# How to Take Input from User in Java

Java brings various Streams with its I/O package that helps the user to perform all the input-output operations. These streams support all the types of objects, data types, characters, files, etc to fully execute the I/O operations. There are**two ways** by which we can take input from the user or from a file

* Buffered Reader Class
* Scanner Class

**1. Buffered Reader**

It is a simple class that is used to read a sequence of characters. It has a simple function that reads a character another read which reads, an array of characters, and a readLine() function which reads a line.

Input Stream Reader() is a function that converts the input stream of bytes into a stream of characters so that it can be read as Buffered Reader expects astream of characters.

Buffered Reader can throw checked Exceptions

**2. Scanner**

It is an advanced version of BufferedReader which was added in later versions of Java. The scanner can read formatted input. It has different functions for different types of data types.

* The scanner is much easier to read as we don’t have to write throws as there is no exception thrown by it.
* It was added in later versions of Java
* It contains predefined functions to read an Integer, Character, and other data types as well.

**Syntax:**Scanner

Scanner scn = new Scanner(System.in);

**Syntax: Importing Scanner Class:** To use the Scanner we need to import the Scanner Class

import java.util.Scanner;

**I**

Scanner scn = new Scanner(System.in);

**Syntax: Importing Scanner Class:** To use the Scanner we need to import the Scanner Class

import java.util.Scanner;

**Differences Between BufferedReader and Scanner**

* BufferedReader is a very basic way to read the input generally used to read the stream of characters. It gives an edge over Scanner as it is faster than Scanner because Scanner does lots of post-processing for parsing the input; as seen in nextInt(), nextFloat()
* BufferedReader is more flexible as we can specify the size of stream input to be read. (In general, it is there that BufferedReader reads larger input than Scanner)
* These two factors come into play when we are reading larger input. In general, the Scanner Class serves the input.
* BufferedReader is preferred as it is synchronized. While dealing with multiple threads it is preferred.
* For decent input, easy readability. The Scanner is preferred over BufferedReader.