# Lectura y almacenamiento en archivos

Hay varias formas de almacenar en archivo la información de los objetos. Una manera es a través del uso de archivos planos, con datos separados por un carácter especial como (;) o (,):

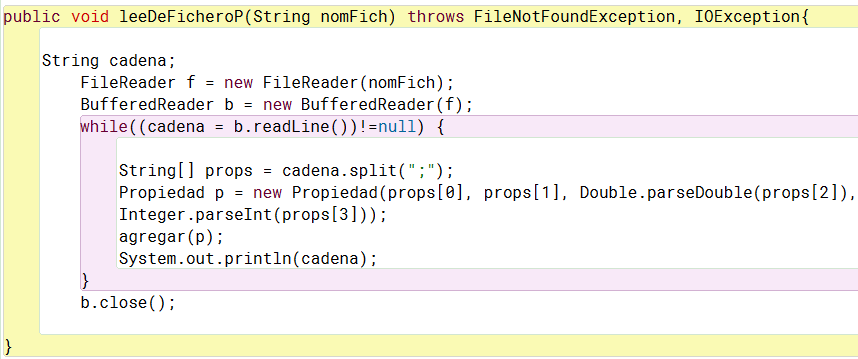
* <http://chuwiki.chuidiang.org/index.php?title=Lectura_y_Escritura_de_Ficheros_en_Java>
* <https://geekytheory.com/como-crear-y-modificar-ficheros-con-java>

Otra manera es almacenando los objetos como un flujo de bits:

* <http://www.sc.ehu.es/sbweb/fisica/cursoJava/fundamentos/archivos/objetos.htm>

## Lectura de archivos planos

La lectura de archivo permite a través del FileReader obtener un archivo, y el BufferedReader permite realizar la lectura de cada línea a través del método readLine(), pero también se puede realizar la lectura de caracteres. Después de leer los archivos es importante hacer un Split para separarlos por el carácter definido y cargarlos en una estructura dinámica, que para este caso es la List de Propiedades:



## Escritura de archivos planos

La escritura en archivos planos a diferencia del flujo de bits, almacena los datos en texto plano en un archivo especifico. Para este caso se separan por el sigo (;). Aca se deben tener en cuenta dos clases, el FileWriter que permite obtener la instancia del archivo y el PrinWriter que permite realizar la escritura de datos sobre el archivo seleccionado, es como un System.out.println, pero no sobre consola sino sobre el archivo seleccionado.



## Lectura de objetos como flujo de bits

La lectura de información de un objeto corresponde a la lectura desde un archivo de un flujo de bits de la instancia actual de un objeto, es decir es una foto de toda la información y del estado actual del objeto almacenado en un flujo de bits.



Cabe resaltar que para poder almacenar la información y que la misma al ser recuperada se pueda reconocer, las clases que se almacenen deben extender de la interface Serializable, así de esta forma podremos obtener el dato o el tipo de clase exacto de los objetos almacenados en el flujo de bits.



Se debe tener en cuenta que si se modifica y se vuelve a compilar la clase ya las clases no son compatibles con el archivo.

## Escritura de objetos como flujo de bits

La escritura de objetos consiste en tomar toda la información de una instancia convertirla en un flujo de bits y almacenarla en un archivo que puede ser leído posteriormente. El siguiente código permite tomar la clase actual y almacenarla en un archivo definido.

