

# Lectura de Ficheros:

Vamos en primer lugar a leer el contenido del siguiente fichero (llamado 'fichero\_leer.txt') e imprimir por pantalla lo que leemos:

```
Linea 1  
Linea 2  
Linea 3  
Linea 4  
Linea 5  
Linea 6  
Linea 7  
Linea 8  
Linea 9  
Linea 10
```

En primer lugar tenemos que decir la ruta o directorio donde se encuentra el fichero que queremos leer. Si el fichero lo tenemos en la carpeta de nuestro proyecto (como es el caso del proyecto que estamos realizando) solo debemos de poner el nombre del fichero, sino hay que especificar la ruta donde se encuentra el fichero (ejem -> Windows: c:\carpeta\.....\fichero.txt, Linux o Unix: \var\www\.....\fichero.txt). Esto lo hacemos de la siguiente manera: `File fichero = new File("fichero_leer.txt");`

Una vez que tenemos el fichero que queremos leer, nos creamos un objeto de la clase "Scanner" e iremos leyendo linea a linea el fichero. A continuación vemos el programa que nos leera linea a linea el fichero e imprimirá por pantalla linea a linea el contenido del fichero:

```

import java.io.File;
import java.util.Scanner;

public class LecturaFicheros {

    public static void main(String[] args) {

        // Fichero del que queremos leer
        File fichero = new File("fichero_leer.txt");
        Scanner s = null;

        try {
            // Leemos el contenido del fichero
            System.out.println("... Leemos el contenido del fichero ...");
            s = new Scanner(fichero);

            // Leemos linea a linea el fichero
            while (s.hasNextLine()) {
                String linea = s.nextLine();    // Guardamos la linea en un String
                System.out.println(linea);      // Imprimimos la linea
            }

        } catch (Exception ex) {
            System.out.println("Mensaje: " + ex.getMessage());
        } finally {
            // Cerramos el fichero tanto si la lectura ha sido correcta o no
            try {
                if (s != null)
                    s.close();
            } catch (Exception ex2) {
                System.out.println("Mensaje 2: " + ex2.getMessage());
            }
        }
    }
}

```

En el código vemos que nuestro objeto 's' de la clase "Scanner" va posicionandose linea a linea en el fichero y en el bucle "while" le vamos pidiendo la siguiente linea si es que existe.

Vemos también en el código como tenemos que definir varias excepciones, por si ocurre alguna de ellas, el programa no pare la ejecución. Por ejemplo vemos al final del código (en el "finally") como independientemente de si ha habido o no error en la lectura del fichero, nos lo cierre.

En definitiva aquí os hemos presentado el código para que podáis leer un fichero de texto

(en este caso un 'txt') linea a linea.

## Escritura de Ficheros:

Para la escritura de ficheros, vamos a presentar dos formas de hacerlo, una para escribir un fichero sin una codificación determinada y otra para escribirlo con una determinada codificación (en este caso en UTF-8). En estos ejemplos vamos a escribir en el fichero (linea a linea) cada uno de los elementos del siguiente array: `String[] lineas = { "Uno", "Dos", "Tres", "Cuatro", "Cinco", "Seis", "Siete", "..."};`

La forma más sencilla de hacer esta escritura (haciendo un foreach del array) es la que presentamos a continuación: `import java.io.FileWriter;`

```
public class EscrituraFicheros {
```

```
    public static void main(String[] args) {

        String[] lineas = { "Uno", "Dos", "Tres", "Cuatro", "Cinco", "Seis", "Siete", "..."};

        /** FORMA 1 DE ESCRITURA **/
        FileWriter fichero = null;
        try {

            fichero = new FileWriter("fichero_escritura.txt");

            // Escribimos linea a linea en el fichero
            for (String linea : lineas) {
                fichero.write(linea + "\n");
            }

            fichero.close();

        } catch (Exception ex) {
            System.out.println("Mensaje de la excepción: " + ex.getMessage());
        }
    }
}
```

Como vemos en este programa, solo utilizamos un objeto de la clase "FileWriter" en el que le pasamos el nombre del fichero. Si el fichero no existe nos lo crea. En realidad no solo hay que pasarle el nombre del fichero, sino que hay que pasarle la ruta donde queremos guardar el fichero más el nombre del fichero, aunque si lo hacemos como en el programa, nos guardará el fichero en la carpeta del proyecto.

Otra cosa muy importante es que abrimos el fichero y vamos escribiendo en el poco a poco (o elemento a elemento del array) y al final cuando ya hemos escrito todo cerramos el fichero (método "close()"). En realidad a la hora de escribir en un fichero, lo que hacemos es pasarle un determinado contenido (en este caso un String) y este lo escribe en el fichero. Se recomienda que la escritura la hagáis poco a poco ya que si almacenais en una variable de tipo String el contenido a escribir y luego lo escribís de golpe, os puede dar una excepción de exceso de memoria si lo que tenéis que escribir en el fichero es muy grande; por tanto hacerlo como se os indica aquí y no tendreis ese problema.

Como resultado de la ejecución de este programa, tendréis un nuevo fichero llamado "fichero\_escritura.txt" con el siguiente contenido:

```
Uno  
Dos  
Tres  
Cuatro  
Cinco  
Seis  
Siete  
...
```

La segunda forma de escribir es un fichero con una codificación determinada es la que presentamos a continuación:

```

import java.io.BufferedWriter;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.io.UnsupportedEncodingException;
import java.io.Writer;

public class EscrituraFicheros {

    public static void main(String[] args) {

        String[] lineas = { "Uno", "Dos", "Tres", "Cuatro", "Cinco", "Seis", "Siete", "..."};

        /** FORMA 2 DE ESCRITURA. Con el fichero codificado en UTF-8 */
        Writer out = null;
        try {
            out = new BufferedWriter(new OutputStreamWriter(new FileOutputStream("fichero_escritura2.txt"), "UTF-8"));

            // Escribimos linea a linea en el fichero
            for (String linea : lineas) {
                try {
                    out.write(linea+"\n");
                } catch (IOException ex) {
                    System.out.println("Mensaje excepcion escritura: " + ex.getMessage());
                }
            }

        } catch (UnsupportedEncodingException | FileNotFoundException ex2) {
            System.out.println("Mensaje error 2: " + ex2.getMessage());
        } finally {
            try {
                out.close();
            } catch (IOException ex3) {
                System.out.println("Mensaje error cierre fichero: " + ex3.getMessage());
            }
        }
    }
}

```

Esta forma es un poco más compleja y hay que tratar diferentes excepciones pero nos

permite escribir en el fichero con el formato que le especificamos.

Como resultado tenemos el mismo contenido que en el fichero anterior pero en este caso con el nombre "fichero\_escritura2.txt".