
Hoja04_BDRelacionales_04

PostgreSQL

En los ejercicios de esta actividad tienes que usar la base de datos PostgreSQL **concursumusicacompleto**.

Descarga el script SQL e importálo en el contenedor de Docker donde tienes la base de datos. Para ello usa el comando: **cat concursumusicacompleto.sql | docker exec -i postgresql psql -U postgres**

En esta base de datos se gestiona un concurso de música que se realiza en una página web. En esa página web los usuarios registrados pueden votar canciones de grupos de música (las canciones registradas en la base de datos). De cada grupo puede haber varias canciones.

La tabla **grupos** contiene información sobre todos los grupos y artistas en solitario que participan con alguna canción en el concurso (pueden participar con varias). Cuando se trata de un artista en solitario, indica con false o 0 en la columna **esgrupo**. Por defecto en esta columna se carga true o 1. En **fechaestreno** se registra la fecha de la primera actuación en directo del grupo y en **año grabacion** el año en que grabaron el primer disco. El identificador del grupo **id** es una columna numérica y autoincrementada.

De cada grupo, se tiene en la tabla **componentes** información sobre los componentes del grupo. En la columna **función** se tienen valores como **batería, vocalista, voz y guitarra, etc.**

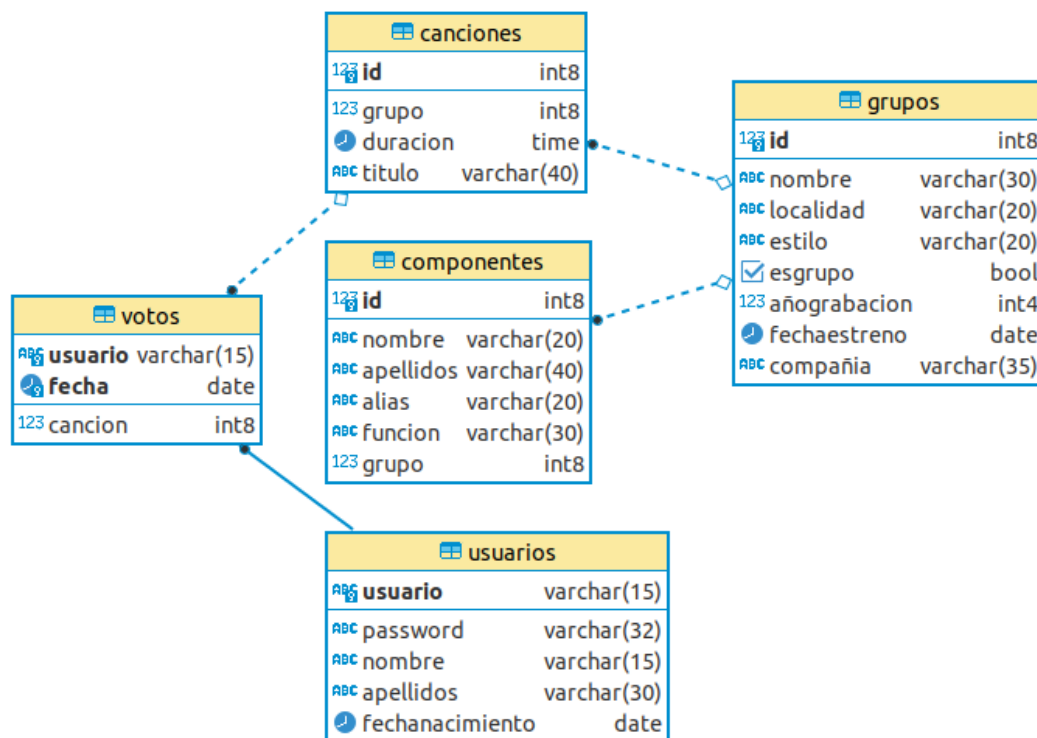
En la tabla **canciones** se tienen todas las canciones que se pueden votar. El identificador de cada canción es el número de canción (**id que es autoincrementada**). La duración es una columna de tipo **TIME**, por tanto, se representan sus datos en formato **HH:MM:SS**. La columna grupo es clave ajena y contiene el identificador del grupo al que pertenece la canción.

Los usuarios de la web se almacenan en la tabla **usuarios**. Cada usuario elige su identificador (**usuario**) y su contraseña. La contraseña debe almacenarse con encriptación **MD5**.

En la tabla **votos** se inserta una fila cada vez que un usuario vota una canción. En esta tabla es **PK fecha+usuario**, por lo que un usuario no pueda dar más de un voto en un mismo día. En la columna **cancion** se registra el número o identificador de canción votada en cada voto.

El esquema relacional correspondiente a esta base de datos es:

Hoja04_BDRelacionales_04



EJERCICIO 1

Realiza un programa que permite añadir un usuario a la tabla usuarios con los datos recogidos por teclado.

No es necesario validar los datos introducidos.

Si que hay que comprobar si al hacer la inserción se produce excepción por clave duplicada debido a que el usuario ya exista.

A partir de ahora en nuestras aplicaciones con bases de datos tendremos una clase Conexion que seguirá el patrón de diseño **Singleton**. Busca información acerca de él y pregunta al profesor en caso de dudas.

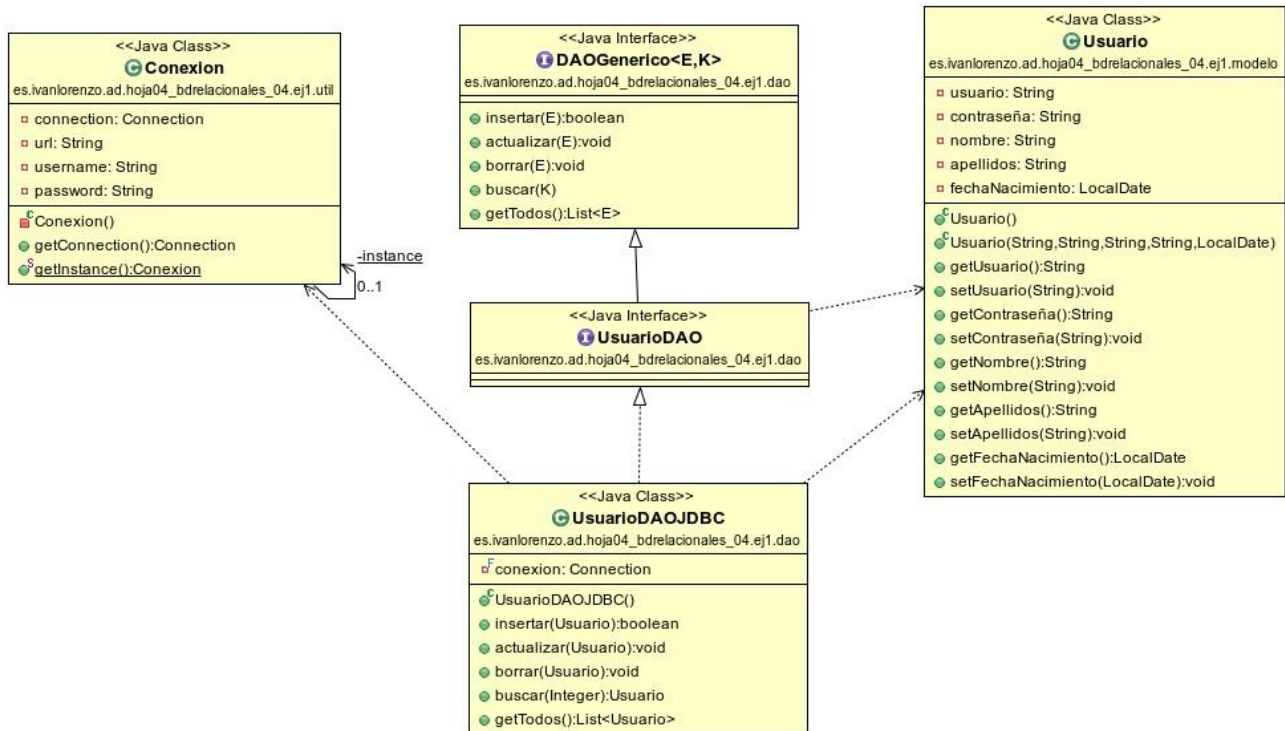
Además, utilizaremos el patrón DAO para los accesos a bases de datos. Para ello crearemos una interface genérica DAOGenerico<E,K> con las operaciones básicas.

UsuarioDAO será una interface que herede de DAOGenerico cuya entidad (E) sea Usuario y cuya clave (K) sea un String (su clave primaria).

Por último, realizaremos la clase UsuarioDAOImpl que implemente UsuarioDAO y en la que desarrollaremos los métodos que necesitemos (en este primer ejercicio únicamente el método insertar).

El diagrama de clases sería similar al siguiente:

Hoja04_BDRelacionales_04



EJERCICIO 2

Realiza un que presenta el siguiente menú (en este caso no es obligatorio utilizar el patrón DAO ya que incrementaría la complejidad del ejercicio):

- 1.- Listado de grupos
- 2.- Listado de canciones
- 3.- Número de canciones por grupo
- 4.- Canciones de un grupo
- 5.- Las 5 canciones más votadas
- 6.- Grupos sin canciones
- 7.- Los últimos 5 votos
- 8.- Eliminar canciones de un grupo
- 9.- Modificar datos de grupo

Y escribe en pantalla los resultados de la opción seleccionada.

Opción 1: El listado de grupos mostrará el código del grupo, el nombre, la localidad y el estilo y debe estar ordenado por el id.

Opción 2: El listado de canciones debe estar ordenado por nombre de grupo y debe tener este formato:

Astrud

Bailando

Hoja04_BDRelacionales_04

Todo nos parece una mierda
Carlos Sadness
Qué electricidad
Perseide
.....

Opción 3: Deben mostrarse todos los grupos y, para cada grupo, el número de canciones que tiene.

Opción 4: Se pide un nombre de grupo y muestran los datos de las canciones del grupo (id, título y duración).

Opción 5: Se deben mostrar, para las canciones con más votos, títulos de canciones y nombres de los grupos.

Opción 6: Se muestra un listado de nombres de grupos sin canciones en la tabla canciones

Opción 7: Se muestran datos de los 5 votos más recientes. Hay que mostrar el título de la canción votada, el grupo al que pertenece y la fecha del voto.

Opción 8: Se pide por teclado el nombre de un grupo y, si existe, se eliminan todas las canciones del grupo. Como está restringido el borrado de canciones si tienen votos, se han de eliminar primero los votos de las canciones del grupo. *NOTA: Buscad primero cómo realizar un DELETE con Postgresql.*

Opción 9: Se pide por teclado el nombre de un grupo y, si existe, se escribe en pantalla, el estilo, el año de primera grabación de un disco, la fecha de la primera actuación, la localidad y la compañía. A continuación, se escribe un menú preguntando cual de los anteriores datos se quiere modificar, incluido el nombre del grupo, y se modifica ese dato con el nuevo valor que se introduzca por teclado.