

Universidad del Quindío - Ingeniería de Sistemas - Leonardo Hernández R.
leonardohernandez@telesat.com.co - www.geocities.com/leoher314
Sep. 7/2002, Mar. 3/2003

Tutorial de MySQL

Lección I

1. Introducción

MySQL es un poderoso interpretador de SQL. Según su sitio web, MySQL es el servidor de base de datos de código fuente abierto más popular en el mundo, con más de dos millones de instalaciones. Importantes instituciones como Yahoo! Finance, MP3.com, Motorola, NASA, Silicon Graphics, y Texas Instruments usan el servidor MySQL en aplicaciones de misión crítica.

2. Instalación de MySQL en Windows, activación y uso del servicio.

2.1 Instalación de MySQL

Para instalar MySQL, y de paso el servidor de páginas web Apache y el lenguaje PHP, ejecute el siguiente programa del CD distribuido por el profesor:

```
D:\PHPTriad\SERVER\PHPTRIADSETUP2_11.exe
```

Se supone que D: corresponde a la unidad de CD-ROM.

Este software también se puede descargar de Internet.

2.2 Activación del servicio de Bases de Datos MySQL

La distribución de MySQL incluye la tarea servidor y la tarea cliente, la primera hace que nuestro PC preste el servicio de Bases de Datos. La segunda permite enviar solicitudes a la primera.

Para activar el servicio de MySQL ejecute el programa:

```
C:\apache\mysql\bin\winmysqladmin.exe
```

y responda:

```
User name = root  
Password = root
```

Aparece un semáforo a la derecha de la barra de tareas de Windows. Si el semáforo está en verde todo quedó bien. Si el semáforo está en rojo, Ud. está en graves dificultades: posiblemente le convenga reinstalar Windows.

La próxima vez que encienda el PC, se activará este servicio automáticamente, por lo que no tendrá que realizar este paso otra vez.

2.3 Ejecución de la tarea cliente MySQL

La tarea cliente MySQL permite crear bases de datos y tablas, insertar y modificar datos, realizar consultas y cientos de operaciones más.

Para ejecutar la tarea cliente MySQL, ejecute el programa:

```
C:\apache\mysql\bin\mysql.exe
```

Esto ocasionará que aparezca en pantalla una consola (una ventana de solo texto) con el prompt de MySQL:

```
mysql>_
```

Note que la ventana en modo texto nos indica que la tarea cliente se está ejecutando, mientras que el semáforo en verde nos indica que el servidor de bases de datos esta activo.

Para salir de MySQL teclee:

```
mysql> quit
```

El prompt `mysql>` no debe teclearse porque ya aparece en la pantalla. El símbolo `␣` representa la tecla enter o intro.

3. Ingreso de comandos en MySQL

Con la siguiente instrucción se puede observar la fecha:

```
mysql> select current_date();
```

Se obtiene el siguiente resultado:

```
+-----+
| current_date() |
+-----+
| 2003-03-04     |
+-----+
1 row in set (0.00 sec)
```

Puede usar dos renglones para la instrucción. El prompt `->` indica que MySQL está esperando a que Ud. termine de entrar la instrucción. Ejemplos:

```
mysql> select
-> current_date();
```

o también:

```
mysql> select current_date();
-> ;
```

El prompt `->` aparecerá cuando a Ud. le falte teclear el punto y coma final de una instrucción.

Aparecerá un prompt especial cuando falten las comillas finales de una cadena de caracteres. Ejemplo:

```
mysql> select "ho"
"> la";
```

Cuando ya no se desea ejecutar una instrucción, de la cual se han escrito dos o más renglones, se utiliza `\c enter` (*alt-92 c enter*), por ejemplo:

```
mysql> select
-> current\c
```

4. Trabajo a nivel de bases de datos

Para este apartado usaremos de ejemplo la base de datos de una nómina.

Puede crear la base de datos con:

```
mysql> create database nomina;
```

En Windows crear una base de datos de MySQL es simplemente crear una subcarpeta de la carpeta `C:\apache\mysql\data`. En este caso la subcarpeta `c:\apache\mysql\data\nomina`.

MySQL no diferencia entre mayúsculas y minúsculas en las palabras clave. La siguiente instrucción es equivalente a la anterior:

```
mysql> Create DATABASE nomina;
```

En el sistema operativo Windows tampoco se diferenciará entre mayúsculas y minúsculas en los nombres de tablas y de bases de datos. Al contrario, el sistema operativo Linux sí tendrá en cuenta esta diferencia. En Windows la siguiente instrucción es equivalente a las anteriores, pero no en Linux:

```
mysql> Create database NoMiNa;
```

Puede listar las bases de datos MySQL existentes en el sistema con:

```
mysql> show databases;
```

Es posible borrar una base de datos creada previamente. Por ejemplo:

```
mysql> drop database nomina;
```

En este caso, se remueve la carpeta correspondiente.

5. Tipos de datos en MySQL

Convenciones:

m: máximo número de caracteres que puede ocupar el dato en la pantalla.

d: Número de dígitos que siguen al punto decimal

Tipo	Bytes	Rango	Precisión
tinyint	1	-128 a 127	
tinyint unsigned	1	0 a 255	
smallint	2	-32.768 a 32.767	
smallint unsigned	2	0 a 65.535	
mediumint	3	-8'388.608 a 8'388.607	
mediumint unsigned	3	0 a 16'777.215	
int o integer	4	-2.147'483.648 a 2.147'483.647	
int unsigned o integer unsigned	4	0 a 4.294'967.295	
bigint	8	-9'''223.372''036.854'775.808 a 9'''223.372''036.854'775.807	
bigint unsigned	8	0 a 18'''446.744''073.709'551.615	
float o float(4)	4	-3.4e+38 a -1.1e-38 1.1e-38 a 3.4 e+38	9 cifras
double o float(8) o real o double precision	8	-1.7e+308 a -2.2e-308 2.2e-308 a 1.7e+308	16 cifras
decimal(m,d) o numeric(m,d)	m , d+2 si m<d	determinado por m y d	
date	3	una fecha	
datetime	8	un par fecha-hora	
timestamp	4	un par fecha-hora	
time	3	una hora	
year	1	un año	
char(m)	m	una cadena de caracteres de longitud fija de 1 a 255 caracteres	
varchar(m)	*	Una cadena de caracteres de longitud variable de 1 a 255 caracteres.	

* Para almacenarse en RAM o en disco una cadena de m caracteres, se necesitan m + 1 bytes.

Para representar cantidades con centavos, o calificaciones con un decimal debe usarse el tipo numeric(m,d). Los tipos float y double no representan exactamente las cifras decimales.

En el tipo timestamp el sistema almacena automáticamente la fecha y la hora del sistema. Esto es útil para averiguar en que momento se actualizó o insertó un registro.

Es preferible el tipo varchar que el tipo char para el atributo nombre. Por ejemplo, para almacenar el nombre "Pedro" se necesitan 6 bytes si se usa el tipo varchar(30), en cambio se requieren 30 con el tipo char(30).

En cambio es mejor usar char(1) que varchar(1) para el atributo sexo (m o f), en el primer caso se usa un byte de almacenamiento y en el segundo dos.

Para mayor información sobre los tipos de datos consulte el manual, el cual puede descargarse de Internet o también se encuentra en el CD distribuido por el profesor.

6. Trabajo a nivel de tablas

En esta sección usaremos la base de datos *nomina* con la tabla:

```
Empleado( idEmpleado, nombre, sueldo).
```

donde *idEmpleado* es la llave primaria.

Primero que todo se debe crear la base de datos, si ya no está creada, y activarla:

```
mysql> create database nomina;
mysql> use nomina;
```

La siguiente instrucción crea la tabla e indica que no se permitirán valores nulos para el atributo *idEmpleado*. La cláusula *not null* es obligatoria para los atributos que formen parte de una llave primaria.

```
mysql> create table Empleado (
-> idEmpleado mediumint unsigned not null,
-> nombre varchar(20),
-> sueldo int unsigned,
-> primary key(idEmpleado));
```

Una lista de la tablas existentes en la base de datos se obtiene con:

```
mysql> show tables;
```

Una descripción de la tabla se obtiene con:

```
mysql> describe Empleado;
```

Con la anterior instrucción se obtiene:

```
+-----+-----+-----+-----+-----+-----+
| Field      | Type                               | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idEmpleado | mediumint(8) unsigned             |      | PRI | 0        |      |
| nombre     | varchar(20)                       | YES  |     | NULL     |      |
| sueldo     | int(10) unsigned                  | YES  |     | NULL     |      |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

Si deseara borrar la tabla use:

```
mysql> drop table Empleado;
```

En Windows, la creación de la tabla se refleja en la creación de tres archivos en la carpeta *c:\apache\mysql\data*:

```
Empleado.MYD
Empleado.MYI
```

Empleado.frm

Los archivos contendrán respectivamente los datos, los índices y el formato (diseño de registro o esquema) de la tabla. Un índice de una tabla es similar al índice de un libro, permite acceder los datos más rápidamente.

Para adicionar los atributos sexo y fecha de ingreso a la tabla use:

```
mysql> alter table Empleado add sexo char(1);
mysql> alter table Empleado add fechaIngreso date;
```

Para cambiar el atributo *sexo*, dejándole por nombre *sex* y además para especificar que no se permitirán valores nulos en este atributo, use:

```
mysql> alter table Empleado change sexo sex char(1) not null;
```

Para cambiarle el tipo al atributo *sueldo* use:

```
mysql> alter table Empleado modify sueldo mediumint unsigned;
```

Se puede suprimir un atributo:

```
mysql> alter table Empleado drop sueldo;
```

MySQL no exige que las tablas tengan llave primaria, puede dejar la tabla sin llave primaria especificada con:

```
mysql> alter table Empleado drop primary key;
```

Para especificar de nuevo la llave primaria use:

```
mysql> alter table Empleado add primary key(idEmpleado);
```

Se le puede cambiar el nombre a la tabla:

```
mysql> alter table Empleado rename Empleado2;
```

7. Trabajo con registros:

Para este apartado usaremos de ejemplo la tabla:

Estudiante (carne, nombre, edad)

Es de aclarar que generalmente no se usa el atributo edad sino el atributo fecha de nacimiento, ya que este último no se desactualiza. A veces se usa el atributo edad, calculando sus valores en lugar de ser ingresados.

Ahora veremos la forma de agregar un registro a la tabla, por ejemplo el registro del estudiante Pedro, de carné número 200 y de 15 años. Esto se logra con:

```
mysql> insert into Estudiante values (200, "Pedro", 15);
```

Un listado de los registros de la tabla se obtiene con la instrucción `select`, que se explicará con mayor profundidad posteriormente:

```
mysql> select * from Estudiante;
```

```
+-----+-----+-----+
| carne | nombre | edad |
+-----+-----+-----+
|   200 | Pedro  |   15 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

No se preocupe por la presentación de la salida, más adelante el lenguaje HTML nos ayudará en este aspecto.

Puede insertar dos o más registros con una sola instrucción insert. Ejemplo:

```
mysql> insert into Estudiante values (300,"Juan",16),
-> (400,'Ana',14);
```

Cuando un dato es desconocido o no aplica, se usar el valor nulo, null. Ejemplo, si la edad de Luz es desconocida se puede usar:

```
mysql> insert into Estudiante values (700,"Luz",null);
```

Para borrar un registro, por ejemplo el de "Pedro" use:

```
mysql> delete from Estudiante where nombre = "Pedro";
```

Puede suprimir todos los registros (lo cual es muy peligroso) con:

```
delete from Estudiante;
```

Es posible corregir un dato, sin necesidad de volver a entrar todo el registro. Para dejar en 17 la edad del estudiante de carne 300:

```
mysql> update Estudiante set edad = 17
-> where carne = 300;
```

Para incrementar en uno la edad de todos los estudiantes use:

```
mysql> update Estudiante set edad = edad + 1;
```

Puede cambiar a 19 la edad de TODOS los estudiantes con:

```
mysql> update Estudiante set edad = 19;
```

Para que los estudiantes con carné menor a 500 queden con edad de 15 años use:

```
mysql> update Estudiante set edad = 15
-> where carne < 500;
```

8. Consejo para usuarios de Windows

Es muy recomendable que desactive la opción de Windows "Ocultar extensiones para los tipos de archivos conocidos". Si no hace esto podría tener algunos problemas cuando practique en el computador los siguientes apartados.

Para lograr esto en Windows 98, haga clic en *inicio*, luego en *configuración*, posteriormente en *opciones de carpeta*, a continuación en *ver*, y finalmente desactive la casilla correspondiente.

En Windows ME, haga clic en *inicio*, luego en *configuración*, después en *panel de control*, posteriormente en *opciones de carpeta*, a continuación en *ver*, y finalmente desactive la casilla correspondiente.

9. Importación de datos

Es posible cargar a una tabla MySQL datos que se encuentren en un archivo de texto. Esto permite hacer diversas migraciones de datos. Por ejemplo se podrían pasar los datos de un documento Word a un archivo de texto (.txt) y luego de aquí a una tabla MySQL.

Para esta sección utilizaremos la tabla:

Estudiante (carne, nombre, edad, fechaGrado)

Se usará también un archivo de texto con los registros que se muestran a continuación y con el supuesto nombre c:\datos\datos.txt:

105	María	12	2000-12-15
125	Martha	17	2001-10-18
236	Sofía Margarita	12	2000-11-15
405	Britney	17	2002-1-16
417	Cristina	23	\N
517	Rafael	\N	2000-2-1
612	Julio Alberto		\N\n
716	Nino	14	2000-12-2
815	Germán	15	1999-1-4
911	Cristian	16	1998-11-2
919	Shakira	23	\N

Puede crear un archivo de texto con el bloc de notas u otro editor de texto cualquiera pero teniendo en cuenta lo siguiente:

- Para separar un dato del siguiente, no presione la barra espaciadora ni una sola vez, en su lugar presione la tecla de tabulación (`⇧`) una y sólo una vez. No importa que los datos no queden bien alineados (ver el ejemplo).
- Use la tecla intro (o enter `↵`) para separar un renglón del siguiente
- Para valores nulos use `\N`, pero tenga en cuenta que la N debe ser mayúscula.
- Presione la tecla de tabulación después de los `\N` que se encuentren al final del renglón.

Para cargar los datos a la tabla utilice la siguiente instrucción:

```
mysql> load data local infile "c:/datos/datos.txt" into table estudiante;
```

Note que se utiliza la barra normal (/) y no la contraria (\) que se usa en MS-DOS.

Si los datos no fueron tecleados correctamente, se pueden remover todos los registros de la tabla con:

```
mysql> delete from Estudiante;
```

y hacer un nuevo intento.

10 Uso de MySQL como calculadora

La instrucción select permite evaluar expresiones algebraicas. Ejemplos:

a) `mysql> select 1+7*2;`

```
+-----+
| 1+7*2 |
+-----+
|    15 |
+-----+
1 row in set (0.01 sec)
```

b) `mysql> select 7%3, sqrt(25), floor(3.4), ceiling(3.9), sin(pi()/4);`

```
+-----+-----+-----+-----+-----+
| 7%3 | sqrt(25) | floor(3.4) | ceiling(3.9) | sin(pi()/4) |
+-----+-----+-----+-----+-----+
|    1 | 5.000000 |          3 |          4 |    0.707107 |
+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

El signo de porcentaje, al igual que en lenguaje C, representa la operación módulo (residuo). `floor` es el redondeo por defecto (piso) y `ceiling` es el redondeo por exceso (techo). Las funciones trigonométricas, como es usual, trabajan con el ángulo en radianes. En el ejemplo se está calculando el seno de 45 grados ya que :

$$\frac{\pi}{4} \text{radianes} = 45^\circ.$$

MySQL dispone de otras funciones trigonométricas y de funciones trigonométricas inversas, a saber: `sin`, `con`, `tan`, `cot`, `asin`, `acos`, `atan`.

Algunas funciones nos permiten obtener información del sistema. La siguiente instrucción nos informa respectivamente la versión de MySQL, el nombre de usuario de MySQL (que es diferente del usuario de Linux o de Windows), la fecha del sistema, la hora del sistema, la fecha-hora del sistema y el nombre de la base de datos activa (la última que se utilizó en una instrucción `use`).

```
mysql> select version(), user(), curdate(), curtime(), now(),
-> database();
```

Hasta ahora no hemos visto la manera de entrar a MySQL con un nombre de usuario, solo sabemos ingresar sin especificar un nombre de usuario, en este caso Windows asume el usuario `ODBC@localhost` y Linux asume el código de usuario Linux, por ejemplo: `root@localhost`.

MySQL incluye muchas funciones para trabajar con fechas, entre ellas la función `to_days`. Esta función convierte una fecha al número de días contados a partir de enero 1 de 0000. Ejemplo:

```
mysql> select to_days('2002-02-28');
```

```
+-----+
| to_days('2002-02-28') |
+-----+
|                731274 |
+-----+
1 row in set (0.03 sec)
```

Puede hallar el número de días que Ud. ha vivido con una instrucción similar a la siguiente:

```
mysql> select to_days(current_date()) - to_days('1983-10-1');
```

11. Ejercicio

La base de datos de una biblioteca incluye la tabla:

```
Libro ( idLibro, titulo, nombreAutor, valor, numeroEdicion,
fechaEdicion, prestado)
```

Los valores permitidos para el atributo prestado son 'S' y 'N', según el libro se encuentre prestado o no. Escriba las instrucciones necesarias para:

- Crear la tabla (Use el tipo de datos más apropiado para cada atributo).
- Agregarle un atributo a la tabla anterior (invéntese uno adecuado).
- Borrarle un atributo a la tabla anterior (escoja uno cualquiera).
- Agregar la cláusula not null a algún atributo que no lo tenga (al que Ud. quiera).
- Cambiarle el nombre a la tabla (invéntese el nombre nuevo).
- Insertarle tres registros a la tabla.
- Borrarle uno de los tres registros anteriores a la tabla.
- Actualizar un dato de algún registro.