



TRABAJO PRÁCTICO N°6

Asignatura: Programación Orientada a Objetos

Importante: recordar que utilizaremos un guión bajo inicial en cada nombre de atributo como convención para indicar que deben tratarse como atributo privado, mientras que todos los métodos serán considerados públicos.

Ejercicio 1

Crear una clase Jugador, que contenga atributos con información de nombre y un valor booleano `_lesionado`, que indique si el jugador está en condiciones de participar en un partido o no (con valor por defecto `False`).

Luego crear la clase EquipoBasquet, que contenga atributos con información del nombre del equipo y una lista de jugadores. En esta clase se debe implementar el método `agregarJugador`, cuya funcionalidad es agregar un Jugador a la lista, indicando también el número de camiseta que utilizará. Ejemplo:

```
jugador1 = Jugador('Juan')
equipo1 = Equipo('TeamTeam')
equipo1.agregarJugador(jugador1, 6) # el jugador1 usará la camiseta 6
```

En este método, se debe hacer uso de excepciones para considerar las siguientes situaciones:

- El primer parámetro debe ser una instancia de la clase Jugador, mientras que el segundo parámetro debe ser un entero (excepción `TypeError`).
- No se puede agregar a un equipo un jugador que esté lesionado (excepción `JugadorLesionadoError`)
- No se puede asignar un número de camiseta que ya esté siendo utilizado por otro jugador (excepción `CamisetaEnUsoError`).

Crear previamente las excepciones personalizadas que se indican y elegir un mensaje apropiado para cada caso. Comprobar el funcionamiento en un script de pruebas, controlando cada una de las excepciones implementadas.

Ejercicio 2

Crear una clase Pantalla, con atributos `_ancho` y `_alto` para sus dimensiones. En esta clase se deben implementar también algunos métodos para dibujar figuras sencillas en la pantalla, en las cuales se deben incorporar excepciones ante usos incorrectos.

Por ejemplo, el método `dibujarPunto(x,y)` debe considerar:

- Los parámetros deben ser enteros.
- `x` no puede ser negativo ni mayor al ancho de la pantalla.
- `y` no puede ser negativo ni menor al alto de la pantalla.

Implementar este método haciendo uso de excepciones `TypeError` y `RangeError`. Definir otros métodos adicionales (`dibujarCuadrado`, `dibujarLinea`, etc) e implementarlos incluyendo las excepciones que se consideren apropiadas.