

Fundamentos de Bases de Datos.

Práctica 10.

Profesor: M.I. Gerardo Avilés Rosas

gar@ciencias.unam.mx

Laboratorio: Luis Eduardo Castro Omaña

lalo_castro@ciencias.unam.mx

26 de abril de 2017

Se dan a conocer especificaciones de entrega para la práctica 10.

1. SQL

IBM desarrolló la versión original en su Laboratorio de Investigación de San José (San José Research Center, actualmente Centro de Investigación de Almadén, Almadén Research Center). IBM implementó el lenguaje, originalmente denominado Sequel, como parte del proyecto System R, a principios de 1970. El lenguaje Sequel ha evolucionado desde entonces y su nombre ha pasado a ser SQL (Structured Query Language, Lenguaje estructurado de consultas).

En 1986, ANSI (American National Standards Institute, Instituto Nacional Americano de Normalización) e ISO (International Standards Organization, Organización Internacional de Normalización), publicaron una norma SQL, denominada SQL-86. En 1987, IBM publicó su propia norma de SQL corporativo, Interfaz de bases de datos para arquitecturas de aplicación a sistemas (Systems Application Architecture Database Interface, SAA-SQL). En 1989 se publicó una norma extendida para SQL denominada SQL-89 y actualmente los sistemas de bases de datos son normalmente compatibles al menos con las características de SQL-89. La siguiente versión de la norma fue SQL-92 y la versión más reciente es SQL:1999.

El lenguaje SQL tiene varios componentes:

- Data Definition Language¹ (DDL). El DDL de SQL proporciona órdenes para la definición de esquemas de relación, borrado de relaciones, creación de índices y modificación de esquemas de relación.

¹Su traducción al español: Lenguaje de definición de datos.

- Data Manipulation Language² (DML). El DML de SQL incluye un lenguaje de consultas, basado tanto en el álgebra relacional como en el cálculo relacional de tuplas. Incluye también órdenes para insertar, borrar y modificar tuplas de la base de datos.
- Definición de vistas. El DDL de SQL incluye órdenes para la definición de vistas.
- Control de transacciones. SQL incluye órdenes para la especificación del comienzo y final de transacciones.
- SQL incorporado y SQL dinámico. SQL dinámico e incorporado define cómo se pueden incorporar las instrucciones SQL en lenguajes de programación de propósito general, tales como C, C++, Java, PL/I, Cobol, Pascal y Fortran.
- Integridad. El DDL de SQL incluye órdenes para la especificación de las restricciones de integridad que deben satisfacer los datos almacenados en la base de datos. Las actualizaciones que violen las restricciones de integridad se rechazan.
- Autorización. El DDL de SQL incluye órdenes para especificar derechos de acceso para las relaciones y vistas

2. DML

2.1. Insert

Para insertar datos en una relación, o bien se especifica la tupla que se desea insertar o se formula una consulta cuyo resultado sea el conjunto de tuplas que se desean insertar. Obviamente, los valores de los atributos de la tuplas que se inserten deben pertenecer al dominio de los atributos. De igual modo, las tuplas insertadas deberán ser de la aridad correcta.

La instrucción insert más sencilla corresponde a la de inserción de una tupla.

```
INSERT INTO <TABLA>
VALUES (v1, v2, ... , vn)
```

En este ejemplo los valores se especifican en el mismo orden en que los atributos se listan en el esquema de relación. Para beneficio de los usuarios, que pueden no recordar el orden de los atributos, SQL permite que los atributos se especifiquen en la cláusula insert. Así, el siguiente ejemplo tiene una función idéntica al anterior:

```
INSERT INTO <TABLA> (c1, c2, .. ., cm)
VALUES (v1, v2, ... , vm)
```

²Su traducción al español: Lenguaje interactivo de manipulación de datos

Es posible insertar tuplas resultantes de una consulta:

```
INSERT INTO <TABLA>
SELECT *
FROM <TABLA>
```

2.2. Actualizar

En determinadas situaciones puede ser deseable cambiar un valor dentro de una tupla, sin cambiar todos los valores de la misma. Para este tipo de situaciones se utiliza la instrucción update.

```
UPDATE <TABLE> SET c1 = v1, c2 = v2, ... , cn = vn
```

2.3. Borrar

Se pueden borrar sólo tuplas completas, es decir, no se pueden borrar valores de atributos concretos. Un borrado se expresa en SQL del modo siguiente:

```
DELETE FROM r
WHERE P
```

donde P representa un predicado y r representa una relación. La declaración delete selecciona primero todas las tuplas t en r para las que P (t) es cierto y a continuación las borra de r. La cláusula where se puede omitir, en cuyo caso se borran todas las tuplas de r.

Hay que señalar que una orden delete opera sólo sobre una relación. Si se desea borrar tuplas de varias relaciones, se deberá utilizar una orden delete por cada relación.

3. Actividad

EL objetivo de la práctica es poblar la base de datos que han construido hasta ahora. Para esto, deben utilizar DML para insertar como mínimo 100 registros por tabla. Deben tener en cuenta las relaciones y restricciones definidas en su esquema, por lo que no todas las tablas deben tener únicamente 100 registros.

Pueden utilizar un generador de datos si así lo desean, <http://www.generatedata.com/> es un ejemplo de generador.

Se tomará como puntos extras si utilizan inserciones a partir de tablas ya pobladas.

4. Entregables

Se debe entregar un script llamado DML.sql el cual tendrá las instrucciones correspondientes para poblar su base de datos.

La entrega deberá ser el día martes 2 de Mayo de 2017.