

## Práctica Laboratorio – Temporada de Fútbol

Vamos a simular una aplicación para modelar la competición de la LFP. Lógicamente será una aproximación a la realidad, mucho más compleja que la que es posible realizar en una práctica de FP. Antes de comenzar la temporada, la LFP realiza un sorteo para establecer el calendario, que determina el orden de los partidos. En principio la fecha asignada a cada partido es domingo o miércoles, sin especificar ni día (por ejemplo cada fin de semana hay partidos viernes, sábados y domingos) ni hora ni, por supuesto, resultado. A lo largo de la temporada se va fijando con unas semanas de anticipación el calendario con día y hora y al final de cada partido se actualizan los registros con los resultados y otras muchas estadísticas.

Podéis ver algunas de las estadísticas que se guardan de un partido incluyendo apuestas por ejemplo aquí: <https://datahub.io/sports-data/spanish-la-liga>

El significado de las siglas lo podéis ver aquí <http://www.football-data.co.uk/notes.txt>

De esta forma Partido y Calendario serían el tipo básico y el tipo agregado para modelar el sorteo. Para modelar la competición tendríamos PartidoResultado como tipo base y Temporada como tipo agregado. Vamos a limitar el objetivo de esta práctica a partir de un fichero csv con los resultados de los partidos de una temporada:

```
17/08/2018;Betis;Levante;0;3;0;1
17/08/2018;Girona;Valladolid;0;0;0;0
18/08/2018;Barcelona;Alaves;3;0;0;0
18/08/2018;Celta;Espanol;1;1;0;1
18/08/2018;Villarreal;Sociedad;1;2;1;1
...
```

Para procesar esta información vamos a definir los siguientes tipos:

1. **Partido:** dispone de las propiedades Fecha, Equipo Local y Equipo Visitante. Todas las propiedades son consultables y la fecha es además modificable. La fecha debe cumplir que esté entre el 15 de agosto y el 20 de mayo. Dos objetos Partido son iguales si tienen la misma fecha y los mismos equipos local y visitante. El criterio de orden será fecha y a igualdad de fecha orden alfabético de equipo local. Este tipo Partido se agregaría en tipo Calendario que en esta práctica no se va a implementar.
2. **PartidoResultado:** hereda de Partido y dispone de las propiedades para los goles de los equipos al final y al descanso. Por ejemplo en el primer resultado, el partido terminó 0-3 y al descanso 0-1. Como propiedad derivada devuelve un tipo enumerado RESULTADO con los valores: LOCAL, VISITANTE o EMPATE para indicar quién gana el partido. Igualmente para el resultado al descanso. Como restricción compruebe que el número de goles es positivo y menor de 10. El criterio de igualdad y orden se conserva del tipo Partido. *(En la realidad habría que incorporar la hora del partido como propiedad y el orden se redefiniría ya que en caso de igualdad de día se ordenaría por hora)*
3. **Temporada:** Una temporada tendrá una propiedad nombre y un conjunto ordenado de partidos con sus resultados. El nombre se extraerá del fichero con los resultados quitando la extensión csv. Dos temporadas serán iguales si sus nombres coinciden.

Implemente:

1. El tipo Partido
2. El tipo PartidoResultado que hereda de Partido
3. Una Factoría para crear un objeto Temporada a partir de los datos de un fichero como el facilitado.
4. El tipo Temporada tendrá las siguientes funcionalidades:
  - a. Método tal que dado un equipo devuelva el total de los goles marcados
  - b. Método que devuelva el número de victorias visitantes
  - c. Método que devuelva en cuántos partidos ganaba el equipo local al descanso y terminó ganando el equipo visitante.
  - d. Método tal que dado un equipo y una fecha devuelva los puntos conseguidos por ese equipo hasta esa fecha
  - e. Método que devuelva el conjunto de equipos participantes
  - f. Método que devuelva el partido con más goles en el descanso
  - g. Método que devuelva el campeón de la temporada. A igualdad de puntos, el campeón se determinará por la diferencia de goles a favor menos los goles en contra. Si persiste el empate por mayor número de goles a favor. Si finalmente persiste el empate por orden alfabético.