



Universidad Tecnológica Nacional
Instituto Nacional Superior del Profesorado Técnico

Estructura y Base de Datos

Prof.: Juan Capia

Lenguaje de definición de datos DDL

El lenguaje de definición de datos (en inglés Data Definition Language, o DDL), es el que se encarga de la modificación de la estructura de los objetos de la base de datos. Incluye órdenes para modificar, borrar o definir las tablas en las que se almacenan los datos de la base de datos. Existen cuatro operaciones básicas: CREATE, ALTER, DROP y TRUNCATE.

Comando	Descripción
CREATE	Utilizado para crear nuevas tablas, campos e índices
DROP	Empleado para eliminar tablas e índices
ALTER	Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.

CREATE

Este comando crea un objeto dentro del gestor de base de datos. Puede ser una base de datos, tabla, índice, procedimiento almacenado o vista.

Ejemplo (crear una tabla):

```
CREATE TABLE Empleado
(Id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
Nombre VARCHAR(50),
Apellido VARCHAR(50),
Direccion VARCHAR(255),
Ciudad VARCHAR(60),
Telefono VARCHAR(15),
Email VARCHAR (30),
Edad (2),
Actividad Específica (100),
Id_Cargo INT);
```



ALTER

Este comando permite modificar la estructura de un objeto. Se pueden agregar/quitar campos a una tabla, modificar el tipo de un campo, agregar/quitar índices a una tabla, modificar un Triggers, etc.

La sintaxis SQL para ALTER TABLE es:

ALTER TABLE "nombre tabla" `[Modificador];

[modificador] depende del tipo de modificación que deseamos realizar. Para los usos mencionados anteriormente, las instrucciones [modificador] son:

Agregar una columna: ADD "columna 1" "tipos de datos para columna 1"

Eliminar una columna: DROP "columna 1"

Cambiar el nombre de una columna: CHANGE "nombre antiguo de la columna" "nuevo nombre de la columna" "tipos de datos para la nueva columna".

Cambiar el tipo de datos para una columna: MODIFY "columna 1" "nuevo tipo de datos"

Ejemplo (agregar columna a una tabla):

```
# ALTER TABLE 'NOMBRE_TABLA' ADD NUEVO_CAMPO INT;  
# ALTER TABLE 'NOMBRE_TABLA' DROP COLUMN NOMBRE_COLUMNNA;
```

DROP

Este comando elimina un objeto de la base de datos. Puede ser una tabla, vista, índice, Triggers, función, procedimiento o cualquier otro objeto que el motor de la base de datos soporte. Se puede combinar con la sentencia ALTER.

Ejemplo:

```
# DROP TABLE 'NOMBRE_TABLA';  
# DROP SCHEMA 'ESQUEMA';  
# DROP DATABASE 'BASEDATOS';  
# DROP TABLE IF EXISTS 'NOMBRE_TABLA';
```



TRUNCATE

Este comando trunca todo el contenido de una tabla. La ventaja sobre el comando DROP, es que, si se quiere borrar todo el contenido de la tabla, es mucho más rápido, especialmente si la tabla es muy grande. La desventaja es que TRUNCATE sólo sirve cuando se quiere eliminar absolutamente todos los registros, ya que no se permite la cláusula WHERE. Si bien, en un principio, esta sentencia parecería ser DML (Lenguaje de Manipulación de Datos), es en realidad una DDL, ya que internamente, el comando TRUNCATE borra la tabla y la vuelve a crear y no ejecuta ninguna transacción.

Ejemplo:

```
# TRUNCATE TABLE 'NOMBRE_TABLA';
```



Ejercicio 1:

Supongamos utilizamos nuestra tabla "libro", definida con la siguiente estructura:

- código, int auto_increment, clave primaria,
- título, varchar (40) not null
- autor, varchar (30)
- editorial, varchar (20)
- precio, decimal (5,2)
- edición date

- Agregar el campo "cantidad", de tipo int NOT NULL
- Eliminar el campo "edición" .
- Modificar el tipo del campo "título" para poder almacenar una longitud de 30 caracteres y que no permita valores nulos.
- Agregar el campo ISBN carácter (20) e indexar el campo.

Ejercicio 2:

Disponemos de la siguiente Base de Datos para gestionar la información de los pubs de una determinada provincia.

PUB	TITULAR	EMPLEADO
COD_PUB	DNI_TITULAR	DNI_EMPLEADO
NOMBRE	NOMBRE	NOMBRE
LICENCIA_FISCAL	DOMICILIO	DOMICILIO
DOMICILIO	COD_PUB	
FECHA_APERTURA		
HORARIO		
COD_LOCALIDAD		
EXISTENCIAS	LOCALIDAD	PUB_EMPLEADO
COD_ARTICULO	COD_LOCALIDAD	COD_PUB
NOMBRE	NOMBRE	DNI_EMPLEADO
CANTIDAD		FUNCION
PRECIO		
COD_PUB		



Universidad Tecnológica Nacional

Instituto Nacional Superior del Profesorado Técnico

Se pide escribir los comandos SQL que permitan la creación de las tablas anteriores teniendo en cuenta las siguientes restricciones:

- Todos los valores son de tipo carácter excepto los campos FECHA_APERTURA (fecha) , CANTIDAD, PRECIO y COD_LOCALIDAD (numéricos).
- Los únicos campos que no son obligatorios son los campos DOMICILIO.
- Los valores del campo horario sólo pueden ser HOR1, HOR2 y HOR3.
- No es posible dar de alta EXISTENCIAS a precio 0.
- El campo función de la tabla PUB_EMPLEADO sólo puede tener los valores CAMARERO, SEGURIDAD, LIMPIEZA.
- Se ha de mantener la integridad referencial entre las tablas.
- Las claves primarias deberán determinarse o agregarse a su criterio.