

## Ficheros de Acceso Aleatorio

Los archivos de acceso secuencial son útiles para la mayoría de las aplicaciones, pero a veces son necesarios archivos de acceso aleatorio que permiten acceder a su contenido de forma secuencial o aleatoria.

La clase **RandomAccessFile** del paquete `java.io` implementa un archivo de acceso aleatorio. Puede ser usada tanto para la lectura como para la escritura de bytes. Dicha clase dispone de métodos para acceder al contenido de un fichero binario de forma aleatoria y para posicionarnos en una posición determinada del mismo. Esta clase no es parte de la jerarquía **InputStream** / **OutputStream**, ya que su comportamiento es totalmente distinto puesto que se puede avanzar y retroceder dentro de un fichero.

Cuando queramos abrir un fichero de acceso aleatorio tendremos que crear un objeto de tipo **RandomAccessFile** y en el constructor indicaremos la ruta del fichero y el modo de apertura: sólo lectura "**r**", o lectura/escritura "**r/w**".

Hay dos posibilidades para abrir un fichero de acceso aleatorio:

```
RandomAccessFile(String path, String modo); // Con el nombre
del fichero

RandomAccessFile(File fichero, String modo); // Con un objeto
File
```

El parámetro modo puede tener dos valores:

Modo de Acceso	Significado
<b>r</b>	Abre el fichero en modo lectura. El fichero debe existir. Si intentamos escribir se lanza la excepción <b>IOException</b>
<b>rw</b>	Abre el fichero en modo lectura y escritura. Si el fichero no existe se crea.

Ejemplos:

```
RandomAccessFile fichero = new RandomAccessFile("Datos.txt", "rw");
RandomAccessFile fichero = new RandomAccessFile(File archivo, "rw");
```

Todo objeto, instancia de **RandomAccessFile** soporta el concepto de puntero que indica la posición actual dentro del archivo. Cuando en el fichero se crea el puntero se coloca en 0, apuntando al principio del mismo. Las sucesivas llamadas a los métodos `read()` y `write()` ajustan el puntero según la cantidad de bytes leídos o escritos.

**Desplazamiento:** cualquier operación de lectura/escritura de datos se realiza a partir de la posición actual del "**puntero**" del archivo.

Métodos:	Función
<code>long getFilePointer()</code>	Devuelve la posición actual del puntero del archivo.
<code>void seek( long k )</code>	Coloca el puntero del archivo en la posición indicada por <b>k</b> (los archivos empieza en la posición 0).
<code>int skipBytes( int n )</code>	Intenta saltar <b>n</b> bytes desde la posición actual.
<code>long length()</code>	Devuelve la longitud del archivo.
<code>void setLength( long t)</code>	Establece a <b>t</b> el tamaño del archivo.

## Unidad didáctica 1 MANEJO DE FICHEROS.

Métodos Escritura:	Función
<code>void write(byte b[], int ini, int len)</code>	Escribe len caracteres del vector b.
<code>void write(int i )</code>	Escribe la parte baja de i (un byte) en el flujo.
<code>void writeXxx(xxx)</code>	Escribe el tipo indicado en xxx.

Métodos Lectura:	Función
<code>xxx readXxx();</code>	Lee y devuelve el tipo leído
<code>void readFully( byte b[] );</code>	Lee bytes del archivo y los almacena en un vector de bytes.
<code>void readFully( byte b[], int ini, int len )</code>	Lee len bytes del archivo y los almacena en un vector de bytes.
<code>String readUTF()</code>	Lee una cadena codificada con el formato UTF-8.

Tamaño en bytes de los tipos de datos:

Tipo de Dato	Tamaño en bytes
byte	1 byte
boolean	1 byte
short	2 bytes
int	4 bytes
long	4 bytes
float	4 bytes
double	8 bytes
String	2 bytes por cada carácter. Java utiliza UNICODE, y cada carácter de una cadena ocupa 16 bits (2 bytes)

```
package ejemplos10FicherosAleatorios;
```

```
import java.io.*;
```

```
import aaIntroducirDatos.IntroducirDatos;
```

```
import java.io.BufferedReader;
```

```
/* crea un fichero de acceso directo  
 * registro: clave (int), nombre (String 20), edad (int)  
 * la clave es la posición del registro dentro del fichero  
 *  
 * Tamaño del registro 4 + 40 + 4 = 48  
 */
```

```
public class Ej01ClaveEsDireccion {
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
// en un fichero ramdow hay que establecer el tamaño del registro
//recordar que un caracter son dos bytes en archivo
private static final long tamanhoRegistro = 48;

static File f = new File("NombresEdades.DAT");

public static void main(final String[] args) {

    String opcion;

    do{
        System.out.println("1.- Introducir nuevo registro");
        System.out.println("2.- Listado completo");
        System.out.println("3.- Buscar registro");
        System.out.println("4.- Modificar registro");
        System.out.println("5.- Salir");

        opcion = IntroducirDatos.introducirDatos("Introduce
una opcion: ");
        try{
            switch (Integer.parseInt(opcion)){
                case 1:
                    insertarRegistro();
                    break;
                case 2:
                    if(f.exists())
                        ListadoCompleto();
                    else
                        System.out.println("El fichero no
existe. Tiene que insertar datos");
                    break;
                case 3:
                    buscarRegistro();
                    break;
                case 4:
                    modificarRegistro();
                    break;
                case 5:
                    System.exit(0);
                default:
                    System.out.println("Opcion erronea");
            }
        }catch (NumberFormatException e){
            System.out.println("La opcion tiene que ser un
numero");
        }
    }while(!opcion.equals("5"));
}
}
}
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
/*-----  
 * Metodo que inserta los registros en el fichero  
 * el campo clave determina la posición del registro dentro del  
fichero  
 */  
public static void insertarRegistro(){  
  
    RandomAccessFile puntero = null;  
    int clave=0;  
    String nombre="";  
    int edad=0;  
    try{  
        // abriendo archivo, capturando y grabando datos  
        puntero = new RandomAccessFile (f,"rw");  
  
        String respuesta = null;  
  
        do{  
            // teclea los datos  
            clave =  
Integer.parseInt(IntroducirDatos.introducirDatos("Introduce la clave:  
"));  
  
            // comprueba la longitud del nombre tecleado si es  
menor que 20 lo rellena  
            // si es mayor lo acorta  
            nombre =  
IntroducirDatos.introducirDatos("Introduce el nombre: ");  
            if (nombre.length() < 20) {  
                for(int i=nombre.length(); i <20; i++)  
                    nombre=nombre+" ";  
            }  
            else {  
                nombre=nombre.substring(0,20);  
            }  
            edad =  
Integer.parseInt(IntroducirDatos.introducirDatos("Introduce la edad:  
"));  
  
            // grabando el registro en el archivo  
  
            // colocamos el puntero según la clave  
            puntero.seek((clave-1) * tamanoRegistro);  
            puntero.writeInt(clave);  
            puntero.writeUTF(nombre);  
            puntero.writeInt(edad);  
  
            respuesta =  
IntroducirDatos.introducirDatos("Desea continuar S/N");  
        }while(respuesta.equalsIgnoreCase("s"));
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
}
catch(NumberFormatException nfe){
    System.out.println("Error al introducir los datos");
}catch(FileNotFoundException fnf){
    System.out.println("Fichero inexistente");
}catch (IOException ioe) {
    System.out.println(" Error al escribir en el
fichero");
}finally{
    try{
        puntero.close();
    }
    catch(IOException e){
        System.out.println(" Error al cerrar el fichero
");
    }
}
}
//-----

public static void listadoCompleto(){
    int clave=0;
    String nombre="";
    int edad=0;
    RandomAccessFile puntero = null;
    long contadorRegistros = 0;
    try {
        // abriendo archivo, capturando datos
        puntero = new RandomAccessFile (f,"r");

        //calculando el numero de registros
        contadorRegistros = puntero.length()/tamanhoRegistro;

        for (int r=0; r <= contadorRegistros; r++) {
            puntero.seek(r* tamanhoRegistro);
            clave=puntero.readInt();
            nombre = puntero.readUTF();
            edad=puntero.readInt();
            if(clave != 0)
                System.out.println(clave+" "+nombre+"
"+edad);
        }
    }catch EOFException eof){
        System.out.println("Final del fichero ");
    }catch(FileNotFoundException fnf){
        System.out.println("Fichero inexistente");
    }catch (IOException ioe) {
        System.out.println("Error al leer el fichero ");
    }finally{

```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        try{
            puntero.close();
        }
        catch(IOException e){
            System.out.println(" Error al cerrar el fichero
");
        }
    }
}
//-----
public static void buscarRegistro(){
    int clave=0;
    String nombre="";
    int edad=0;
    RandomAccessFile puntero = null;

    try {
        // abriendo archivo, capturando datos
        puntero = new RandomAccessFile (f,"r");

        clave =
Integer.parseInt(IntroducirDatos.introducirDatos("Introducir la clave.
"
            + "< 0 para Finalizar>: "));
        while(clave != 0){
            //colocamos el puntero según la clave
            puntero.seek((clave-1) * tamanhoRegistro);

            //leemos los campos del registro
            clave=puntero.readInt();
            if(clave != 0){
                nombre = puntero.readUTF();
                edad=puntero.readInt();

                //visualizamos los datos
                System.out.println(clave+" "+nombre+"
"+edad);
            }else
                System.out.println("El registro no existe");

            clave =
Integer.parseInt(IntroducirDatos.introducirDatos("Introducir la clave.
< 0 para Finalizar>: "));
        } // fin while
    } catch (NumberFormatException nfe){
        System.out.println("Error al introducir los datos");
    } catch (IOException ioe) {
        System.out.println("Error de posicionamiento o
lectura");
    }
}
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        System.out.println(ioe.getMessage());
    }finally{
        try{
            puntero.close();
        }catch(IOException e){
            e.printStackTrace();
        }
    }
}

/*-----
 * Metodo que modificar datos de un registros en el fichero
 * el campo clave determina la posición del registro dentro del
fichero
 * hay que posicionarse para leer el registro y hay que volver a
posicionarse
 * antes de escribir
 */
public static void modificarRegistro(){
    int clave=0;
    String nombre="";
    String respuesta = "";
    int edad=0;
    RandomAccessFile puntero = null;
    try {
        // abriendo archivo, capturando datos
        puntero = new RandomAccessFile (f,"rw");

        clave =
Integer.parseInt(IntroducirDatos.introducirDatos("Introducir la clave
"
        + "a modificar. < 0 para Finalizar>: "));
        while(clave != 0){
            //colocamos el puntero según la clave para leer
            puntero.seek((clave-1) * tamanoRegistro);

            //leemos los campos del registro
            clave=puntero.readInt();
            nombre = puntero.readUTF();
            edad=puntero.readInt();

            //visualizamos los datos
            System.out.println(clave+" "+nombre+" "+edad);

            respuesta =
IntroducirDatos.introducirDatos("Desea modificar el registro. S/N");
            if (respuesta.compareToIgnoreCase("s") == 0){
                //teclea los nuevos valores de los campos

```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        nombre =
IntroducirDatos.introducirDatos("Introduce el nombre: ");
        if (nombre.length() < 20) {
            for(int i=nombre.length(); i <20; i++)
                nombre=nombre+" ";
        }
        else {
            nombre=nombre.substring(0,20);
        }
        edad =
Integer.parseInt(IntroducirDatos.introducirDatos("Introduce la edad:
"));

        //recolocamos el puntero según la clave para
escribir

        puntero.seek((clave-1) * tamanhoRegistro);
        puntero.writeInt(clave);
        puntero.writeUTF(nombre);
        puntero.writeInt(edad);
    }
    clave =
Integer.parseInt(IntroducirDatos.introducirDatos("Introducir la clave.
< 0 para Finalizar>: "));
    } // fin while
} catch (NumberFormatException nfe){
    System.out.println("Error al introducir los datos");
} catch (IOException ioe) {
    System.out.println("Error de posicionamiento o
lectura");
    System.out.println(ioe.getMessage());
} finally{
    try{
        puntero.close();
    } catch (IOException e){
        e.printStackTrace();
    }
}
} // fin metodo modificarEdad()
}
```

Segundo ejemplo en el que la clave es autoincrementable y la posición del registro en el fichero.

```
package ejemplos10FicherosAleatorios;

import java.io.*;
import aaIntroducirDatos.IntroducirDatos;

/*
```



## Unidad didáctica 1 MANEJO DE FICHEROS.

---

\* En el siguiente ejemplo crearemos una aplicación que nos permita guardar

\* datos de los alumnos de un curso en un fichero de texto con RandomAccessFile.

\* Crearemos un menú para poder seleccionar entre las opciones: Introducir nota,

\* Listado completo, Ver notas de un alumno, Modificar nota y Salir.

\*/

```
public class Ej02ClaveAutoIncrementable {

    //atributos
    private static final int TAMANHOREGISTRO=94; // tamaño del
registro 60 + 30 +4
    private static final int NOMBRELONGMAX=30; // longitud máxima del
nombre
    private static final int MODULOLONGMAX=15; // // longitud máxima
del modulo

    static File f = new File("Notas01.Dat");

    public static void main(final String[] args) {

        int opcion;
        do{
            System.out.println("1.- Introducir Alumnos");
            System.out.println("2.- Listado Alumnos");
            System.out.println("3.- Buscar un Alumno");
            System.out.println("4.- Modificar notas un Alumno");
            System.out.println("5.- Salir");

            opcion =
Integer.parseInt(IntroducirDatos.introducirDatos("Introduce una
opcion: "));
            try{
                switch (opcion){
                    case 1:
                        insertarNotaEnFichero();
                        break;
                    case 2:
                        ListadoCompleto();
                        break;
                    case 3:
                        notasDe1Alumno();
                        break;
                    case 4:
                        modificarNota();
                        break;
                    case 5:
                        System.exit(0);
                }
            }
        }
    }
}
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        default:
            System.out.println("Opción errónea");
        }
    }catch (NumberFormatException e){
        System.out.println("La opción tiene que ser un
numero");
    }
}while(opcion != 5);
} // fin del main

/*-----
 * Método que devuelve el número de registros del fichero
 */
private static int numRegistros(File f){
    RandomAccessFile raf;
    try {
        raf = new RandomAccessFile(f, "r");
        return (int)Math.ceil ((double)raf.length()/
(double)TAMANHOREGISTRO);
    } catch (FileNotFoundException e1) {
        System.out.println("Error el fichero no existe");
        return 0;
    }catch (IOException e){
        System.out.println("ERROR GRAVE DE ENTRADA/SALIDA");
        return 0;
    }
}

/*-----
 * Escribe los datos en el fichero
 */
public static void insertarNotaEnFichero(){
    RandomAccessFile raf;
    try{
        String aux="";
        raf = new RandomAccessFile(f, "rw");
        raf.seek( numRegistros(f)* TAMANHOREGISTRO); // se
coloca al final del fichero

        // si el nombre es demasiado largo vuelve a
solicitarlo, en caso contrario lo escribe en el fichero
        do{
            aux=IntroducirDatos.introducirDatos("Introduce
nombre: ");

            if (aux.length()>=NOMBRELONGMAX){
                System.out.println("\nNombre demasiado
largo\n");

                System.out.println("-----
-----");
            }
        }
    }
}
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        }while(aux.length()>=NOMBRELONGMAX);
        raf.writeUTF(aux);

        // si el nombre del modulo es demasiado largo vuelve a
solicitarlo, en caso contrario
        // lo escribe en el fichero
        do{
            aux=IntroducirDatos.introducirDatos("Introduce
modulo: ");
            if (aux.length()>=MODULOLONGMAX){
                System.out.println("\nNombre de modulo
demasiado largo");
                System.out.println("-----
-----");
            }
        }while(aux.length()>=MODULOLONGMAX);
        raf.writeUTF(aux);
        System.out.println("-----
-----");

        boolean SW = false;
        do{
            SW = false;
            try{

                aux=IntroducirDatos.introducirDatos("Introduce nota: ");
                raf.writeInt(Integer.parseInt(aux));
            }catch (NumberFormatException e){
                System.out.println("Debes introducir un
numero");

                SW = true;
            }
        }while(SW);
    }catch(IOException ioe){
        System.out.print("\nError "+ioe.toString());
    }
}
// fin del metodo meterNotaEnFichero()

/*-----
* Listado de todos los datos del fichero
*/
public static void listadoCompleto(){
    RandomAccessFile raf;

    if (numRegistros(f)!=0){
        System.out.println("\n\t LISTADO COMPLETO\n-----
-----");
        System.out.println("Nombre\tModulo\tNota");
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        System.out.println("\n-----");
    -----");
        try{
            raf = new RandomAccessFile(f, "r");
            for (int i=0; i<numRegistros(f);i++){
                raf.seek(i* TAMANHOREGISTRO);
                System.out.println(raf.readUTF() +"\t"+
raf.readUTF() +"\t"+ raf.readInt());
            }
            System.out.println("\n-----");
    -----");
        }catch (IOException ioe){
            System.out.println("\nError "+ioe.toString());
        }
    }else
        System.out.println("\nNo hay registros introducidos");
    }// fin método listadoCompleto()

//-----
public static void notasDe1Alumno(){
    RandomAccessFile raf;
    String aux = "";
    boolean esta = false;
    if (numRegistros(f)>0) {
        aux = IntroducirDatos.introducirDatos("Introduce el
nombre de un alumno: ");
        try{
            raf = new RandomAccessFile(f, "r");
            for (int i=0; i<numRegistros(f);i++){
                raf.seek(i* TAMANHOREGISTRO);
                if (aux.compareToIgnoreCase(raf.readUTF())
== 0){
                    raf.seek(i* TAMANHOREGISTRO);
                    System.out.println("\n-----");
    -----");
                    System.out.println("Nombre: "+
raf.readUTF());
                    System.out.println("Modulo: "+
raf.readUTF());
                    System.out.println("Nota: "+
raf.readInt());
                    esta = true;
                }
            }
            if (esta==false)
                System.out.println("\nAlumno inexistente");
        }catch (IOException ioe){
            System.out.println("\nError "+ioe.toString());
        }
    }
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        }else
            System.out.println("No hay datos en el fichero");
    }// fin metodo notasDe1Alumno()

    /*-----
    * Metodo que modifica la nota de un determinado alumno
    */
    public static void modificarNota(){
        String aux="";
        String aux2="";
        boolean esta = false;
        RandomAccessFile raf = null;

        if (numRegistros(f)>0) {
            aux = IntroducirDatos.introducirDatos("Introduce el
nombre de un alumno: ");
            aux2 = IntroducirDatos.introducirDatos("Introduce el
modulo: ");

            int nota;
            try{
                raf = new RandomAccessFile(f, "rw");
                for (int i = 0; i < numRegistros(f); i++){
                    raf.seek(i* TAMANHOREGISTRO);
                    if (aux.compareToIgnoreCase(raf.readUTF())
== 0){
                        if
(aux2.compareToIgnoreCase(raf.readUTF())==0){
                            raf.seek(i* TAMANHOREGISTRO);
                            System.out.println("Nombre: "+
raf.readUTF());
                            System.out.println("Modulo: "+
raf.readUTF());
                            nota =
Integer.parseInt(IntroducirDatos.introducirDatos("Introduce la nueva
nota: "));
                            raf.writeInt(nota);
                            esta = true;
                        }
                    }
                }
            }// fin for
            if (esta == false)
                System.out.println("Alumno inexistente");

        }catch (IOException ioe){
            System.out.println("\nError "+ioe.toString());
            ioe.printStackTrace();
        }
    }else
        System.out.println("No hay datos en el fichero");
```

## Unidad didáctica 1 MANEJO DE FICHEROS.

---

```
        try { raf.close();  
        } catch (IOException e) {  
            e.printStackTrace();  
        }  
    }  
} // fin metodo modificarNota
```