

ENTREGA 2 – PROYECTO SPOTIFY SQL

1. Introducción

Este proyecto tiene como objetivo gestionar y analizar información sobre canciones, artistas y álbumes dentro de una plataforma de música en streaming, utilizando una base de datos estructurada en **MySQL**. A lo largo de esta entrega, hemos incorporado nuevas funcionalidades como **vistas, funciones, stored procedures y triggers**, además de poblar la base de datos con datos iniciales.

2. Objetivo del Proyecto

El propósito de esta base de datos es permitir el análisis de popularidad, tendencias y características musicales a partir de los datos almacenados. Con las funcionalidades agregadas en esta segunda entrega, se facilita la consulta eficiente de información clave sobre la música disponible en la plataforma.

3. Vistas Implementadas

Para facilitar la consulta de información relevante, se crearon varias **vistas SQL** que permiten acceder de manera más rápida a los datos más consultados:

- **v_canciones_mas_populares_por_artista**: Muestra las canciones más populares por cada artista.
- **v_duracion_promedio_albumes**: Calcula la duración promedio de las canciones en cada álbum.
- **v_artistas_canciones_mas_populares**: Lista los artistas junto con sus canciones más reproducidas.
- **v_canciones_por_genero**: Permite ver cuántas canciones hay por cada género musical.
- **v_albumes_mas_canciones**: Muestra los álbumes con más canciones registradas.

Estas vistas simplifican el análisis y optimizan las consultas sobre el rendimiento de canciones y álbumes en la plataforma.

4. Funciones Implementadas

Se implementaron **funciones SQL** para calcular valores específicos dentro de la base de datos:

- **calcular_duracion_promedio_album(id_album_param)**: Devuelve la duración promedio de las canciones dentro de un álbum determinado.

- **calcular_popularidad_media_artista(id_artista_param):** Calcula la popularidad media de un artista en función de sus canciones más escuchadas.

Estas funciones mejoran la capacidad de análisis y permiten obtener métricas relevantes sin necesidad de cálculos manuales.

5. Stored Procedures

Los **stored procedures** facilitan la automatización de tareas comunes dentro de la base de datos. Se incluyeron:

- **insertar_artista(nombre, genero):** Permite agregar un nuevo artista a la base de datos.
- **insertar_cancion(titulo, id_album, duracion_ms):** Facilita la inserción de nuevas canciones en la base de datos, asegurando su correcta asociación con un álbum.

Estos procedimientos agilizan la gestión de la información y reducen la posibilidad de errores manuales.

6. Triggers

Para garantizar la integridad de los datos y automatizar ciertos procesos, se implementaron **triggers** que ejecutan acciones de manera automática:

- **before_delete_cancion:** Asegura que al eliminar una canción, también se eliminan sus registros en las tablas de Popularidad y Características.
- **after_insert_cancion:** Cuando se inserta una nueva canción, automáticamente se le asigna una popularidad inicial de **0** y se agregan sus reproducciones como **0**.

Esto permite mantener los datos sincronizados sin necesidad de intervención manual.

7. Inserción de Datos

Para poblar la base de datos y probar las funcionalidades implementadas, se insertaron datos iniciales de prueba, incluyendo:

- **5 artistas populares.**
- **5 álbumes icónicos.**
- **5 canciones reconocidas** con sus respectivas características y datos de popularidad.

8. Conclusión

Con esta segunda entrega, la base de datos ha evolucionado de un simple almacenamiento de información a una herramienta funcional y optimizada para el análisis de datos musicales. Las vistas, funciones y procedimientos agregados permiten una **gestión más eficiente y automatizada** de la información, mejorando el acceso y análisis de los datos.

Este proyecto sigue en desarrollo y puede expandirse con nuevas funcionalidades como recomendaciones basadas en hábitos de escucha, integración con APIs de música o implementación de análisis más avanzados utilizando inteligencia artificial.

✦ **Adjunto:** Script SQL con todas las funciones, procedimientos y datos insertados.