

ARCHIVOS DE TEXTO EN JAVA

Un archivo es un grupo de datos estructurados que son almacenados en algún medio de almacenamiento y pueden ser usados por las aplicaciones (programas). La forma en que una computadora organiza, da nombre, almacena y manipula los archivos se denomina “**sistema de archivos**” y suele depender del Sistema Operativo y del medio de almacenamiento (disco duro, disco externo, memorias USB, etc)

La característica principal de los archivos es el Nombre y su extensión,

El nombre es la identificación del archivo y su extensión indica el tipo de archivo al que corresponde, la extensión generalmente es de tres caracteres, por ejemplo:

Nombre.**doc** → es que un archivo tipo documento (de Microsoft Word)

Nombre.**mp3** → es un archivo de música

Nombre.**xls** → es un archivo de hoja de cálculo (Microsoft Excel)

Nombre.**txt** → es un archivo de texto

Lectura y escritura de archivos de texto

Clase File

En el paquete **java.io** se encuentra la clase File pensada para poder realizar operaciones de información sobre archivos. No proporciona métodos de acceso a los archivos, sino operaciones a nivel de sistema de archivos (listado de archivos, crear carpetas, borrar ficheros, cambiar nombre, etc)

Clase FileReader

La clase FileReader permite leer caracteres, se usa para leer el contenido de un archivo de texto.

Clase FileWriter

La clase FileWriter permite escribir caracteres, se usa para escribir texto en un archivo de texto.

Clase BufferedReader

La clase BufferedReader es muy usada para leer archivos de texto plano que contenga información que el programa puede usar.

Clase PrintWriter

La clase PrintWriter se utiliza para escribir en archivos de texto.

La apertura de archivos y su posterior lectura pueden lanzar **excepciones** que debemos capturar. Por ello, la apertura y la lectura deben meterse en un bloque **try-catch**. Además, el fichero hay que cerrarlo cuando términos.

Ejemplo de lectura de un archivo de texto

En el siguiente ejemplo se visualizan todos los datos que se encuentran en el archivo de texto “**numeros.txt**” que se encuentra en la unidad “**D:**” en la carpeta “**archivos**”.

```
import java.io.*;
public class archivos1 {

    public static void main(String[] args) {
        File f=new File("d:\\archivos\\numeros.txt"); // f → carga el archivo
        FileReader fr;
        try
        {
            fr=new FileReader(f); // fr → crea el flujo hacia f
            BufferedReader br=new BufferedReader(fr); // br → permite la lectura
            String linea=br.readLine(); // → linea toma el valor de la primera linea
            while(linea!=null) // mientras haya datos en el archivo
            {
                linea=br.readLine();
                System.out.println(linea);
            }
            fr.close(); // cierra el archivo
        }
        catch(IOException e)
        {
            System.out.println("error"); // si no se pudo accede al archivo
        }
    }
}
```

Ejemplo de Escritura de un archivo de texto

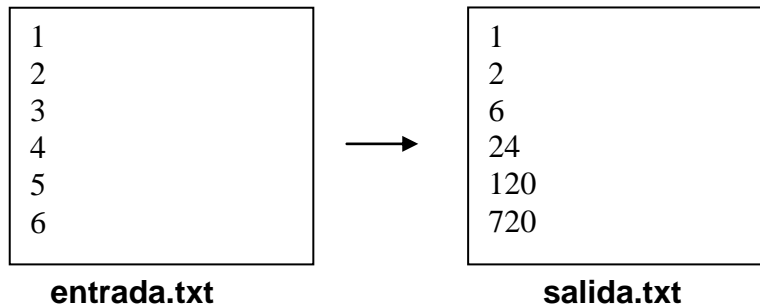
En el siguiente ejemplo se escriben en el archivo de texto “**numeros.txt**” que se encuentra en la unidad “**D:**” en la carpeta “**archivos**”. Los números de 1 al 50

```
import java.io.*;
public class archivos2 {
    public static void main(String[] args) {
        File f=new File("d:\\archivos\\numeros.txt"); // f → carga el archivo
        FileWriter fw;
        PrintWriter pw;
        try
        {
            fw=new FileWriter(f); // fw → crea el flujo hacia f
            pw=new PrintWriter(fw); // pw → permite la escritura en el archivo
            for(int i=1;i<=20;i++)
            {
                pw.println(i); // se introduce el valor de i en el archivo
            }
            fw.close(); // se cierra el flujo del archivo
        }
    }
}
```

```
        catch(IOException e)
        {
            System.out.println("error");
        }
    }
}
```

Ejemplo de entrada de datos desde un archivo y escritura a otro archivo

Programa que lee el archivo “entrada.txt” donde se encuentran números y escribe en el archivo “salida.txt” donde escribe sus correspondientes factoriales



```
import java.io.*;
public class Archivos {
    public static void main(String[] args) {
        File in=new File("d:\\entrada.txt");
        File out=new File("d:\\salida.txt");
        FileReader fr; // fr → flujo de lectura
        FileWriter fw; // fw → flujo de escritura

        BufferedReader br; // br → lector
        PrintWriter pr; // pr → escritor

        try
        {
            fr=new FileReader(in);
            fw=new FileWriter(out);
            br=new BufferedReader(fr);
            pr=new PrintWriter(fw);

            String linea=br.readLine(); //lee la primera linea de entrada

            while(linea!=null)
            {
                int num=Integer.parseInt(linea); //convierte de string a int
                int fac=factorial(num); //calcula el factorial
                pr.println(fac); //escribe en el archivo de salida
                linea=br.readLine(); //lee una linea de la entrada
            }

            fr.close();
            fw.close();

        }
        catch(IOException e)
        {
        }
```

```
        System.out.print("error");
    }
}

// funcion que devuelve el factorial de un numero

static int factorial(int n)
{
    int f=1;
    for(int i=1;i<=n;i++)
        f=f*i;
    return f;
}
}
```

Sitios recomendados para complementar el tema

- http://chuwiki.chuidiang.org/index.php?title=Lectura_y_Escritura_de_Ficheros_en_Java
- <http://fisiwikipedia.wikispaces.com/file/view/Manejo+de+Archivos+en+Java.pdf>
- acm.asoc.fi.upm.es/static/events/2/java_dia3_fich.pdf