

## Descripción de la aplicación

Se desea desarrollar una aplicación para gestionar la base de datos de una empresa de reproducción de música vía *streaming*.

De cada **canción (Song)** se guardará:

- Título
- Año
- Duración
- Intérpretes (una colección con los nombres de sus intérpretes)

De cada **álbum (Album)** se almacenará:

- Título
- Intérpretes (una colección con los nombres de sus intérpretes)
- Año
- Duración
- Número de canciones
- Tipo (álbum o sencillo)
- Canciones (colección con las canciones (objetos canción) del álbum)

De cada intérprete, **artista (Artist)** (puede ser individual o grupo) se guardará:

- Nombre
- Biografía
- Instagram
- Twitter
- Facebook
- Wikipedia
- Álbumes (colección de los títulos de los álbumes del artista)

De cada **playlist (PlayList)** se guardará:

- Nombre
- Canciones (colección con las canciones (objetos canción) de la playlist).

En el paquete del modelo, además de las clases anteriores, se dispondrá de una clase llamada **Musicfy**, que contendrá los siguientes atributos:

- Una colección de **Song**.
- Una colección de **Album**.
- Una colección de **Artist**.
- Una colección de **PlayList**.

## Funcionalidad de la aplicación

### Arranque del programa

Al arrancar la aplicación se comprueba si existe un fichero binario en la carpeta del escritorio “~/Desktop/musicfy/binarios/”, llamado musicfy.bin. Si existe, lo lee y carga todos los datos en memoria. Si no existe, se importan los datos a partir de ficheros de datos que se encuentran en la carpeta del escritorio

“~/Desktop/musicfy/datos/”. Estos ficheros son:

- **albumes.txt** Su contenido se añadirá a la colección de **Album**, pero además se crearán las canciones ya que se dispone de todos sus datos y se añadirán a la colección de **Song**. El formato del fichero es delimitado por tabulador (\t), a su vez los intérpretes y los títulos de las canciones vienen delimitados por (;) Ejemplo de 2 líneas del fichero:

```
Origen      Sweet California      2018   43 min 13      álbum   Loca;Moving
Target;Amarte;Adicto;Telaraña;Todo a la vez;El amor es el amor;Guay;Me siguen unos
cuantos;Levanta poeira;Serás mi talismán;Voy a olvidarte;Hey Hola Hello
```

```
Se te olvidó      Ana Mena;Deorro      2019   2 min 39 seg   1       sencillo
```

Con la primera línea se generará un objeto de la clase **Album**, y 13 objetos de la clase **Song**. Con la segunda línea se generará un objeto de la clase **Album** y un objeto de la clase **Song**. Los objetos de la clase **Song** se almacenarán en el atributo correspondiente del objeto de la clase **Album** que se crea, pero también se añadirán a la colección de **Song** de **Musicfy**.

- **artistas.txt** Su contenido se añadirá a la colección de artistas. El formato del fichero es delimitado por el carácter (#) A su vez los títulos de los álbumes vienen delimitados por el carácter (;). Ejemplo de línea del fichero:

```
Ana Guerra#Siempre exigente, Ana Guerra avanza por el casillero de nuestra música
popular sin prisa, pero con paso firme. Solo tiene 24 años, pero ya ha demostrado -a su
imponente participación en OT 2017 les remitimos- que es una versátil y rotunda, llamada
a darnos muchas
alegrías.##@anaguerra_ot2017#@anaguerra##https://es.wikipedia.org/wiki/Ana_Guerra#
Reflexión;Lo malo;Ni la hora;Bajito;Sayonara;El viajero
```

### Salida de programa

La aplicación guarda con formato binario todos los datos de la aplicación, mediante la escritura de la clase Musicfy en el archivo musicfy.bin, en la carpeta

~/Desktop/musicfy/binarios/. Todas las clases creadas en este proyecto son *Serializable*.

Una vez iniciada la aplicación aparecerá un menú con diferentes opciones.

- Opción **Generación aleatoria**.
  - Inicializa todas las colecciones con valores aleatorios. Se debe intentar que los valores sean lo más aleatorios posibles, por lo que sería recomendable que los valores se cargaran desde ficheros. Los datos no tienen que ser reales ni guardar relación. Se podrían tener colecciones de títulos tanto de canciones como de álbumes, nombres de artistas, generar años y duraciones, etc.
- Opción **Archivos**: el programa permite
  - Exportar artistas a un archivo llamado `artistas.col`, con formato de columnas. El archivo se escribe en la carpeta `~/Desktop/musicfy/salida/`.
  - Exportar álbumes (sin incluir las canciones) a un archivo llamado `albumes.html`, con formato de tabla html. El archivo se escribe en la carpeta `~/Desktop/musicfy/salida/`.
- Opción **Álbum**. El programa permite efectuar las siguientes operaciones con los álbumes:
  - Altas (añadir un álbum a la colección de álbumes. Habrá que añadir las canciones tanto a la colección del álbum, como a la colección general de Song de Musicfy. En caso de que no exista el artista en su colección correspondiente se añadirá, dejando los atributos de los que no se dispone información con sus valores por defecto.
  - Bajas (borrar un álbum de la colección de álbumes, además hay que quitar el nombre del álbum de la colección de títulos del artista y borrar las canciones de la colección general de Song de Musicfy).
  - Modificaciones (cambia cualquiera de los atributos de un álbum, excepto su título, intérpretes y la colección de canciones).
  - Consulta (se pedirá el título de un álbum por teclado y se mostrarán todos sus datos).
- Opción **Artista**.
  - Altas (añadir un artista a la colección de artistas, se añadirán también los títulos de sus álbumes, pero no se comprobará si existen, ni se darán de alta.
  - Bajas (borrar un artista de la colección de artistas, no se permitirá borrar si alguno de sus álbumes está dado de alta en la colección de álbumes). Se indicará mediante un mensaje los álbumes que todavía tiene dados de alta.
  - Modificaciones (cambiar cualquiera de los atributos del artista, excepto su nombre y la colección de álbumes).
  - Álbumes de un artista, cuyo nombre se pedirá por teclado. Sacará un listado por pantalla, en forma de tabla ordenada por año, de los álbumes, con todos sus atributos, pero sin incluir sus canciones.

- Opción **PlayList**.
  - Altas (añadir una playlist a la colección de playlists), se pedirá el título y el número de canciones que va a tener. Se cogerán aleatoriamente de la colección de canciones de Musicfy, el número de canciones indicadas y se añadirán a la playlist.
  - Eliminar canción. Mostrará la lista de canciones de la playlist y pedirá el título de la que se quiera eliminar. Sólo se eliminará de la playlist.
  - Añadir canción. Pedirá el título de una canción, la buscará en la colección de canciones globales y la añadirá a la playlist.
- Opción **Canciones**.
  - Listado general. Saca por pantalla la relación de canciones ordenada por el año y por el título respectivamente.

## Indicaciones Adicionales

1. La aplicación, realizada individualmente, se entregará en forma de un proyecto de NetBeans (NetBeans 11.1 y Java 12 o Java 13) comprimido en formato **ZIP**. Recuérdese que el proyecto debe estar correctamente identificado con el nombre y DNI del autor.
2. El proyecto debe funcionar correctamente en todas las plataformas; no se admitirán proyectos “cableados” en que las rutas de archivos estén generadas para una sola máquina o plataforma en particular.
3. La aplicación debe respetar las normas de correcta nomenclatura en Java empleadas a lo largo del curso. En particular, no se utilizarán signos diacríticos en identificadores y la codificación del proyecto será UTF-8. Todos los archivos de texto hacen uso de la codificación UTF-8.
4. Todos los listados deberán presentarse en forma de columnas, con encabezados y ancho de columnas fijas, para que aparezcan perfectamente alineados.
5. Se valorará la creación de un `makefile` y el correspondiente `jar`.

## Arquitectura de la aplicación

1. La aplicación debe hacer uso del patrón MVC empleado a lo largo del curso.
2. Todos los ficheros que maneja la aplicación estarán situados en una carpeta del escritorio llamada “musicfy”, según se especifica en diversos apartados.
3. Todos los años son de tipo entero.