

Bases de Datos 2022

Parcial I: SQL

Sergio Canchi, Juan Cabral,
Cristian Cardellino, Ramiro Demasi

Contexto

Se dispone de la base de datos ``videogames`` que se detalla en el siguiente diagrama:



Preparación

Para cargar localmente los datos, deberán realizar los siguientes pasos. En el caso de la máquina virtual, tener en cuenta que se deberán modificar algunas cosas de ``schema.sql`` para utilizar la base de datos que tienen asignada, que tiene el formato ``mysql_xxxxx_2022``. **NO ELIMINEN LA BD DE LA VM O NO PODRÁN HACER EL PARCIAL. SI LA BD TIENE TABLAS ELIMINEN DICHAS TABLAS:**

```
$ mysql -h <host> -u <user> -p<password> < schema.sql
$ mysql -h <host> -u <user> -p<password> < data.sql
```

Consignas

1. Crear la tabla ``reviews`` que tendrá los reviews hechos por los usuarios de distintos juegos, deberá constar con los siguientes campos:
 - a. ``user``: Usuario que hizo la review. Debe estar asociado a un usuario existente.
 - b. ``game``: Juego al que corresponde la review. Debe estar asociado a un juego existente.
 - c. ``rating``: Rating asignado. Es un valor de [punto fijo](#) asignado al rating de la review. El valor puede estar entre 0 y 5 y puede tener 1 valor decimal. No debe ser nulo.
 - d. ``comment``: Es un texto, que puede ser nulo, de un máximo de 250 caracteres.Un usuario sólo puede hacer 1 review por juego, por lo que deberán asegurar unicidad. Tener en cuenta a la hora de elegir los tipos de datos que sean lo más eficientes posibles. Además, deberán coordinar con los valores que se definen en el archivo ``reviews.sql``, que deberán cargar mediante el siguiente comando:

```
mysql -h <host> -u <user> -p<password> < reviews.sql
```
2. Eliminar de la tabla ``reviews`` todas aquellas filas cuyo campo ``comment`` sea nulo y modificar la tabla ``reviews`` de manera que no acepte valores nulos en el campo ``comment``.
3. Devolver el nombre y el rating promedio del género con mayor rating promedio y del género con menor rating promedio de acuerdo a los ratings de los reviews de juegos de cada género. Deberán realizar una sólo consulta para dicha tarea.
4. Agregar una columna a la tabla ``user`` llamada ``number_of_reviews`` que deberá ser un entero. La columna deberá tener por defecto el valor 0 y no podrá ser nula.
5. Crear un procedimiento ``set_user_number_of_reviews`` que tomará un nombre de usuario y actualizará el valor ``number_of_reviews`` de acuerdo a la cantidad de reviews hechos por dicho usuario.

6. Crear dos triggers:
 - a. Un trigger llamado ``increase_number_of_reviews`` que incrementará en 1 el valor del campo ``number_of_reviews`` de la tabla ``user``.
 - b. Un trigger llamado ``decrease_number_of_reviews`` que decrementará en 1 el valor del campo ``number_of_reviews`` de la tabla ``user``.

El primer trigger se ejecutará luego de un ``INSERT`` en la tabla ``reviews`` y deberá actualizar el valor en la tabla ``user`` de acuerdo al valor introducido (i.e. sólo aumentará en 1 el valor de ``number_of_reviews`` para el usuario que hizo la review). Análogamente, el segundo trigger se ejecutará luego de un ``DELETE`` en la tabla ``reviews`` y sólo actualizará el valor en ``user`` correspondiente.
7. Devolver el nombre y el rating promedio de las 5 compañías desarrolladoras (i.e. pertenecientes a la tabla ``developers``) con mayor rating promedio, entre aquellas compañías que hayan desarrollado un mínimo de 50 juegos.
8. Crear el rol ``moderator`` y asignarle permisos de eliminación sobre la tabla ``reviews`` y permiso de actualización sobre la columna ``comment`` de la tabla ``reviews``.
9. Actualizar la tabla ``user`` de manera que ``user.number_of_reviews`` refleje correctamente la cantidad de reviews hechas por el usuario. **Hint:** Este ejercicio se resuelve haciendo uso de `INSERT INTO ... ON DUPLICATE KEY UPDATE`. **Punto Extra:** Este ejercicio suma hasta 1 punto, pero no resta.

Puntos a tener en cuenta

- Algunos nombres de columnas o tablas pueden ser palabras reservadas de SQL. Para evitar problemas, utilizar ``backticks``:
 - E.g. `SELECT u.username FROM `user` u;`
- Mostrar únicamente los campos pedidos en la consigna y en el orden en el que se los pide (tanto a nivel fila como a nivel columna).
- Buscar hacer la consulta de la forma más sencilla posible.
- Se evaluará el correcto formato de las soluciones:
 - El código entregado debe ser legible.
 - Utilizar indentación de 4 espacios.
 - No escribir toda la consulta en una sola línea. Usen buen criterio para separar partes de la consulta.
 - Utilizar mayúsculas para denotar palabras clave de SQL (e.g. ``SELECT``, ``INSERT``, ``FROM``, etc.).
 - Utilizar CTEs (i.e. ``WITH``).

Entrega

- Se entregará un archivo comprimido ``soluciones.sql.gz`` o ``soluciones.zip`` (con ``soluciones.sql`` adentro) con las soluciones de los 9 ejercicios. Separar las soluciones mediante comentarios de SQL.
- La entrega se hará mediante el [Aula Virtual](#) en el [correspondiente apartado](#).
 - Tendrán hasta las 18:30 para que se considere una entrega completa. La recomendación es empezar a subir el archivo a las 18 para evitar cualquier eventualidad.
 - Si se entrega después de esa hora, el límite serán las 19:00 y se descontará 1 punto por entrega tardía.
 - Después de las 19:00 se cerrará la entrega y el parcial se considerará desaprobado.