

## ACCESO A DATOS

### UD01: PRÁCTICA 2 - FICHEROS DE TEXTO EN JAVA

Realiza un proyecto Java denominado **adt1\_practica2** que contenga todos los ejercicios que se piden a continuación. Cada ejercicio es independiente del anterior y tendrá su fichero AppX.java donde estará su método *main()*.

Es decir, para cada ejercicio realizaremos un AppX.java distinto dentro del mismo proyecto, para no tener que crear un nuevo proyecto cada vez.

Tener en cuenta que vamos a crear una carpeta denominada **resources** donde se alojan los ficheros binarios o de texto necesarios para cada uno de los ejercicios.

#### App01 - Fichero de texto

Realiza un programa que contenga un menú que aparezca en pantalla y muestre las siguientes opciones:

- 1.- Leer desde fichero
- 2.- Escribir en el fichero
- 3.- Fin

1. Nos pedirá la ruta del fichero de texto a leer y lo imprimimos por pantalla.
2. Nos pedirá la ruta del fichero a escribir y nos pedirá que introduzcamos un texto por pantalla, que luego escribiremos en el fichero. Si el fichero existe, añadiremos el texto al final del fichero.
3. Finalizará el programa.

Nota: Mientras no finalicemos seguirá mostrando el menú para volver a realizar tareas.

#### App02 - Fichero de texto

Crear un programa que convierta el fichero de texto a mayúsculas. Es decir, dado un archivo de texto, el resultado será el mismo fichero pero con todo el contenido en mayúsculas.

#### App03 - Fichero de texto

Crea un programa que cuente el número de vocales y el número de palabras total.

#### **App04 - Fichero de texto**

Crear un método llamado *compareFileContent()* que podrá comparar si dos ficheros son idénticos o no. El método tiene la siguiente sintaxis:

```
boolean compareFileContent(File file1, File file2)
```

Devuelve cierto en caso de que los dos ficheros coincidan y falso en caso contrario.

#### **App05 - Fichero de texto**

Construye un programa que permita buscar palabras en un fichero de texto. Por lo tanto, se pedirá al usuario la palabra a buscar y para cada línea que contenga la palabra buscada, se debe mostrar el núm. de línea y su contenido.

NOTA: Utilizar método *contains()* de la clase *String*.