

ACCESO A DATOS - UNIDAD 02 – FICHEROS

Para la realización de estas actividades podría ser necesario utilizar clases, métodos u opciones no vistos en el capítulo, por lo que se recomienda consultar la documentación de Java SE 8 (<https://docs.oracle.com/javase/8/docs/api/>)

Nota: Al desarrollar la clase, se debe crear un método `main()` que pruebe suficientemente la funcionalidad de la clase. Si se piden modificaciones de una clase existente, las pruebas deberán incidir sobre la funcionalidad añadida o modificada y sobre cualquier otra funcionalidad que haya podido verse afectada por el cambio.

1. Crea una clase que implemente las operaciones de añadir, recuperar y modificar registros de un fichero con organización secuencial. Los nuevos registros se añadirán siempre al final. El fichero debe tener un campo clave de manera que no se permita que existan dos registros con el mismo valor para el campo clave. Para recuperar un registro se proporcionará el valor de su campo clave. Para modificarlo se proporcionará el valor del campo clave para buscar el registro que se va a modificar, el nombre de un campo y el nuevo valor de ese campo. Ten en cuenta que el hecho de que el registro tenga organización secuencial no significa que no se pueda utilizar acceso aleatorio para él, si esto supone un beneficio para realizar algunas operaciones puedes implementar la clase con una estructura fija de fichero para que sea más sencillo. Por ejemplo: campos DNI y nombre de cliente, campo clave DNI.
2. Basándote en el ejercicio anterior, crea una nueva clase, más genérica, que permita definir la estructura del registro al crear un nuevo fichero, como se hace en un programa de ejemplo para ficheros de acceso aleatorio. Todo funcionará igual, solo que la clase valdrá para cualquier estructura de registro, en lugar de para una estructura fija.
3. Añade un método para borrar registros a la clase anterior. Pero los registros realmente no se borrarán, sino que se marcarán como borrados. Se puede, sencillamente, poner un carácter especial al principio del registro. Cuando se inserte un registro, se insertará, como antes, al final del fichero. Se pueden explorar otras posibilidades para marcar los registros como borrados. Por ejemplo, asignar todo espacios o asignar un valor cero a todos los bytes.
4. Añade a la clase anterior un método para compactar el fichero, es decir, para eliminar los registros marcados como borrados. Habrá que utilizar un fichero temporal para construir el nuevo fichero y, finalmente, sustituir el antiguo por el nuevo.

ACCESO A DATOS - UNIDAD 02 – FICHEROS

5. Crea una clase que implemente un fichero secuencial que contenga registros de longitud fija, compuestos por campos de longitud fija, y que siempre esté ordenado por el valor el campo clave. Cuando se inserta un nuevo registro hay que hacerlo en la posición que deba ocupar en el fichero para preservar su ordenación. Se sugiere generar el nuevo fichero en un fichero temporal y sustituir el antiguo por este al final. En el fichero temporal se irán copiando los registros del fichero hasta llegar a uno con un valor mayor para el campo clave. Entonces, se insertará el nuevo registro en el fichero temporal, y después se copiará el resto de los registros en el fichero temporal. Por último, el fichero actual se sustituirá por el fichero temporal. En esta unidad se han visto programas de ejemplo que utilizan ficheros temporales como ayuda para modificar los contenidos de un fichero.

6. Crea una clase que implemente operaciones para añadir, buscar, borrar y modificar registros de un fichero con organización secuencial indexada. Existirá un único índice para el campo clave del fichero. El índice será un fichero adicional cuyo nombre indique el fichero principal (de datos) y el campo de indexación (que, en este caso y según se ha dicho, es el campo clave). El índice lo creará el constructor de la clase, junto con el fichero principal. Los registros borrados no se deben borrar, sino marcarse como borrados tanto en el fichero principal como en el fichero de índice. En este último se podría poner un número negativo como posición para indicar que el registro está borrado. Los nuevos registros se añadirán siempre al final del fichero principal. La operación de inserción debe recomponer el índice, para ello hay que insertar una nueva entrada en el lugar apropiado. El índice no es más que un tipo especial de fichero secuencial ordenado, y el fichero principal un fichero secuencial no ordenado, por lo que se pueden utilizar las clases desarrolladas en ejercicios anteriores.