



READ

FileInputStream

Java FileInputStream class obtains input bytes from a file. It is used for reading byte-oriented data (streams of raw bytes) such as image data, audio, video etc. You can also read character-stream data. But, for reading streams of characters, it is recommended to use FileReader class.

This method reads text from a character-input stream. It does buffering for efficient reading of characters, arrays, and lines. The buffer size may be specified, or the default size may be used. The default is large enough for most purposes.

In general, each read request made of a Reader causes a corresponding read request to be made of the underlying character or byte stream. It is therefore advisable to wrap a BufferedReader around any Reader whose read() operations may be costly, such as FileReaders and InputStreamReaders. For example,

```
BufferedReader in = new BufferedReader(Reader in, int size);
```

FileReader

Convenience class for reading character files. The constructors of this class assume that the default character encoding and the default byte-buffer size are appropriate.

WRITE

FileOutputStream

Java FileOutputStream is an output stream used for writing data to a file. If you have to write primitive values into a file, use FileOutputStream class. You can write byte-oriented as well as character-oriented data through FileOutputStream class. But, for character-oriented data, it is preferred to use FileWriter than FileOutputStream.

BufferReader

used to provide buffering for Writer instances. It makes the performance fast. It inherits Writer class. The buffering characters are used for providing the efficient writing of single arrays, characters, and strings.

PrintWriter

the implementation of Writer class. it is used to print the formatted representation of objects to the text-output stream.

BufferReader

CODING IMPLEMENTATION

```
File f = new File("data/someFile.txt");
if(f.exists()) {
    System.out.println("File already exists!");
}
else {
    try(PrintWriter pw = new PrintWriter(f)) {
        pw.println("Budi");
        pw.print("Ani");
        pw.println("Candra\n");
        pw.println("Dedi");
    }
}
```

```
System.out.println("Please type something and press
Enter.");
BufferedReader reader = new BufferedReader(new
InputStreamReader(System.in));
try {
    System.out.println(reader.readLine().split(" ").length);
} catch(IOException e) {
    System.out.println("An error occurred: " +
e.getMessage());
}
```