

UT4. TAREA 6: Sistema de almacenamiento de música

- 1- Crea una interfaz llamada "MusicStorage" que tenga los siguientes métodos:
 - a. boolean addMusic(String music)
 - b. boolean deleteMusic(String music)
 - c. String[] getAllMusic()
 - 2- Crea una clase abstracta llamada "MusicStorageSystem" que implemente la interfaz "MusicStorage". Esta clase debe tener un array de música como atributo, un método para inicializar esta lista y un constructor que recibe como parámetro el tamaño de la lista y la inicializa.
 - 3- Crea dos clases que hereden de "MusicStorageSystem", llamadas "MusicStorageSystemA" y "MusicStorageSystemB". El sistema A almacena la música por orden alfabético* mientras que el sistema B no la ordena. En ambos sistemas, se permite que queden elementos vacíos al inicio, medio o final del array. Consideramos que se borra un elemento del array cuando establecemos su valor a null.

** Para ordenar el array, puedes usar el método sort de la clase Arrays. Sin embargo, el compareTo de String no permite la comparación con valores nulos, por lo que deberás pasarle al método sort, un comparador programado por ti, que implemente la interfaz Comparator.*
 - 4- Crea una clase llamada "MusicPlayer" que tenga como atributo una instancia de la interfaz "MusicStorage". Esta clase debe tener métodos para añadir, eliminar y mostrar (sin nulos) la música almacenada en el sistema. Estos métodos son de tipo String y devuelven las siguientes cadenas:
 - a. *añadir*: "No puede añadir más música." o "Añadida correctamente."
 - b. *eliminar*: "No se ha podido borrar." o "Borrada correctamente."
 - c. *mostrar*: "1. Primera canción
2. Segunda canción
3. Tercera canción
"
- Entrega un fichero comprimido con las clases java implementadas.
 - Se proporciona una clase de pruebas *MusicPlayerTest* para validar el funcionamiento correcto del sistema.