# proyecto-java-RaulLozano03

proyecto-java-RaulLozano03 created by GitHub Classroom

# Proyecto del Segundo Cuatrimestre Fundamentos de Programación (Curso 2022/23)

Autor/a: Raúl Lozano Santervás uvus:raulozsan

## Estructura de las carpetas del proyecto

\* \*\*/src\*\*: Directorio con el código fuente.

\* \*\*PAQUETE FP:\*\*

\* \*\*PAQUETE fp.leagueOfLegends\*\*

\* \*\*fp.leagueOfLegends.Champions\*\*: clase que contiene los tipos del proyecto.

\* \*\*fp.leagueOfLegends.test.ChampionsTest\*\*: clase que contiene las clases de test del proyecto.

\* \*\*PAQUETE fp.common\*\*

\* \*\*fp.common.Ap\*\*: contiene el record que usaré en el proyecto.

\* \*\*fp.common.PositionEnum\*\*: clase que contiene el enumerado del proyecto.

\* \*\*fp.utiles\*\*: Paquete que contiene las clases de utilidad, en este caso solo contiene la clase fp.checkers, para usar dicho metodo.

\* \*\*/data\*\*: Contiene el dataset del proyecto.

\* \*\*champions.csv\*\*: Archivo csv que contiene datos de diferentes campeones del juego League Of legends.

## Estructura del \*dataset\*

El dataset original League of Legends Champion Stats se puede obtener de la URL http://kaggle.com/carralas/league-of-legends-champion-stats-922. He tenido que modificar el dataset para tener los tipos necesarios para realizar esta primera entrega, hasta el punto de conseguir 11 columnas, la mayoría con datos generados de forma aleatoria. A continuación se describen las 11 columnas del dataset:

\* \*\*name\*\*: de tipo String, es el nombre del campeon.

\* \*\*shield\*\*: de tipo Integer, indica el valor del escudo que tiene cada campeon.

\* \*\*healthPoint\*\*: de tipo Float, son los puntos de salud de los personajes.

\* \*\*attackDamage\*\*: de tipo Integer, contiene la estadistica de ataque fisico de cada campeon.

\* \*\*useMana\*\*: de tipo Boolean, especifica el recurso del campeon a la hora de atacar, es decir, si consume mana o no.

\* \*\*releaseDate\*\*: de tipo LocalDate, contiene la fecha de salida de cada campeon.

\* \*\*skins\*\*: de tipo List, tiene una lista de la gama de skins que han sacado de este campeon.

\* \*\*positionEnum\*\*: de tipo Enum, indica por la calle a la que suele ir cada campeon.

\* \*\*speed\*\*: de tipo Float, indica la velocidad del personaje.

\* \*\*apMin y apMax\*\*: de tipo Integer, usados para el record.

## Tipos implementados

Los tipos que se han implementado en el proyecto son los siguientes:

### Tipo Base - Champions

\*\*Propiedades\*\*:

- \_name\_, de tipo \_String\_, consultable y modificable. Indica el nombre del campeon.

- \_positionEnum\_, de tipo \_PositionEnum\_, consultable y modificable. Indica la calle por la que suele ir el campeon. Puede tomar los valores TOP, MID, JNG, BOT.

- \_shield\_, de tipo \_Integer\_, consultable y modificable. Indica el escudo.

- \_healthPoint\_, de tipo \_Float\_, consultable y modificable. Contiene los puntos de salud.

- \_attackDamage\_, de tipo \_Integer\_, consultable y modificable. Contiene el ataque.

- \_useMana\_, de tipo \_Boolean\_, consultable y modificable. Especifica si el campeon consume mana o no.

- \_releaseDate\_, de tipo \_LocalDate\_, consultable y modificable. Contiene la fecha de salida del campeon.

- \_skins\_, de tipo \_List\<String\>\_, consultable y modificable. Lista la gama de skins del campeon .

- \_speed\_, de tipo \_Float\_, consultable y modificable. Velocidad del campeon.

- \_ap\_, de tipo \_Ap\_, consultable y modificable. Es el record, creado a partir de dos Integer.

\*\*Constructores\*\*:

- C1: Crea un objeto de tipo ```Champions``` a partir de los siguientes parámetros: ```String name, Integer shield, Float healthPoint, Integer attackDamage, Boolean usemana,

LocalDate releaseDate, PositionEnum positionenum, Float speed, Ap ap, List<String> skins```.

- C2: Crea un constructor vacio.

\*\*Restricciones\*\*:

- R1: El nombre no puede estar vacio.

- R2: El valor de "healthPoint" no puede ser inferior a 400.

\*\*\*Criterio de igualdad\*\*: Dos campeones son iguales si todas sus propiedades básicas son iguales.

\*\*Criterio de ordenación\*\*: en orden del que se han creado los atributos, excepto ap que va antes que speed.

\*\*Propiedad Derivada\*\*:

- \_Boolean getFast()\_: Devuelve false si la velocidad del campeon es inferior a 50, se usa para saber si un campeon es rapido o no.

#### Tipos auxiliares

- PositionEnum, enumerado. Puede tomar los valores TOP, BOT, JNG, MID.

- Ap, record. Se hace a partir de los Integer apMin y apMax del csv.