
Gestión de equipo de fútbol

Crear una clase **Futbolista** con los siguientes atributos y métodos:

- Atributos:
 - **Nombre:** nombre del futbolista
 - **Edad:** edad del futbolista
 - **Posición:** Posición que ocupa en el campo el futbolista. Será un tipo enumerado. Podrá ser cualquiera de las siguientes: Portero, Defensa, Centrocampista, Delantero. Cada posición tendrá asociada un salario máximo y un salario mínimo).
 - Portero, entre 2000 y 3000 euros.
 - Defensa, entre 2500 y 3500 euros.
 - Centrocampista, entre 3500 y 4000 euros.
 - Delantero, entre 4000 y 5000 euros.
 - Métodos:
 - Getters y setters
 - toString(): Imprimirá la información asociada al Futbolista

Además Futbolista implementará la interfaz **Deportista**. Esta interfaz poseerá los siguientes métodos:

- **getAnyosProfesional.** Retorna el número de años que un deportista ha estado compitiendo en nivel profesional
- **getListadoEquipos.** Devuelve un listado con el nombre de todos los equipos en los que ha estado el deportista.
- **getTotalTrofeos.** Obtiene el total de trofeos conseguidos por el deportista

También se debe crear una clase **Equipo** que tenga como atributos:

- **Nombre:** nombre del equipo
- **Listado de Futbolistas.** Listado de futbolistas del equipo
- Y se deberá controlar el número máximo de futbolistas por posición. Que serán los que a continuación se definen:
 - Portero: 2 como máximo
 - Defensa: 5 como máximo
 - Centrocampista: 5 como máximo
 - Delantero: 4 como máximo

Además, constará de los siguientes métodos:

- **agregarFutbolista.** Añadirá un futbolista al equipo siempre que la posición de este no esté completa. En caso contrario lanzará una excepción de tipo **RegistroFutbolistaException.**
- **listarFormaciónDelEquipo.** Listará todos los miembros del equipo ordenados por posición. Se mostrarán en el siguiente orden:
 - Porteros, Defensas, Centrocampistas, Delanteros

La gestión se hará en dos pasos:

- Primero generaremos el listado ordenado, para ello utilizaréis el método sort asociado al ArrayList, y deberéis sobrescribir el método compare.
- Luego recorreremos este con un iterator y mostraremos la información de cada elemento.

Respecto a la clase Aplicación, simplemente crear un equipo y añadir futbolistas de diferente tipo, generar una solución dónde se pruebe todo lo implementado. Es decir, el hecho de superar el número máximo de futbolistas en cada posición en cada equipo y también que se vea el funcionamiento del método que ordena.