

Asignatura

Acceso a datos



UNIVERSAE
Instituto Superior de FP

Asignatura

Acceso a datos

UNIDAD 2

Ficheros



UNIVERSAE
Instituto Superior de FP

Tipos de ficheros y codificación



Ficheros de texto

- Contienen caracteres
- Es legible
- Es posible modificarlo directamente con otros programas

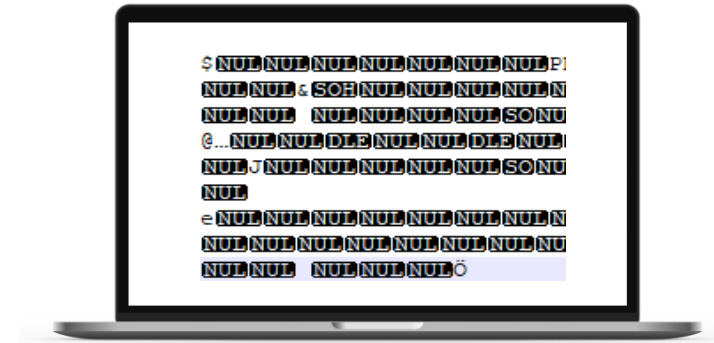
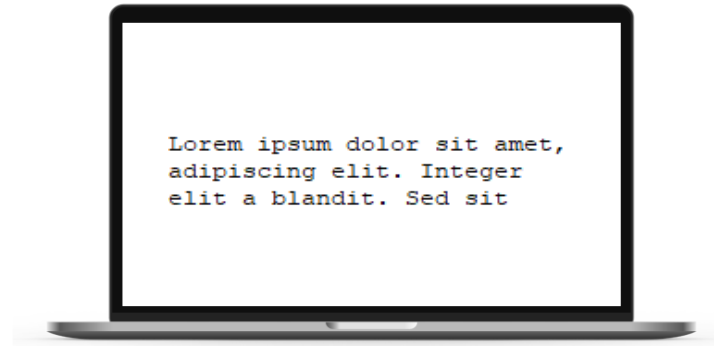


Ficheros binarios

- Contienen bytes
- No es legible
- Hace falta programas específicos para modificarlo
- Ejemplo, .exe

Codificación

- Forma de agrupar la información
- Bit, byte, hexadecimal, octal
- ASCII (7 bits) , UTF-8 (1-4 bytes)



	Binario	Decimal	Hexadecimal
A	01000001	65	41
B	01000010	66	42
C	01000011	67	43

Java. Las clase File

- Permite navegar por la estructura de ficheros y directorios
- Funciona por sistema de URL
- Opera con el sistema operativo
- Puede conocer permisos sobre ficheros o directorios.

Operación	Descripción
<code>boolean exists()</code>	Comprueba si la ruta existe.
<code>boolean isFile()</code>	Comprueba si es un fichero.
<code>boolean isDirectory()</code>	Comprueba si es un directorio.
<code>File[] listFiles()</code>	Obtiene un listado de tipo File de los ficheros que hay en la ruta.
<code>String getName()</code>	Obtiene el nombre del fichero o directorio.
<code>String getParent()</code>	Obtiene el directorio padre.
<code>String getPath()</code>	Obtiene la ruta.
<code>Boolean canRead()</code>	Comprueba si se puede leer.
<code>Boolean canWrite()</code>	Comprueba si se puede escribir.
<code>Boolean canExecute()</code>	Comprueba si se puede ejecutar.



Formas de acceso

Secuencial

- De principio a fin
- Escribe siempre al final

Clases en java

- Tipo texto
 - FileReader / FileWriter
 - BufferedReader / BufferedWriter
- Tipo binario
 - FileInputStream / FileOutputStream
 - BufferedInputStream / BufferedOutputStream

Aleatorio

- Cualquier parte del fichero
- Necesidad de puntero
- Solo disponible para trabajar en binario

Clases en java

- Tipo binario
 - RandomAccessFile

Proceso de operar con ficheros



Apertura

- Abrir o crear el fichero
- Inicializar buffer, si lo usa.



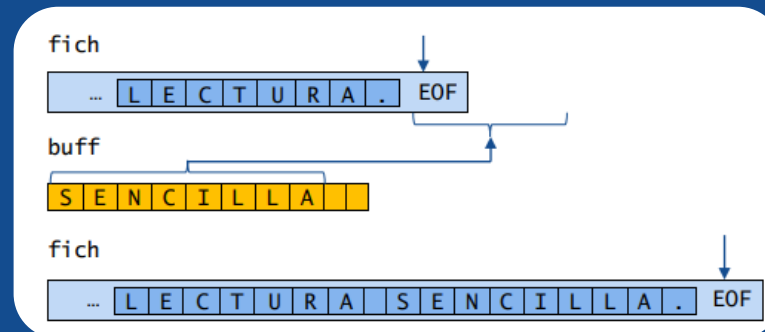
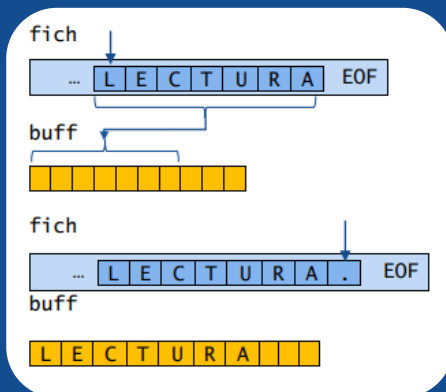
Lectura o escritura

- Métodos de read() o write()



Cierre

- Cerrar buffer y fichero
- Método close()



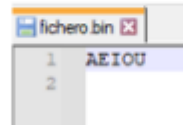
Ejemplos



FileOutputStream y FileInputStream

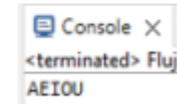
Escritura

```
char[] vocales = { 'A', 'E', 'I', 'O', 'U' };  
  
String rutaFichero = "C:\\documentos\\fichero.bin";  
  
FileOutputStream fos = new FileOutputStream(rutaFichero);  
for (int i = 0; i < vocales.length; i++) {  
    fos.write((byte) vocales[i]);  
}  
  
fos.close();
```



Lectura

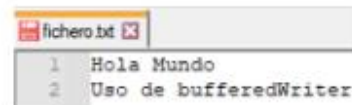
```
// Tiene que existir previamente el fichero, si no, lanzará una excepción  
String rutaFichero = "C:\\documentos\\fichero.bin";  
  
FileInputStream fis = new FileInputStream(rutaFichero);  
int i;  
while ((i = fis.read()) != -1) {  
    System.out.print((char) i);  
}  
  
fis.close();
```



BufferedWriter y BufferedReader

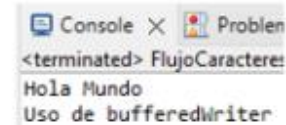
Escritura

```
String rutaFichero = "C:\\documentos\\fichero.txt";  
  
FileWriter fw = new FileWriter(rutaFichero);  
BufferedWriter bw = new BufferedWriter(fw);  
  
bw.write("Hola Mundo");  
bw.newLine();  
bw.write("Uso de bufferedWriter");  
bw.close();  
fw.close();
```



Lectura

```
// Tiene que existir previamente el fichero, si no, lanzará una excepción  
String rutaFichero = "C:\\documentos\\fichero.txt";  
  
FileReader fr = new FileReader(rutaFichero);  
BufferedReader br = new BufferedReader(fr);  
  
int i;  
while ((i = br.read()) != -1) {  
    System.out.print((char) i);  
}  
  
br.close();  
fr.close();
```





Resumen

1. Tipos de ficheros y codificación
2. Java. La clase File
3. Formas de acceso a los ficheros
4. Proceso para operar con ficheros
5. Ejemplos

The background is a solid blue color. Overlaid on this are several faint, light-blue geometric patterns. These include a grid of small squares that form larger, irregular shapes, and numerous small, light-blue arrows pointing in various directions. The overall effect is a sense of movement and digital connectivity.

UNIVERSAE

— CHANGE YOUR WAY —