**SQL**

**Comandos:**

**Create table** miprimerabasededatos; // permite crear tablas desde 0 con sus columnas, tipos y constrains. Ej:  
CREATE TABLE nombre (  
 nombre\_de\_la\_columna\_1 TIPO\_DE\_DATO CONSTRAINT,  
 nombre\_de\_la\_columna\_1 TIPO\_DE\_DATO CONSTRAINT  
)  
  
CREATE TABLE post (  
 id INT PRIMARY KEY AUTO\_INCREMENT,  
 titulo VARCHAR(200)  
 FOREIGN KEY (cliente\_id) REFERENCES clientes(id)  
)

Si creamos una columna con id foránea será necesario usar la sentencia **FOREIGN KEY** para aclarar a que tabla y a que columna hace referencia aquel dato. La tabla en cuestión debe existir ante de correr esta sentencia.  
  
---------

**TABLAS**  
  
**Drop table** // Borrara la tabla que le especifiquemos en la sentencia. EJ:  
**DROP TABLE IF EXIST películas;**

**Alter table** // permite alterar una tabla ya existente y va a operar con tres comandos:  
ADD: para agregar una columna  
MODIFY: para modificar una columna.  
DROP: para borrar una columna  
  
Ejemplos de ALTER TABLE  
  
**ALTER TABLE películas  
ADD rating DECIMAL (3,1) NOT NULL;***agrega la columna rating.*  
  
**ALTER TABLE películas**  
**MODIFY rating DECIMAL (4,1) NOT NUILL;***Modifica el decimal de la columna rating. Aunque el resto de configs de la tabla no se modifiquen, es necesario escribirlas en la sentencia.*  
  
**ALTER TABLE** **películas  
DROP rating;**  
*Borra la columna rating.*  
-----------

**FILAS**

**INSERT**

Se pueden agregar datos en una tabla especifica o en todas.  
  
Todas las columnas: si no aclaramos los nombres de cada columna se crearan en todas ellas. El orden en el que insertamos los valores, deberá ser el mismo orden que tengan asignadas las columnas en la tabla.  
  
INSERT INTO table\_name (columna\_1, columna\_2, columna\_3, …)  
VALUES (valor\_1, valor\_2, valor3, …);  
  
INSERT INTO artistas (id, nombre, rating)  
VALUES ( DEFAULT, ‘Shakira’, 1.0);  
  
Columnas especificas: aclaramos la tabla y luego escribimos el nombre de la o las columnas entre los paréntesis:  
  
INSERT INTO artistas (nombre)  
VALUES (‘Calle 13’)  
  
INSERT INTO artistas (nombre, rating)  
VALUES (‘Maluma’, 1.0)  
  
**UPDATE**UPDATE modificara los registros existentes de una tabla. Al igual que con DELTE, es importante no olvidar el WHERE cuando escribimos la sentencia, aclarando la condición.  
  
UPDATE nombre\_tabla  
SET columna\_1 = valor\_1, columna\_2 =valor\_2, …  
WHERE condición;  
  
UPDATE artistas  
SET nombre = ‘Charly Garcia’, rating = 1.0  
WHERE id = 1;  
  
  
**DELETE**  
Con DELETE podemos borrar información de un tabla. Es importante recordar utilizar siempre el WHERE en la sentencia para agregar la condición de cuales son las filas que queremos eliminar. Si no escribimos el código WHERE, estaríamos borrando toda la tabla y no un registro en particular.  
  
DELETE FROM nombre\_tabla WHERE condición;  
  
DELETE FROM artistas WHERE id=4;

-------------

**CONSULTA DE DATOS**



Son a través de la consulta SELECT

**SELECT \* FROM “lo\_que\_queremos\_consultar**”



Podemos seleccionar para consultar datos de columnas o tablas (¿?) especificas.:  
**SELECT title, rating, awards FROM movies**



Podemos ser mas específicos, por ejemplo queremos que solo consulte los rating con una puntuación de 10.  
SELECT rating \* 10



-------  
**WHERE:** para hacer filtros.  
EJ:  
Where rating > 5



Where awards > 3  
  
Podemos hacer varios filtros, EJ:  
  
SELECT \* FROM movies  
WHERE reléase\_date > “2000-01-01”  
AND awards > 2  
  
Tambien podríamos poner OR en vez de AND.  
  
Podemos tener algún criterio de búsqueda para ordenarloss con ORDER BY.  
EJ:  
  
SELECT first\_name, last\_name, rating FROM actors  
WHERE rating > 2  
ORDER BY first\_name  
  
Tambien podemos hacer que el orden sea descendiente.  
  
SELECT first\_name, last\_name, rating FROM actors  
WHERE rating > 2  
ORDER BY first\_name DESC  
  
Despues de escribir el order by podemos escribir un criterio de desempate si es que algún dato repite su valor.  
  
SELECT first\_name, last\_name, rating FROM actors  
WHERE rating > 2  
ORDER BY first\_name DESC, rating DESC



**OPERADORES**

* =   
  <  
  >  
  >=  
  <=  
  <> // diferente a  
  != // diferente a  
  IS NULL // es nulo  
  BETWEEN // entre dos valores  
  IN // lista de valores  
  LIKE // se ajusta a  
    
    
  ¿Qué es SQL?
* ¿Qué sentencias SQL existen?
* ¿Cómo creamos una base de datos? ¿Y una tabla?
* ¿Qué sucede si nos equivocamos al crear la estructura de una tabla?
* ¿Cómo se insertan datos en una tabla? ¿Cómo se borran?