**How to read file in Java using BufferedReader**

[**JAVA I/O**](http://beginnersbook.com/category/java-io/)

In this tutorial we will see two ways to read a file using BufferedReader.

Method 1: Using **[readLine()](http://docs.oracle.com/javase/7/docs/api/java/io/BufferedReader.html" \l "readLine()" \t "_blank)** method of BufferedReader class.

public String readLine() throws IOException

It reads a line of text.

Method 2: Using [**read()**](http://docs.oracle.com/javase/7/docs/api/java/io/BufferedReader.html#read()) method

public int read() throws IOException

It reads a character of text. Since it returns an integer value, it needs to be explicitly cast as char for reading the content of file.

**Complete example**

Here I have two txt files myfile.txt and myfile2.txt. In order to demonstrate both the ways to read file. I’m reading first file usingreadLine() method while the second file is being read using read()method.

package beginnersbook.com;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class ReadFileDemo {

public static void main(String[] args) {

BufferedReader br = null;

BufferedReader br2 = null;

try{

br = new BufferedReader(new FileReader("B:\\myfile.txt"));

//One way of reading the file

System.out.println("Reading the file using readLine() method:");

String contentLine = br.readLine();

while (contentLine != null) {

System.out.println(contentLine);

contentLine = br.readLine();

}

br2 = new BufferedReader(new FileReader("B:\\myfile2.txt"));

//Second way of reading the file

System.out.println("Reading the file using read() method:");

int num=0;

char ch;

while((num=br2.read()) != -1)

{

ch=(char)num;

System.out.print(ch);

}

}

catch (IOException ioe)

{

ioe.printStackTrace();

}

finally

{

try {

if (br != null)

br.close();

if (br2 != null)

br2.close();

}

catch (IOException ioe)

{

System.out.println("Error in closing the BufferedReader");

}

}

}

}