

# Programación Orientada a Objetos

## Curso 2023/24

### Sesión 9

#### Ejercicio

Queremos desarrollar una aplicación que analice los mensajes publicados en redes sociales. En el sistema tendremos *rastreadores* encargados de registrar los mensajes para que podamos consultar aquellos que son de interés.

Un **mensaje** se caracteriza por el autor que lo ha publicado y por un texto. Ambas propiedades son cadenas de caracteres y no cambian una vez construido el objeto. El texto de los mensajes puede contener *temas*, que son palabras precedidas por #, por ejemplo, el texto de un mensaje podría ser: "Esta temporada el #ValenciaCF no tiene nada que hacer en #LaLiga". Los temas son una propiedad del mensaje y por tanto pueden ser consultados. El mensaje anterior contendría los temas #ValenciaCF y #LaLiga.

---

#### Ayuda:

- Una cadena que contiene palabras separadas por espacios se puede dividir utilizando el método `split` de la clase `String`. El método retorna las palabras en un array. Ejemplo de uso:

```
String cadena = "Esto es una cadena";
String[] palabras = cadena.split("\\s+");
for (String palabra : palabras) {
    System.out.println(palabra);
}
```

- La clase `String` declara el método `startsWith` que permite consultar si una cadena comienza por otra cadena.
- 

Implementa el método `toString` en la clase `Mensaje` para poder visualizar el contenido de los mensajes en el programa de prueba.

Un **rastreador** se encarga de registrar los mensajes para que puedan ser consultados. Las propiedades que lo caracterizan son (todas son consultables):

- Mensajes (**lista**). Esta colección contiene todos los mensajes registrados por el rastreador.
- Autores (**conjunto ordenado**). Representa el conjunto de autores de los mensajes registrados en el rastreador.
- Temas (**conjunto ordenado**). Corresponde con los temas que aparecen en los mensajes.

Como soporte para la implementación de las propiedades y la funcionalidad se incluyen los siguientes **atributos de implementación**:

- Mapa de autores. Es un mapa ordenado que asocia el nombre de los autores con una lista de mensajes que han publicado. El orden del mapa corresponde al orden alfabético de las cadenas de texto.
- Mapa de temas. Es un mapa ordenado que asocia temas con una lista de mensajes en los que aparecen. El orden del mapa corresponde al orden alfabético de las cadenas de texto.

Inicialmente un rastreador recién construido no tiene mensajes.

La funcionalidad de la clase es la siguiente:

- Método para consultar la lista de mensajes publicados por un autor.
- Método para consultar la lista de mensajes publicados de un tema.
- Método para **registrar un mensaje**. La operación se encarga de registrar un mensaje adecuadamente en las colecciones. Ten en cuenta que el mensaje, además de ser insertado en la lista de mensajes, se tendrá que insertar en las listas de los mapas, esto es, en todos los temas que incluya y en la lista de los mensajes del autor.

## Orden de los mensajes

En primer lugar, implementa el **orden natural** (interfaz `Comparable`) en la clase `Mensaje`. El orden de un mensaje se basa, en primer lugar, en el orden alfabético del autor, y en caso de igualdad, en el orden alfabético del texto.

Incluye un par de métodos sobrecargados encargados de retornar los mensajes ordenados:

- El primer método retornará una lista de mensajes ordenados de acuerdo con el orden natural de la clase `Mensaje`.
- El segundo método recibe como parámetro un criterio de comparación (interfaz `Comparator`). Retornará los mensajes ordenados de acuerdo con el criterio establecido.

## Programa

Implementa un programa que incluya la siguiente funcionalidad:

- Construye un rastreador y registra los siguientes mensajes:

```
("Ernesto", "Los rivales también juegan, no sólo el #Barça")
("Manolete", "El #Barça es poco competitivo esta temporada")
("Manolete", "#Barça y #RealMadrid han roto el mercado de fichajes")
```

- Muestra todos los mensajes registrados en orden secuencial.
- Muestra los mensajes que incluyan el tema "#Barça".
- Muestra todos los autores que han publicado mensajes.
- Muestra todos los mensajes del autor "Manolete".
- Muestra todos los mensajes ordenados de acuerdo con el orden natural de la clase `Mensaje`.
- Muestra todos los mensajes ordenados utilizando como criterio de ordenación el opuesto al orden natural de la clase `Mensaje`.
- Muestra todos los mensajes ordenados por tamaño del texto del mensaje, de menor a mayor.