

1. (2 puntos) Crea una **estructura de clases** que permita almacenar información de una red social (RedSocial). Dentro de una RedSocial tendremos:
 - a. Nombre de la red social
 - b. Canales indexados por nombre de canal. Por cada canal almacenaremos:
 - i. Nombre del canal
 - ii. Autor/a
 - iii. Los vídeos del canal ordenados por fecha. Por cada vídeo:
 1. Título del vídeo
 2. Fecha
 3. Duración en segundos (número entero)
 4. Un conjunto no ordenado de etiquetas asociadas al vídeo. Por ejemplo: “juegos” y “RPG”. Cada etiqueta es un String
 - iv. Listado de seguidores del canal. Por cada seguidor (Seguidor):
 1. Username
 2. Nombre
 3. Apellidos
2. (4 puntos) A partir de la estructura de clases anterior, implementa los siguientes métodos **en la clase RedSocial**:
 - a. Dado un nombre de canal, devuelve los seguidores de dicho canal. Si el canal no existe se debe lanzar una excepción.
 - b. Dado un nombre de canal y un título de vídeo, devuelve el vídeo en el canal. Si el canal no existe se debe lanzar una excepción, si no existe el vídeo se devuelve null.
 - c. Dada una etiqueta (String), devuelve los vídeos (de todos los canales) que tienen esa etiqueta.
 - d. Dado un autor, devuelve un listado con todo los seguidores que tiene dicho autor en sus distintos canales.
3. (2 puntos) Crea los siguientes test unitarios para probar el método del ejercicio 2.a
4. (2 punto - **Para quien tenga SOLO la 3ª evaluación**) Crea un componente RedSocialApp que realice lo siguiente:
 - a. Crea una RedSocial hardcodeada (reutiliza la RedSocial del test)
 - b. Pide al usuario un nombre de canal
 - c. Busca en la red social los seguidores de dicho canal y muéstranos en pantalla
 - d. En caso de que se produzca alguna excepción, captúrala y muestra un mensaje de error al usuario.
 - e. Asegúrate de que ninguna excepción provoca la finalización abrupta del programa.
5. (2 puntos - **Para quien tenga la 2ª evaluación**) Crea una jerarquía de clases que represente los diferentes tipos de ordenadores existentes. Todo ordenador tendrá:
 - a. Una velocidad del procesador (entero)
 - b. Una cantidad de memoria disponible (entero)
 - c. Un modelo de placa base

Existen tres tipos de vehículos posibles:

- Sobremesa: tendrán como campo específico
 - Tipo de caja (String)
- Portátil: los portátiles tendrán como campo específico
 - Modelo de cargador (String)
- Tablet: las tablet tendrán como campo específico
 - Pulgadas (entero)

Todo ordenador debe ser de sobremesa, portátil o tablet.

Se deben implementar los siguientes métodos:

1. Un método que muestre en pantalla la información del ordenador, incluyendo los campos específicos y el tipo de ordenador.
2. Otro método canRun que dada una cantidad de memoria, devuelve un boolean indicando si el ordenador puede ejecutarlo o no (si tiene suficiente memoria).