

BASES DE DATOS

Desarrollo de Aplicaciones Web

GESTIÓN DE BASES DE DATOS

Administración de Sistemas Informáticos en Red

PRÁCTICA GUIADA DDL I



Práctica

Luis Dorado Garcés

Basado en el trabajo de Alba Tortosa López

1 Agenda

Queremos almacenar los datos de nuestros amigos en una agenda.

1. Creamos una base de datos "agenda"

```
CREATE DATABASE agenda;
```

2. Mostramos la lista de bases de datos. ¿Cuáles hay? ¿Para qué sirven?

```
SHOW DATABASES;
```

3. Volvemos a crear la misma base de datos.

- a. ¿Qué ocurre?

Nos salta un error ya que la base de datos ya existe.

```
Error Code: 1007. Can't create database 'agenda'; database exists
```

- b. ¿Cómo lo solucionamos?

Debemos usar otro nombre para la nueva BD. Para próximas ocasiones podemos usar la cláusula IF NOT EXISTS. Así se provocará una advertencia (warning) en lugar de un error.

```
CREATE DATABASE IF NOT EXISTS agenda;

1 row(s) affected, 1 warning(s): 1007 Can't create database 'agenda'; database exists
```

4. ¿Podemos tener dos bases de datos con el mismo nombre?

No, ya que no habría manera de distinguirlas dentro del SGBD.

5. Seleccionamos una base de datos y mostramos las tablas que tiene.

```
USE agenda;

SHOW TABLES;
```

6. Creamos la tabla "amigos1" en nuestra base de datos para almacenar:

- ✓ Nombre y apellidos (clave primaria)
- ✓ Teléfono (no nulo)
- ✓ Email
- ✓ Fecha de cumpleaños
- ✓ Tipo (estudios, vecino, trabajo, otros)

```
CREATE TABLE amigos1 (
    nombre_apellidos VARCHAR(50),
    teléfono NUMERIC(9,0) NOT NULL,
    email VARCHAR(30),
    fecha_cumple DATE,
    tipo ENUM('estudios','vecino','trabajo','otros'),
    PRIMARY KEY (nombre_apellidos)
```

```
);
```

7. Creamos la tabla "amigos2" en nuestra base de datos para almacenar:

- ✓ Nombre (clave primaria)
- ✓ Apellidos (clave primaria)
- ✓ Teléfono (no nulo)
- ✓ Email
- ✓ Fecha de cumpleaños
- ✓ Tipo (estudios, vecino, trabajo, otros). Por defecto será 'estudios'.

```
CREATE TABLE amigos2 (
    nombre VARCHAR(30),
    apellidos VARCHAR(30),
    teléfono NUMERIC(9,0) NOT NULL,
    email VARCHAR(30),
    fecha_cumple DATE,
    tipo enum('estudios','vecino','trabajo','otros')
        DEFAULT 'estudios',
    PRIMARY KEY (nombre, apellidos)
);
```

8. Mostramos la lista de tablas.

```
SHOW TABLES;
```

	Tables_in_agenda
►	amigos1
	amigos2

9. Volvemos a crear la tabla amigos2.

a. ¿Qué ocurre?

Nos salta un error ya que la tabla ya existe.

```
Error Code: 1050. Table 'amigos2' already exists
```

b. ¿Cómo lo solucionamos?

Debemos usar otro nombre para la nueva tabla. Para próximas ocasiones podemos usar también la cláusula IF NOT EXISTS. Así se provocará una advertencia (warning) en lugar de un error.

```
CREATE TABLE IF NOT EXISTS amigos2;
```

```
0 row(s) affected, 1 warning(s): 1050 Table 'amigos2' already exists
```

```
'agenda.agenda1'
```

10. ¿Podemos tener dos tablas con el mismo nombre?

No, ya que no habría manera de distinguirlas dentro la BD.

11. Mostramos la definición de la tabla "amigos1".

```
describe amigos1;
```

	Field	Type	Null	Key	Default	Extra
►	nombre_apellidos	varchar(50)	NO	PRI	NULL	
	teléfono	decimal(9,0)	NO		NULL	
	email	varchar(30)	YES		NULL	
	fecha_cumple	date	YES		NULL	
	tipo	enum('estudios','vecino','trabajo','otros')	YES		NULL	

12. Borrarnos la tabla "amigos1" y mostramos la lista de tablas para comprobar que ya no está.

```
DROP TABLE amigos1;
```

```
SHOW TABLES;
```

	Tables_in_agenda
►	amigos2

13. Volvemos a borrar la tabla "amigos1".

- a. ¿Qué ocurre?

Nos salta un error ya que la tabla no existe.

```
Error Code: 1051. Unknown table 'agenda.amigos1'
```

- b. ¿Cómo lo solucionamos?

Para próximas ocasiones podemos usar también la cláusula IF EXISTS. Así se provocará una advertencia (warning) en lugar de un error.

```
DROP TABLE IF EXISTS amigos1;
```

```
0 row(s) affected, 1 warning(s): 1051 Unknown table 'agenda.amigos1'
```

14. Borrarnos la base de datos "agenda" y mostramos la lista de bases de datos para comprobar que ya no está.

```
DROP DATABASE agenda;
```

```
SHOW DATABASES;
```

	Database
►	information_schema
	mysql
	performance_schema
	pizzeria

15. Volvemos a borrar la base de datos "agenda".

a. ¿Qué ocurre?

Nos salta un error ya que la BD no existe.

```
Error Code: 1008. Can't drop database 'agenda'; database  
doesn't exist
```

b. ¿Cómo lo solucionamos?

Para próximas ocasiones podemos usar también la cláusula IF EXISTS. Así se provocará una advertencia (warning) en lugar de un error.

```
DROP DATABASE IF EXISTS agenda;
```

```
0 row(s) affected, 1 warning(s): 1008 Can't drop database  
'agenda'; database doesn't exist
```