



FICHEROS BINARIOS

EJERCICIO 7

Disponemos de un fichero de texto llamado **alumnos.dat** que contiene información de personas y su edad con el siguiente formato:

```
Juan Perez Lima|32  
Jose antonio Callejon Antunez|23  
Felipe Rejon Callado|18
```

Partiendo del proyecto `ejercicios78` se debe hacer un programa que permita crear un fichero de binario a partir de fichero de texto, tomando como base la clase **Persona** vista en los apuntes, usando **serialización**. Para ello completar el método **crearBinario** que creará el fichero **binario**.

EJERCICIO 8

Crear un fichero de texto con el mismo formato del ejercicio anterior a partir del fichero binario que se crea también en el ejercicio anterior. Para ello completar el método **crearTexto** que recibe el nombre del fichero a crear.

VARIACION: Realizar una versión alternativa del **ejercicio8**, donde se copien los datos del fichero original en un fichero XML con el siguiente formato:

```
<colegio>  
  <alumno>  
    <nombre>Juan Perez Lima</nombre>  
    <edad>32</edad>  
  </alumno>  
  ...  
</colegio>
```

EJERCICIO 9

Tenemos información de países del mundo. De cada país se guarda su nombre actual, número de habitantes (en millones), capital y año de creación. Inicialmente no hay datos.

- 1) Completar el método **cargarPaíses** que coge los datos guardados en fichero binario y se cargan en un **HashMap** de objetos del tipo **Pais** (cuya clase ya está implementada)
- 2) Completar el método **mostrarTodo** que devuelve un **String** con toda la información de todos los países. Para ello usar el método auxiliar **crearTabla** que ya está implementado y que recibe como parámetro una colección de países y devuelve un **ArrayList de String[]**.



- 3) Completar el método **añadirPais** que añade un país al **HashMap** recibiendo la información necesaria, controlando que no existen países repetidos con excepciones.
- 4) Completar el método **backupTexto** que hace una copia de seguridad en un fichero de texto cuyo nombre se le pasa como parámetro. El fichero de texto tendrá el siguiente formato en cada línea:

```
Nombre,año,habitantes,capital
```

- 5) Realizar otra versión del ejercicio anterior donde se guarde usando **XML** y usando **JSON**.
- 6) Implementar el método **guardarPaises** que guarda los objetos en un fichero binario con el nombre elegido.

EJERCICIO 10

Tenemos información de las matriculas de un centro de formación. De cada matricula se guarda el nombre del alumno, precio de la matrícula, año del curso académico y si está pagada o no. Inicialmente no hay datos.

- 1) Los datos se guardan en fichero binario llamado **matricula.dat** y al iniciar el programa se cargan en un **HashMap** de objetos del tipo **Matricula** cuya clase ya está implementada, para ello hay que completar el método **cargarMatriculas**.
- 2) Completar el método **visualizarMatriculas** que devuelve un **ArrayList de String[]** con la información de todas las matrículas. Para ello usar el método auxiliar **crearTabla** que ya está implementado y que recibe como parámetro una colección de matrículas y devuelve un **ArrayList de String[]**.
- 3) Completar el método **sumaPagadas** que devolverá un **double** con la suma de todas la matriculas pagadas.
- 4) Completar el método **nuevaMatricula** que recibe los datos de una matrícula y la añade al **ArrayList**. No puede haber alumnos repetidos (lanzar excepción si existe).
- 5) Completar el método el método **guardarMatriculas** que guarda los objetos en un fichero binario con el nombre elegido.
- 6) Completar el método **matriculasResumen** que hace una copia de seguridad en un fichero de texto de las matriculas no pagadas cuyo nombre se le pasa como parámetro. El fichero de texto tendrá un formato con separadores similar a ejercicios anteriores. Cada uno puede elegir cual es el carácter separador.



- 7) Completar el método **rankingMorosos** que devuelve un **ArrayList de String[]** con las con las matriculas no pagadas de más caras a más baratas. Si todas la matriculas están pagadas lanzar una excepción con el mensaje "No hay morosos😊".