

Bases de Datos I

01/10/2020

Apuntes de la clase 01/10

IS-501

José Inestroza

Estudiar en particular: **Libro Principal Capítulo 05 – Modelo relacional** en el cual se deben hacer los ejercicios y preguntas de fin de capítulo.

Manual de referencia SQL → Lectura Capítulo 03, Capítulo 11, Capítulo 13, adicionalmente Capítulo 08.

El propósito de conocer sobre el modelo relacional es diferenciar las herramientas que están hechas para que el profesional pueda diseñar la base de datos VS las herramientas con las que se construyen los sistemas de gestión de bases de datos. El concepto de base de datos relacional es que a partir de las bases de datos de los 80's (SO de IBM o bases de datos DB2) se comenzó a aplicar el modelo relacional para que se pueda hacer que los sistemas se integren con un sistema gestor de bases de datos que es relacional, finalmente los sistemas open source como postgres o MySQL-MariaDB. La representación de un modelo relacional, existe una relación (tabla), donde existen registros (tuplas), donde esos registros tienen información relacionada que eventualmente se representa como una tabla con campos y cada uno de sus campos sus atributos y los valores de esos campos son dominios, los dominios son elementos atómicos de abstracción, ese dominio corresponde a un tipo de dato.

El estado de una relación, si la relación es la definición de la tabla, el estado corresponde con el contenido de esa relación en ese momento dado por tuplas; los atributos que tiene cada tupla son elementos que están relacionados y por lo tanto se le da su componente relacional.

Ejercicio – ej_gameCatalogue.sql:

En base al código realizar:

1. El diagrama entidad relación y el modelo entidad relación.
2. Aplicar restricciones de integridad de Foreign Key basado en el ER en instrucciones SQL, aplicadas tanto en crear las tablas como en alterar las tablas.
3. Aplicar las restricciones de integridad mediante CREATE TABLE y mediante ALTER TABLE.

Diferencia entre ALTER TABLE y CREATE TABLE: Puede haber una base de datos que ya contenga una restricción de estructura donde ya están asociados campos pero los campos no quedaron asociados mediante restricciones de integridad en el SQL por lo tanto se puede agregar un jugador en gamePlayer que no exista en player, el sistema de gestión de bases de datos propicia esa integridad a través de instrucciones SQL, esas instrucciones SQL se agregan en la definición de la tabla, también se pueden agregar restricciones alterando la tabla ya existente.

Para el desarrollo de la base de datos la integridad debe estar separada de la creación de las tablas, de tal manera que cuando se hagan pruebas en el sistema se puedan borrar los elementos de las tablas sin tener las alertas de las restricciones de la integridad.

Las Restricciones de integridad: Tiene que prohibir que se puedan eliminar el primer registro de distintas tablas, es decir llaves foráneas que se utilizan en otras tablas. Si la integridad no existe se producen agujeros de información en la base de datos, por lo tanto se debe tener alguna forma para restringir que información se puede eliminar o modificar

haciendo uso de las restricciones de integridad. La integridad de la información no se debe quebrantar eliminando datos de la base de datos.

Crear base de datos

Comando: **\$ CREATE DATABASE IF NOT EXISTS GameCatalogue;**

Seleccionar la base de datos

Comando: **\$ USE GameCatalogue;**

Crear las tablas de la base

Comando:

```
$ CREATE TABLE IF NOT EXISTS Player(  
    id INT AUTO_INCREMENTAL PRIMARY KEY,  
    tex_name TEXT NOT NULL,  
    tim_creation TIMESTAMP DEFAULT NOW(),  
    cod_state ENUM('active', 'blocked', 'inactive') DEFAULT 'active'  
) CHARACTER SET utf8;
```

```
$ CREATE TABLE IF NOT EXISTS Game(  
    id INT AUTO_INCREMENTAL PRIMARY KEY,  
    Tex_name TEXT NOT NULL  
) CHARACTER SET utf8;
```

Aplicar la restricción de integridad referencial en el campo de id_player y id_game, donde dichas llaves tienen que existir en la tabla correspondiente, dado que no se puede asociar el id de un usuario que no existe.

```
$ CREATE TABLE IF NOT EXISTS GamePlayer(  
    id INT AUTO_INCREMENTAL PRIMARY KEY,  
    id_player INT NOT NULL,  
    id_game INT NOT NULL,  
    tim_lastPlayed TIMESTAMP NULL,  
    cod_state ENUM('on-progress', 'not-played', 'beated') DEFAULT 'not-played'  
) CHARACTER SET utf8;
```

Eliminar tablas

```
DROP TABLE Player;  
DROP TABLE Game;  
DROP TABLE GamePlayer;
```

Eliminar base de datos

DROP DATABASE GameCatalogue;

En Consola

Para iniciar el servicio SQL y cargar el archivo a la base de datos de forma local

Comando: **\$ sudo service mysql start; mysql -u admin -p < ej_gameCatalogue.sql**

Luego iniciar la base de datos

Comando: **\$ mysql -u admin -p;**

