

3.7. Borrado de bases de datos

El comando DROP DATABASE es el utilizado para eliminar bases de datos. En MySQL, el comando se acompaña del nombre de la base de datos a eliminar:

```
>mysql -u root -p
drop database Proveedores;
```

En Oracle, no hay que especificarlo, tan solo es necesario conectarse a la instancia y escribir:

```
>sqlplus / as sysdba
shutdown abort; --parada de la instancia
startup mount exclusive restrict; --reinicio en modo exclusivo
drop database; --borrado
exit; --salir
```

Otra forma menos elegante de borrar una base de datos consiste en borrar físicamente todos los ficheros binarios de la base de datos.

3.8. Creación de tablas

Para crear tablas se usa el comando CREATE TABLE:

```
CREATE [TEMPORARY] TABLE [esquema.]nombre_tabla
  [(definición_create,...)]
  [opciones_tabla]
```

definición_create:

```
  definición_columna
  | [CONSTRAINT [símbolo]] PRIMARY KEY (nombre_columna,...)
  | [CONSTRAINT [símbolo]] FOREIGN KEY (nombre_columna,...)
    [definición_referencia]
```

definición_columna:

```
  nombre_columna tipo_datos [NOT NULL | NULL] [DEFAULT valor]
  [UNIQUE [KEY] | [PRIMARY] KEY]
  [definición_referencia]
```

definición_referencia:

```
  REFERENCES nombre_tabla [(nombre_columna,...)]
    [ON DELETE {CASCADE | SET NULL | NO ACTION} ]
    [ON UPDATE {CASCADE | SET NULL | NO ACTION} ]
```

Para terminar, cada gestor de base de datos efectúa sus propias modificaciones al formato de la sintaxis `create table`. La cláusula *opciones_tabla* permite especificar las peculiaridades de cada gestor con respecto al almacenamiento en soporte físico de sus tablas. Además, cada gestor incorpora diversas características, por ejemplo, Oracle y DB2 implementan tipos de datos distintos a MySQL o a SQL Server y Access.

3.8.2. Tipos de Datos

Los tipos de datos que pueden usarse tanto para MySQL como para Oracle en la definición de una columna son los siguientes:

Tipo de dato	Naturaleza	Tamaño/formato	MySQL	Oracle
TINYINT [UNSIGNED]	Entero	1 byte	X	X
SMALLINT [UNSIGNED]	Entero	2 bytes	X	X
MEDIUMINT [UNSIGNED]	Entero	3 bytes	X	X
INT [UNSIGNED]	Entero	4 bytes	X	X
BIGINT [UNSIGNED]	Entero	8 bytes	X	X
INTEGER [UNSIGNED]	Entero	4 bytes	X	X
DOUBLE [UNSIGNED]	Real Aproximado	8 bytes	X	X
FLOAT [UNSIGNED]	Real Aproximado	4 bytes	X	X
DECIMAL(longitud,decimales)	Real Exacto	Variable	X	
NUMERIC(longitud,decimales)	Real Exacto	Variable	X	
NUMBER(longitud[,decimales])	Real Exacto	Variable		X
DATE	Fecha	'aaaa-mm-dd'	X	X
TIME	Hora	'hh:mm:ss'	X	
TIMESTAMP	Fecha y Hora	'aaaa-mm-dd hh:mm:ss'	X	X
DATETIME	Fecha y hora	'aaaa-mm-dd hh:mm:ss'	X	
CHAR(longitud)	caracteres	Longitud Fija	X	X
VARCHAR(longitud)	caracteres	Longitud Variable	X	X
VARCHAR2(longitud)	caracteres	Longitud Variable		X
BLOB	Objetos binarios	Longitud Variable	X	X
TEXT	Campos Memo	Longitud Variable	X	
CLOB	Campos Memo	Longitud Variable		X
ENUM(valor1,valor2,valor3...)	Enumeraciones	Lista de valores	X	
SET(valor1, valor2, valor3...)	Conjuntos	Conjuntos de valores	X	

Cuadro 3.1: Tipos de datos en MySQL y Oracle/DB2.

Puede verse que no todos los tipos de datos están en todos los gestores. Así por ejemplo, el tipo de datos `ENUM` no está disponible en Oracle y sí en MySQL. Se utiliza para crear enumeraciones, es decir, campos que admiten solo valores fijos, por ejemplo *Color* `ENUM('rojo', 'amarillo', 'verde')`. En Oracle, en su lugar, se puede implementar utilizando la directiva `CHECK`.

DESCRIBE [esquema.]nombre_tabla

Este comando muestra un listado con las columnas de la tabla, aportando información sobre los tipos de datos, restricciones, etc.

```
SQL> describe nba.equipos;
```

Nombre	Nulo	Tipo
NOMBRE	NOT NULL	VARCHAR2(20)
CIUDAD		VARCHAR2(20)
CONFERENCIA		VARCHAR2(4)
DIVISION		VARCHAR2(9)

3.9. Modificación de tablas

El comando para modificar una tabla es ALTER TABLE y su sintaxis también es bastante compleja:

```
ALTER TABLE nombre_tabla
```

```
    especificación_alter [, especificación_alter] ...
```

especificación_alter:

```
    ADD definición_columna [FIRST | AFTER nombre_columna ]
| ADD (definición_columna,...)
| ADD [CONSTRAINT [símbolo]]
    PRIMARY KEY (nombre_columna,...)
| ADD [CONSTRAINT [símbolo]]
    UNIQUE (nombre_columna,...)
| ADD [CONSTRAINT [símbolo]]
    FOREIGN KEY (nombre_columna,...)
    [definición_referencia]
| CHANGE [COLUMN] anterior_nombre_columna definición_columna
    [FIRST|AFTER nombre_columna]
| RENAME COLUMN anterior_nombre_columna TO nuevo_nombre_columna
| MODIFY definición_columna [FIRST | AFTER nombre_columna]
| DROP COLUMN nombre_columna
| DROP PRIMARY KEY
| DROP FOREIGN KEY fk_símbolo
| opciones_tabla
```