

# Práctica 4

## Uso de fichero: lectura y escritura

Grado Ingeniería Telemática  
GSyC, Universidad Rey Juan Carlos

### 1. Introducción

En esta práctica haremos uso del código desarrollado en las prácticas anteriores. El objetivo en este caso es ofrecer la funcionalidad para que toda la información que teníamos recopilada en la **Práctica 2** pueda guardarse y/o cargarse en/desde el disco duro. De este modo, nuestra aplicación de *gestión de tutorías* tendría ya una funcionalidad muy cercana a un uso real.

Para conseguir ese objetivo, necesitamos hacer uso de ficheros (que se guardan en disco duro) que nos sirvan de soporte para almacenar toda la información. Recordemos que, sin estos, los datos cargados hasta ahora en nuestra aplicación están almacenados en memoria RAM, que es volátil; esto es, se pierden una vez cerremos la aplicación (o se interrumpa la alimentación eléctrica).

Pues bien, para poder manejar estos ficheros, nos apoyaremos en las bibliotecas `IO` y `NIO` que nos ofrece Java para todo lo referente a la gestión de entrada/salida (*input/output*).

### 2. Formato

Dado que vamos a leer/escribir datos en ficheros, estos han de tener un determinado formato; de lo contrario, pueden producirse errores de *entendimiento* entre lo que maneja nuestra aplicación y lo que maneja los ficheros. Así, se pide que los datos a leer/escribir deben usar el formato que usa Java para los enteros, y UTF-8 para los *Strings*.

Además, para poder hacer correctamente esa lectura y escritura de datos, es necesario establecer un orden y unos elementos separadores; en definitiva, una determinada sintaxis.

### 3. Ejercicios

Con todo lo anterior, podemos concluir que necesitaremos al menos tres ficheros. Así, crea al menos los siguientes ficheros:

- `Alumnos.txt`: donde se guardará la información de todos los alumnos contenidos en el array de alumnos.
- `Profesores.txt`: donde se guardará la información de todos los profesores contenidos en el array de profesores.
- `Peticiones.txt`: donde se guardará la información de todas las peticiones de citas contenidas en la cola de citas.

Estos ficheros han de estar codificados siguiendo el formato anteriormente descrito y la sintaxis que se desee. Un ejemplo de esta puede ser la que vemos a continuación, que podría ser el contenido del fichero `Alumnos.txt`:

```
#DNI;Nombre;Apellido1;Apellido2;TipoAlumno-
```

Nótese el uso de distintos símbolos para *parsear* correctamente los elementos del array de `Alumnos` y proceder a su inserción en el array.

Además de los ficheros, deberás —evidentemente— implementar la funcionalidad necesaria para que toda la información que teníamos recopilada en la **Práctica 2** pueda guardarse y/o cargarse en/desde el disco duro. Los ejemplos facilitados en el Tema 11 pueden servir de ayuda. Dicha implementación estará por duplicado; en una de ellas deberás hacer uso exclusivamente de la librería `IO` y, en la otra, de la librería `NIO`.

## 4. Entrega

Para la entrega de esta práctica, deja todo el código fuente del proyecto y los ficheros de texto que uses en un fichero `io.zip` que, como siempre, habrá que dejar adjunto en la tarea (Moodle).

La implementación basada en la librería `IO` deberá estar en una única clase/fichero de nombre `LibreriaIO.java`; de igual forma, la basada en la librería `NIO`, en un fichero `LibreriaNIO.java`.