

1 INTRODUCCIÓN

Desde el punto de vista informático, un **fichero** es una colección de información que almacenamos en un soporte magnético, óptico, de estado sólido o de otro tipo para poder manipularla en cualquier momento. Los ficheros tienen un nombre y se ubican en directorios o carpetas. Su principal ventaja es que los datos que se guardan en ellos no son volátiles.

Un **fichero** está formado por un conjunto de **registros** y cada registro por un conjunto de **campos** relacionados, que representan los datos de la entidad de la cual queremos almacenar información

1.1 Clasificación de ficheros

Según el tipo de contenido:

- **Ficheros de caracteres (o de texto):** son aquellos creados exclusivamente con caracteres (código ASCII), por lo que pueden ser creados y visualizados por cualquier editor de texto.
- **Ficheros binarios (o de bytes):** son aquellos que no contienen caracteres reconocibles, sino que los bytes que contienen representan otra información como imágenes, música o vídeo. Estos ficheros solo pueden ser abiertos por aplicaciones concretas que conozcan cómo están organizados los bytes dentro del fichero.

Según el modo de acceso:

- **Secuenciales:** la información se almacena como una secuencia de bytes o caracteres, de manera que para acceder al byte i-ésimo, es necesario pasar antes por todos los anteriores.
- **Aleatorios:** se puede acceder directamente a una posición concreta del fichero, sin necesidad de recorrer los bytes anteriores.

Según la dirección del flujo de datos:

- **De entrada:** el programa lee los datos del archivo.
- **De salida:** el programa escribe datos en el archivo.
- **De entrada/salida:** el programa puede leer o escribir datos en el archivo.

Según la longitud de registro:

- **Longitud variable:** en realidad, en este tipo de archivos no tiene sentido hablar de longitud de registro, podemos considerar cada byte como un registro. También puede suceder que nuestra aplicación conozca el tipo y longitud de cada dato almacenado en el archivo, y lea o escriba los bytes necesarios en cada ocasión. Otro caso es cuando se usa una marca para el final de registro, por ejemplo, en ficheros de texto se usa el carácter de retorno de línea para eso. En estos casos cada registro es de longitud diferente.
- **Longitud constante:** en estos archivos los datos se almacenan en forma de registro de tamaño contante.
- **Mixtos:** en ocasiones pueden crearse archivos que combinen los dos tipos de registros, por ejemplo, dBASE usa registros de longitud constante, pero añade un registro especial de cabecera al principio para definir, entre otras cosas, el tamaño y el tipo de los registros

2 OPERACIONES DE GESTIÓN DE FICHEROS

Las operaciones básicas que se realizan sobre cualquier fichero, independientemente de la forma de acceso al mismo, son las siguientes:

- **Creación del fichero.** El fichero se crea en el disco con un nombre que después se debe usar para acceder a él. Este proceso solo se hace una vez.
- **Apertura del fichero.** Para que un programa pueda operar con un fichero, primero debe realizar su apertura. Para ello, el programa utilizará algún método para identificar el fichero, como usar un descriptor del fichero.
- **Cierre del fichero.** El fichero se debe cerrar cuando el programa no lo vaya a utilizar. Suele ser la última instrucción del programa.
- **Lectura de datos del fichero.** Este proceso consiste en transferir información del fichero a la memoria, a través de variables del programa.
- **Escritura de datos en el fichero.** Este proceso consiste en transferir información de la memoria, por medio de las variables del programa, al fichero.

Una vez abierto, las operaciones típicas que se realizan sobre un fichero son:

- **Altas:** consiste en añadir un nuevo registro al fichero.
- **Bajas:** consiste en eliminar del fichero un registro ya existente.
- **Modificaciones:** consiste en cambiar parte del contenido de un registro. Antes de la modificación, será necesario localizar el registro a modificar dentro del fichero.
- **Consultas:** consiste en buscar en el fichero un registro determinado.

2.1 Clases asociadas con ficheros

En Java, la clase **File** proporciona un conjunto de utilidades relacionadas con ficheros que proporcionan información acerca de los mismos. Un objeto de la clase **File** puede representar el nombre de un fichero o de un directorio que existe en el sistema de ficheros. También se puede usar para crear un nuevo directorio o una trayectoria de directorios completa si ésta no existe.

Para crear un objeto **File**, se puede usar cualquiera de los siguientes constructores:

Constructor	Explicación
File(String directorioYFichero)	Recibe como parámetro el camino completo donde está el fichero junto con el nombre. Por defecto, si no se indica, lo busca en la carpeta del proyecto. <code>directorioYFichero</code> puede ser también el camino a un directorio, sin indicar al final el nombre de ningún fichero. En este caso se crea un directorio.
File(String directorio, String nombreFichero)	Recibe en la instanciación del objeto el camino completo donde está el fichero, como primer parámetro, y el nombre del fichero como segundo parámetro.
File(File directorio, String fichero)	Recibe en la instanciación del objeto un objeto de tipo <code>File</code> , que hace referencia a un directorio, como primer parámetro y el nombre del fichero como segundo parámetro.