

# Práctica: Refactoring

## Introducción

En esta práctica se van a **repartir 6 códigos** diferentes entre los **6 equipos** en los que se ha dividido la clase.

Todos los códigos tiene la misma funcionalidad, pero cada uno ha sido implementado de manera diferente. Es decir, todos son códigos **funcionales**, hacen lo que tienen que hacer de forma correcta, pero la manera de plantear cada solución es diferente.

Todos los códigos tienen una serie de **malos olores** (unos distintos a otros), la idea es que cada grupo **analice** su código, **detecte** el mayor número de malos olores que pueda y trate de **corregirlos**.

Los códigos vienen acompañados de sus correspondientes **tests unitarios**. Como se ha comentado en clase, para asegurarnos de que no hemos cometido errores refactorizando, la idea es que el código **pase todos** los tests unitarios tanto **antes** como **después** de la refactorización.

La idea de esta práctica es intentar refactorizar los códigos haciéndolos más legibles y mantenibles. Si puedes relacionar el **mal olor y/o la solución** con los **vistos en clase** genial, pero habrá otros malos olores que no se habrán explicado en clase (al no estar vinculado a un mal olor en concreto, por ejemplo hecha un vistazo a [Split Temporary Variable](#) ) o que serán fruto de aplicar la lógica que ustedes crean más conveniente para refactorizar (por ejemplo el poner buenos nombres a las variables y funciones).

En clase se han visto las técnicas a través de los malos olores, pero puede haber más técnicas no vistas, para ello puedes consultar el siguiente recurso:

<https://refactoring.guru/es/refactoring/techniques>

## Contexto del ejercicio:

Los códigos se tendrán que refactorizar están relacionados con el **tenis y su marcador de puntos**. Es tarea de cada grupo analizar qué hace el código y cómo lo hace exactamente. Aquí se explicará un poco el contexto de los marcadores del tenis ya que tiene un sistema de puntuación bastante peculiar y para los que no lo conozcan puede ser un poco complicado al principio:

Las **puntuaciones en tenis** se resumen a continuación:

- La **puntuación actual** de cada juego se describe de una manera peculiar para el tenis:
  - 0 puntos → Se describe como "Love" ("Nada" en español)
  - 1 punto → "Fifteen" ("Quince" en español)
  - 2 puntos → "Thirty" ("Treinta" en español)
  - 3 puntos → "Forty" ("Cuarenta" en español)
- Un juego se gana por el primer jugador que haya ganado al menos **cuatro puntos** en total y por una **diferencia** de al menos **dos puntos mas** que el oponente.
- Si las **puntuaciones** son **iguales** al menos se han marcado **tres puntos** por cada jugador → la puntuación es "**Deuce**" ("Iguales" en español).
- Si al menos se han marcado tres puntos por cada jugador y un jugador tiene un **punto más que su oponente**, la puntuación del juego es "Advantaje" ("Ventaja") para el jugador que está en la delantera.

Puedes echar un vistazo al siguiente enlace para más información:

<https://www.usta.com/es/home/improve/tips-and-instruction/national/tennis-scoring-rules.html>

## ¿Qué se pide para esta práctica?

Para esta actividad hay que realizar una presentación de unos 15/20 minutos donde se explique todo el proceso que han seguido para realizar la refactorización, explicando también los malos olores encontrados y cómo los han solucionado.

La idea es que todas la clase pueda ver y familiarizarse con un trabajo real de refactorizar código que no es suyo.

A la hora de **presentar**, cada grupo deberá:

- Explicar el funcionamiento del código que le ha tocado (sin refactorizar).
- Identificar los malos olores que han detectado en el código.
- Explicar las soluciones que han dado para los malos olores (aquí es recomendable presentar el código con malos olores y el código refactorizado en pantalla dividida para que se puedan comparar ambos en clase).
- Ejecutar los tests unitarios sobre el código refactorizado.

En cuanto a la **entrega** para esta actividad, cada grupo deberá subir en un fichero **comprimido .ZIP**:

- El archivo con el **código refactorizado**.
- **Memoria** de la práctica en formato **.PDF** que deberá incluir:
  - Nombre de los miembros del equipo.
  - Script que le ha tocado refactorizar.
  - Explicación del código **sin** refactorizar.
  - **Malos olores** detectados.
  - **Solución** a los malos olores detectados.