
        }  
    }  
}
```

JAVA PROGRAMMING COURSE

www.globalmentoring.com.mx

In addition to the API that we have already reviewed, in Java the `java.util.Scanner` class was introduced to simplify the process of reading data. Therefore we can observe an example in which we can observe the basic use of the Scanner class, which receives in its constructor the input flow, be it the standard console, a file or another type of input.

Once this object has been created, it contains methods to simplify the reading of the data entry, such as the method `nextLine`, `nextInt`, `nextFloat`, and several others, which as its name tells us allows us to read either a new line , a new integer, a new floating type respectively.

This class, as we observed, allows us to simplify the data entry process, and we will see an exercise later in order to understand in more detail how to use this Scanner class.

ONLINE COURSE

JAVA PROGRAMMING

By: Eng. Ubaldo Acosta



JAVA PROGRAMMING COURSE

www.globalmentoring.com.mx