

# Práctica 1

Alejandro Adolfo Perez Raya

Base de Datos

19 de agosto de 2013

# Índice

|  |           |
|--|-----------|
| <b>1. Marco Teórico</b>                  | <b>3</b>  |
| 1.1. Comando CREATE DATABASE . . . . .   | 3         |
| 1.2. Comando USE . . . . .               | 3         |
| 1.3. Comando SHOW DATABASES . . . . .    | 3         |
| 1.4. Comando SHOW TABLES . . . . .       | 3         |
| 1.5. Comando SHOW CREATE TABLE . . . . . | 3         |
| 1.6. Comando DESC/DESCRIBE . . . . .     | 3         |
| 1.7. Comando DROP TABLE . . . . .        | 4         |
| 1.8. Comando DROP DATABASE . . . . .     | 4         |
| 1.9. Comando SELECT DATABASE . . . . .   | 4         |
| 1.10. Comando CREATE TABLE . . . . .     | 4         |
| 1.11. Comando ALTER TABLE . . . . .      | 4         |
| 1.12. Hacer un respaldo . . . . .        | 5         |
| <b>2. Instrucciones</b>                  | <b>5</b>  |
| <b>3. Sentencias SQL y Screenshots</b>   | <b>5</b>  |
| <b>4. Conclusiones</b>                   | <b>11</b> |

## 1. Marco Teórico

### 1.1. Comando CREATE DATABASE

EL comando **CREATE DATABASE** crea una base de datos con el nombre dado. Su estructura es la siguiente:

```
CREATE DATABASE nombreBasedeDatos;
```

### 1.2. Comando USE

El comando **USE** le dice a MySQL que use una base de datos como la base de datos por defecto para los comandos siguientes. Sigue siendo la base de datos por defecto hasta el final de la sesión o hasta que se realiza otro comando USE. Su estructura es la siguiente:

```
USE nombreBaseDeDatos;
```

### 1.3. Comando SHOW DATABASES

El comando **SHOW DATABASES** enlista las bases de datos existentes en el servidor MySQL. Su estructura es la siguiente:

```
SHOW DATABASES;
```

### 1.4. Comando SHOW TABLES

El comando **SHOW TABLES** enlista las tablas existentes en la base de datos que se esta ocupando. Su estructura es la siguiente:

```
SHOW TABLES;
```

### 1.5. Comando SHOW CREATE TABLE

El comando **SHOW CREATE TABLE** muestra un comando CREATE TABLE el cual fue ocupado para crear la tabla dada. Su estructura es la siguiente:

```
SHOW CREATE TABLE nombreTabla;
```

### 1.6. Comando DESC/DESCRIBE

El comando **DESC/DESCRIBE** proporciona información acerca de las columnas en una tabla. Es una abreviación de SHOW COLUMNS FROM. Su estructura es la siguiente:

```
DESC/DESCRIBE nombreTabla;
```

## 1.7. Comando DROP TABLE

El comando **DROP TABLE** borra una o más tablas. Debe tener el permiso DROP para cada tabla. Todos los datos de la definición de tabla son borrados, así que tenga cuidado con este comando. Su estructura es la siguiente:

```
DROP TABLE nombreTabla;
```

## 1.8. Comando DROP DATABASE

El comando **DROP DATABASE** borra todas las tablas en la base de datos y borra la base de datos. Sea muy cuidadoso con este comando. Para usar DROP DATABASE, necesita el permiso DROP en la base de datos. Su estructura es la siguiente:

```
DROP DATABASE nombreBaseDeDatos;
```

## 1.9. Comando SELECT DATABASE

El comando **SELECT DATABASE** permite ver el nombre de la base de datos que se está ocupando. Su estructura es la siguiente:

```
SELECT DATABASE();
```

## 1.10. Comando CREATE TABLE

El comando **CREATE TABLE** crea una tabla con el nombre dado. Por defecto, la tabla se crea en la base de datos actual. Ocurre un error si la tabla existe, si no hay base de datos actual o si la base de datos no existe. Su estructura es la siguiente:

```
CREATE TABLE nombreTabla(campo tipo(tamaño),...);
```

## 1.11. Comando ALTER TABLE

El comando **ALTER TABLE** le permite cambiar la estructura de una tabla existente. Por ejemplo, puede añadir o borrar columnas, crear o destruir índices, cambiar el tipo de columnas existentes, o renombrar columnas o la misma tabla. Puede cambiar el comentario de la tabla y su tipo.

- Cambiar el nombre de una tabla o relación.

```
ALTER TABLE nombreTabla RENAME AS newNombreTabla;
```

- Cambiar el nombre de una columna.

```
ALTER TABLE nombreTabla CHANGE COLUMN  
nombreColumna newNombreColumna tipo(tamaño);
```

- Cambiar tipo de dato.

*ALTER TABLE nombreTabla MODIFY COLUMN nombreColumna newTipo(tamaño);*

- Eliminar columna.

*ALTER TABLE nombreTabla DROP COLUMN nombreColumna;*

### 1.12. Hacer un respaldo

Para crear un respaldo de la base de datos primero tendremos que ubicarnos a travez de la linea de comandos en la carpeta bin de MySQL, después ejecutamos el siguiente comando:

*mysqldump -u root -p nombreBaseDeDatos >  
DireccionDeLaCarpetaDondeQuiereHacerElRespaldo nombre.sql*

## 2. Instrucciones

- Crear una base de datos para un Club.
- Crear las siguientes relaciones (tablas).
  - Proveedor(idProveedor,nombre,dir,email).
  - Asociado(idAsociado,nombre,dir,email,salario).
  - Club(idClub,nombre,direccion,webpage,tel).
  - ClubProveedor(idclub,idProveedor).
  - ClubAsociado(idAsociado,idclub,fechaIngreso).
- **Nota:** El tipo de dato y tamaño sera elegido por cada quien, los id son llaves primarias, en el caso de las relaciones ClubProveedor y ClubAsociado definir las llaves foraneas.

## 3. Sentencias SQL y Screenshots

A continuación se muestran todos los comandos usados en la práctica.

- CREATE DATABASE practica1;
- USE practica1;
- CREATE TABLE Proveedor(
  - idProveedor int not null primary key,
  - nombre varchar(100),
  - dir varchar(200),
  - email varchar(50));

- CREATE TABLE Asociado(  
idAsociado int not null primary key,  
nombre varchar(100),  
dir varchar(200),  
email varchar(50),  
salario double);
- CREATE TABLE Club(  
idClub int not null primary key,  
nombre varchar(80),  
direccion varchar(230),  
webpage varchar(100),  
tel int(10));
- CREATE TABLE ClubProveedor(  
idClub int not null,  
idProveedor int not null,  
primary key(idClub,idProveedor),  
foreign key(idClub) references Club(idClub) on delete cascade on update cascade,  
foreign key(idProveedor) references Proveedor(idProveedor) on delete cascade on update cascade);
- create table ClubAsociado(  
idAsociado int not null,  
idClub int not null,  
primary key(idAsociado,idClub),  
fechaIngreso date,  
foreign key(idAsociado) references Asociado(idAsociado) on delete cascade on update cascade,  
foreign key(idClub) references Club(idClub) on delete cascade on update cascade);

A continuación se muestran algunos screenshots (los más representativos.)

- SHOW DATABASES;

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bdclub      |
| mysql       |
| phpmyadmin  |
| practical   |
| unidad1     |
| verbos      |
+-----+
7 rows in set (0.00 sec)
```

- SHOW TABLES;

```
mysql> show tables;
+-----+
| Tables_in_practical |
+-----+
| asociado             |
| clubasociado         |
| clubproveedor       |
| proveedor           |
| sucursal             |
+-----+
5 rows in set (0.00 sec)
```

- SHOW CREATE TABLE Asociado, SHOW CREATE TABLE ClubAsociado

```
C:\WINDOWS\system32\cmd.exe - mysql -p -u root

mysql> show create table asociado;
+-----+
| Table | Create Table |
+-----+
| asociado | CREATE TABLE `asociado` (
  `idAsociado` int(11) NOT NULL,
  `nombre` varchar(100) DEFAULT NULL,
  `dir` varchar(200) DEFAULT NULL,
  `email` varchar(50) DEFAULT NULL,
  `salario` double DEFAULT NULL,
  PRIMARY KEY (`idAsociado`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 |
+-----+
1 row in set (0.00 sec)

mysql> show create table clubasociado;
+-----+
| Table | Create Table |
+-----+
| clubasociado | CREATE TABLE `clubasociado` (
  `idAsociado` int(11) NOT NULL,
  `idclub` int(11) NOT NULL,
  `fechaIngreso` date DEFAULT NULL,
  PRIMARY KEY (`idAsociado`,`idclub`),
  KEY `idclub` (`idclub`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- SHOW CREATE TABLE ClubProveedor, SHOW CREATE TABLE Proveedor

```
C:\WINDOWS\system32\cmd.exe - mysql -p -u root
1 row in set (0.00 sec)
mysql> show create table clubproveedor;
+-----+-----+
| Table           | Create Table |
+-----+-----+
| clubproveedor | CREATE TABLE `clubproveedor` (
  `idclub` int(11) NOT NULL,
  `idProveedor` int(11) NOT NULL,
  PRIMARY KEY (`idclub`,`idProveedor`),
  KEY `idProveedor` (`idProveedor`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 |
+-----+-----+
1 row in set (0.00 sec)
mysql> show create table proveedor;
+-----+-----+
| Table           | Create Table |
+-----+-----+
| proveedor      | CREATE TABLE `proveedor` (
  `idProveedor` int(11) NOT NULL,
  `nombre` varchar(100) DEFAULT NULL,
  `dir` varchar(200) DEFAULT NULL,
  `email` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`idProveedor`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 |
+-----+-----+
1 row in set (0.00 sec)
```



- SHOW CREATE TABLE Sucursal

```
mysql> show create table sucursal;
+-----+-----+
| Table | Create Table |
+-----+-----+
| sucursal | CREATE TABLE `sucursal` (
  `idclub` int(11) NOT NULL,
  `nombre` varchar(80) DEFAULT NULL,
  `direccion` varchar(230) DEFAULT NULL,
  `webpage` varchar(100) DEFAULT NULL,
  `tel` int(10) DEFAULT NULL,
  PRIMARY KEY (`idclub`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

- DESC/DESCRIBE

```

C:\WINDOWS\system32\cmd.exe - mysql -p -u root

mysql> desc asociado;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idAsociado | int(11) | NO | PRI | NULL | |
| nombre | varchar(100) | YES | | NULL | |
| dir | varchar(200) | YES | | NULL | |
| email | varchar(50) | YES | | NULL | |
| salario | double | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)

mysql> desc clubasociado;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idAsociado | int(11) | NO | PRI | NULL | |
| idclub | int(11) | NO | PRI | NULL | |
| fechaIngreso | date | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

mysql> desc clubproveedor;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idclub | int(11) | NO | PRI | NULL | |
| idProveedor | int(11) | NO | PRI | NULL | |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)

mysql> desc proveedor
-> ;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idProveedor | int(11) | NO | PRI | NULL | |
| nombre | varchar(100) | YES | | NULL | |
| dir | varchar(200) | YES | | NULL | |
| email | varchar(50) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

mysql> desc sucursal;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idclub | int(11) | NO | PRI | NULL | |
| nombre | varchar(80) | YES | | NULL | |
| direccion | varchar(230) | YES | | NULL | |
| webpage | varchar(100) | YES | | NULL | |
| tel | int(10) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)

mysql>

```

## 4. Conclusiones

Esta práctica me sirvió para darme cuenta y aprender como se crea una base de datos así como sus registros. Todos los comando que se vieron en la práctica nos serán de gran ayuda en proximas prácticas y para el proyecto final. Hay que tener mucho cuidado al declarar las llaves foraneas ya que los nombres y referencias deben de coincidir de lo contrario habría un error. También el comando para hacer los respaldos es muy importante por que cuando estemos haciendo nuestro proyecto final podremos realizar respaldos facilmente.

## Referencias

- [1] *Sintaxis de CREATE DATABASE*. Recuperando 19 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/create-database.html>
- [2] *Sintaxis de USE*. Recuperando 19 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/use.html>
- [3] *Sintaxis de SHOW DATABASES*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/show-databases.html>
- [4] *Sintaxis de SHOW TABLES*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/show-tables.html>
- [5] *Sintaxis de SHOW CREATE TABLE*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/show-create-table.html>
- [6] *Sintaxis de DESCRIBE (Informacin acerca de las columnas)*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/describe.html>
- [7] *Sintaxis de DROP TABLE*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/drop-table.html>
- [8] *Sintaxis de DROP DATABASE*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/drop-database.html>
- [9] *Sintaxis de CREATE TABLE*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/create-table.html>
- [10] *Sintaxis de ALTER TABLE*. Recuperando 20 de agosto de 2013.[En línea]. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/alter-table.html>