

Crear un programa en Java para ayudar al profesor de programación a calcular las medias de los exámenes de su alumnado. Hay que tener en cuenta que en el momento del curso en el que estamos, los alumnos y alumnas tan solo han realizado 4 pruebas hasta el momento, por lo que tan solo se almacenará la nota de 4 exámenes por alumno/a.

La información que el programa debe tratar de cada alumno/a es la siguiente: nombre del alumno/a, apellidos del alumno/a, nota del primer examen, nota del segundo examen, nota del tercer examen y nota del cuarto examen.

Aclarar que la nota media no debe almacenarse si no que se calcula cuándo hace falta.

El programa debe tener la capacidad de, a través de un menú vía línea de comandos, es decir, a través de una interfaz CLI, dar al usuario las siguientes opciones:

1. Mostrar todos los datos del alumnado, es decir, una lista con el nombre completo de cada persona y las 4 notas de sus exámenes junto con la nota media actual.
  - a. El formato para mostrar la información de cada alumno/a debe ser como el siguiente ejemplo:

López González, Fernando  
Tema 1: 8  
Tema 2: 6  
Tema 3: 7,5  
Tema 4: 6  
Nota media: 6,87  
-----

**ACLARACIÓN:** Se deben redondear todas las notas a un máximo de dos decimales a la hora de mostrarlas por pantalla, para ello debe usarse el método *format* de la clase **String**. Un ejemplo de esta implementación sería algo así:

```
double value = 34.766674;  
String result = String.format("%.2f",value);  
System.out.print(result);
```

2. Añadir un nuevo alumno o alumna y registrar sus notas.
3. Eliminar un alumno/a existente. Debe mostrarse un listado corto con un número para indicar al usuario que persona seleccionar a través de ese número y el nombre completo de la misma.
4. Mostrar la nota media de la clase.
5. Mostrar al alumno/a con la mejor nota media y su nota media para poder darle la enhorabuena. Tan solo debe mostrarse el nombre completo y la nota media, no todas las notas.
6. Salir del programa.

**ACLARACIÓN:** Hay que tener en cuenta que deben controlarse las posibles excepciones a lo largo de todo el programa. Además, es también primordial separar la lógica del programa

en las clases correspondientes a cada uno de sus usos, de forma correcta y ordenada para hacer que el programa sea legible y mantenible.

Una vez terminado y probado el programa para entregarlo debe comprimirse la carpeta del proyecto de IntelliJ a un archivo zip y darle el siguiente nombre:

apellido\_nombre\_examen\_tema4.zip

El archivo debe subirse al aula virtual en la entrega correspondiente antes de que termine la hora límite de entrega.