**Character Streams**

Java Byte streams are used to perform input and output of 8-bit bytes, whereas Java Character streams are used to perform input and output for 16-bit unicode. Though there are many classes related to character streams but the most frequently used classes are, FileReader and FileWriter. Though internally FileReader uses FileInputStream and FileWriter uses FileOutputStream but here the major difference is that FileReader reads two bytes at a time and FileWriter writes two bytes at a time.

We can re-write the above example, which makes the use of these two classes to copy an input file (having unicode characters) into an output file −

Example

*import java.io.\*;*

*public class CopyFile {*

*public static void main(String args[]) throws IOException {*

*FileReader in = null;*

*FileWriter out = null;*

*try {*

*in = new FileReader("input.txt");*

*out = new FileWriter("output.txt");*

*int c;*

*while ((c = in.read()) != -1) {*

*out.write(c);*

*}*

*}finally {*

*if (in != null) {*

*in.close();*

*}*

*if (out != null) {*

*out.close();*

*}*

*}*

*}*

*}*

Now let's have a file input.txt with the following content −

This is test for copy file.

As a next step, compile the above program and execute it, which will result in creating output.txt file with the same content as we have in input.txt. So let's put the above code in CopyFile.java file and do the following −

$javac CopyFile.java

$java CopyFile