

## Tema 6 - Práctica

Añade a la entrega los ficheros .java y un fichero de ejemplo de cada uno de los ejercicios a realizar. **Si el fichero entregado no coincide con el output del código, se anulará el ejercicio.**

Documenta el código utilizando el formato de **JavaDoc**.

1. (1,5) Crea una clase llamada **GameLogger** en la que se almacenarán los métodos que permitirán crear y trabajar ficheros para esta práctica. Todos sus métodos deberán ser **de clase**. Escribe un primer método que emita la ficha de un personaje pasado por parámetros como fichero de texto. La ficha deberá tener el siguiente formato:

```
Name:Iván
Class:Warrior
Race:Pandaren
Level:80
HP:46967
ATK:127
ARM:70
SPD:52
RES:18
Type:Player
```

2. (1) Escribe un nuevo método en la clase GameLogger que emita como fichero de texto las fichas de un grupo o *party* de Personajes, obtenidos por parámetros como un array de Personajes. El orden en el que se escribirán las fichas es relevante, ya que los personajes irán ordenados de mayor a menor estadística de velocidad. En caso de empate, se ordenará según el nivel descendente.

*Nota: la ordenación puede realizarse mediante los algoritmos de ordenación vistos en la ampliación del tema 4 (algoritmos **burbuja**, **inserción** y **selección**) o, de forma más óptima y totalmente opcional, implementando la interfaz Comparable en la clase Personaje.*

3. (1,5) Sobrecrea un nuevo constructor para cada Personaje, en el que se reciba por parámetros un objeto de la clase File con el path a un fichero que alberga una ficha de Personaje, e inicialízalo leyendo de dicho fichero. Realiza un correcto tratamiento de excepciones, y verifica que el formato del fichero es el adecuado.

*Nota: es posible optimizar esta implementación utilizando polimorfismo. Para ello, investiga el patrón de diseño **Factory**.*

4. (1) Añade un método en la clase Personaje que verifique que el estado de un Personaje casa con las estadísticas que figuran en su ficha. En caso de que ambos elementos no casen, el Personaje debe actualizarse según los datos de su ficha. Ten en cuenta que debes aplicar este comportamiento solo a Personajes que tengan el mismo nombre que su ficha, ya que el nombre es el identificador único.
5. (1) Escribe un método en GameLogger que evalúe si existe un Personaje concreto en los registros del juego. Para ello, recibirá por parámetros un array de File con una serie de paths a distintos ficheros con fichas de Personajes, y un String con el nombre del Personaje a buscar.
6. (1) Escribe un método similar al anterior en la clase GameLogger que evalúe si existen Personajes cuya clase se repita en los ficheros recibidos.
7. (1,5) Escribe un método en la clase GameLogger que reciba el registro de un Combate entre dos Personajes y almacene dicho registro en un fichero.
8. (1,5) Implementa un método en la clase GameLogger para que, a partir de un fichero con el registro de un Combate y un grupo de Personajes, se suba de nivel a un Personaje si ha ganado dicho Combate.