**Conexión y base de datos**

* **Conectar al servidor MySQL** :

MySQL

-u usuario -p

* **Mostrar bases de datos disponibles**:

SHOW DATABASES;

* **Seleccionar una base de datos**:

USE nombre\_base\_datos;

**CARGAR BASE DE DATOS DESDE TERMINAL, TANTO WINDOWS COMO LINUX;**

Source “Base de Datos;

* **Crear una base de datos**:

CREATE DATABASE nombre\_base\_datos;

* **Eliminar una base de datos**:

DROP DATABASE nombre\_base\_datos;

**Tablas.**

* **Mostrar tablas en la base de datos actual**:

SHOW TABLES;

* **Crear una tabla**:

CREATE TABLE nombre\_tabla (

columna1 tipo\_dato,

columna2 tipo\_dato,

...);

* **Eliminar una tabla**:

DROP TABLE nombre\_tabla;

* **Mostrar la estructura de una tabla**:

DESCRIBE nombre\_tabla;

* **Renombrar una tabla**:

RENAME TABLE nombre\_tabla\_actual TO nuevo\_nombre;

**Consultas básicas.**

* **Seleccionar datos**:

SELECT columna1, columna2 FROM nombre\_tabla;

* **Seleccionar todos los datos**:

-SELECT \* FROM nombre\_tabla;

-select Marca, precio from salvadora.producto where Marca = "continental" and precio >"1";

**WHERE.**

**Busqueda con mas de 1 Condicional.**

-select Marca, precio from salvadora.producto where Marca = "continental" and precio >"1";

**(LIKE);**

**Dato que contenga algo después del carácter que indiquemos.**

**Es decir después de la C, si hay algún dato como Carlos, Camino, Culiacan, los va a encontrar.**

**-**select nombre, apellido from salvadora.cliente where apellido like "c%";

**(LIKE),**

**Datos en medio, entre 2 caracteres.**

-select \* from salvadora.proveedor where ID\_PROVEEDOR like "%0%";

**AISLAR DATOS (NOT WHERE)(LIKE);**

**Buscar los datos que no contengan lo que indicamos, en este caso todos los CI que no tengan 18.**

select \* from salvadora.cliente where CI not like "%18%";

**Cuando el siguiente dato solo es exactamente un carácter y añade el conjunto de los primeros 2, es decir, en este caso arrojaría la lista del 100 al 109.**

select \* from salvadora.producto where COD\_PRODUCTO like "%10\_%";

**Having.**

**Se usa en combinación con la clausula GROUP BY, para restringir los grupos de filas devueltas, solo a aquellos cuya condición es VERDADERA.**

**HAVING + GROUP BY + SUM.**

select marca, sum(stock) from salvadora.producto

group by marca

having sum(stock) >30 ;

**HAVING + GROUP BY + COUNT.**

select color, count(\*) from salvadora.producto

group by color

having count(\*) > 3 ;

**HAVING + GROUP BY + MIN.**

select marca, min(precio) from salvadora.producto

group by marca

having min(precio) < 1500 ;

**Between.**

select nombre,stock,precio from salvadora.producto where precio between 1500 and 3000;

**Aislar Datos (Between).**

select nombre,stock,precio from salvadora.producto where precio not between 1500 and 3000;

**BUSCAR DATOS EN ESPECIFICO (IN).**

**Numeros INT**

select nombre,stock,precio from salvadora.producto where precio in (1800, 2300, 3000) ;

**CHAR.  
select nombre,marca,color from salvadora.producto where color in ("gris", "rojo") ;**

* **Insertar datos**:

INSERT INTO nombre\_tabla (columna1, columna2) VALUES (valor1, valor2);

* **Actualizar datos**:

UPDATE nombre\_tabla SET columna1 = valor1 WHERE condicion;

* **Eliminar datos**:

DELETE FROM nombre\_tabla WHERE condicion;

delete from salvadora.producto where COD\_PRODUCTO = "100";

**Funciones Agregadas (SUM,MIN,COUNT,MAX).**

**SUM.**

**(Suma todos los precios de una columna).**

select sum(precio) from producto;

**COUNT.**

**CONTAR DATOS O REGISTROS.**

select count(ID\_PROVEEDOR) from proveedor;

**MIN.**

**VALOR MINIMO**

select min(precio) from producto;

**MAX.**

**VALOR MAXIMO.**

select max(precio) from producto;

**Group By**

**Recopila los datos en varios registros y agrupa los resultados en una o mas columnas.**

**Group by SUMA.**

**Agrupa y suma los precios totales por colores**

select color,sum(precio) from producto group by COLOR;

**GROUP BY COUNT.**

**Cuenta y agrupa los productos totales en este caso por colores.**

select color,count(\*) from producto group by COLOR;

**GROUP BY COUNT Y WHERE.**

**Agrupa por marca y cuenta cada marca que tenga el color negro.**

select marca,count(precio) from producto where color = "negro" group by marca;

**GROUP BY COUNT Y NOT WHERE.**

Busca y agrupa las que no tengan la condicional, es decir todas las marcas que no tengan gris.

**select marca,count(precio) from producto where not color = "gris" group by marca;**

**GROUP BY COUNT Y MIN.**

Agrupa por columna (Color) y te da el mínimo de esa columna (Precio)

**select color,min(precio) from producto group by color;**

**GROUP BY COUNT Y MAX.**

**select color,max(precio) from producto group by color;**

Agrupa por columna (Color) y te da el máximo de esa columna (Precio)

**Filtros y orden (Order By).**

**Se usa para ordenar los registros en un conjunto de resultados.**

* **Agregar una condición a la consulta**:

SELECT \* FROM nombre\_tabla WHERE columna = valor;

* **Ordenar resultados**:

SELECT \* FROM nombre\_tabla ORDER BY columna ASC; -- Orden ascendente

SELECT \* FROM nombre\_tabla ORDER BY columna DESC; -- Orden descendente

* **Limitar resultados**:

SELECT \* FROM nombre\_tabla LIMIT 10; -- Primeros 10 resultados

**Comandos intermedios de MySQL.**

**Gestión de índices**

* **Crear un índice**:

CREATE INDEX nombre\_indice ON nombre\_tabla (columna);

* **Eliminar un índice**:

DROP INDEX nombre\_indice ON nombre\_tabla;

**Uniones (JOINs).**

* **INNER JOIN**:

SELECT \* FROM tabla1

INNER JOIN tabla2 ON tabla1.columna = tabla2.columna;

* **LEFT JOIN**:

SELECT \* FROM tabla1

LEFT JOIN tabla2 ON tabla1.columna = tabla2.columna;

* **RIGHT JOIN**:

SELECT \* FROM tabla1

RIGHT JOIN tabla2 ON tabla1.columna = tabla2.columna;

**Funciones de agregación.**

* **Contar registros**:

SELECT COUNT(\*) FROM nombre\_tabla;

* **Obtener un valor máximo o mínimo**:

SELECT MAX(columna), MIN(columna) FROM nombre\_tabla;

* **Sumar valores**:

SELECT SUM(columna) FROM nombre\_tabla;

* **Promediar valores**:

SELECT AVG(columna) FROM nombre\_tabla;

**Subconsultas**

* **Subconsulta en cláusula WHERE**:

SELECT \* FROM nombre tabla WHERE columna = (SELECT columna FROM otra\_tabla WHERE condicion);

**Administración de usuarios.**

* **Crear un usuario**:

CREATE USER 'nombre\_usuario'@'localhost' IDENTIFIED BY 'contraseña';

* **Otorgar privilegios**:

GRANT ALL PRIVILEGES ON nombre\_base\_datos.\* TO 'nombre\_usuario'@'localhost';

* **Revocar privilegios**:

REVOKE privilegio ON nombre\_base\_datos.\* FROM 'nombre\_usuario'@'localhost';

* **Eliminar un usuario**:

DROP USER 'nombre\_usuario'@'localhost';

**Alterar**

**Alterar o Renombrar Tablas.**

Alter table cliente rename clientes;

**Eliminar columna** (1)

Alter table cliente drop column nombre;

**Eliminar columna (2 o mas)**

Alter table cliente drop column nombre, drop column apellido;

**Eliminar llave primaria;**

Alter table cliente drop primary key;

**Eliminar llave foránea;**

Alter table cliente drop foreign key;

**Añadir columna;**

Alter table clientes add column celular int (13);

**Añadir columna en cierta posicion;**

Alter table clientes add column celular int (13) after Apellido;

**Añadir columna en primera posicion; (FIRST)**

Alter table clientes add column celular int (13) First;

**Cambiar nombre de una columna; (CHANGE)**

Alter Table Cliente change name NOMBRE varchar(20);

**Cambiar el tipo de una coluna (MODIFY)**

Alter Table Cliente modify teléfono int (15) not null;

**Transacciones**

* **Iniciar una transacción**:

START TRANSACTION;

* **Confirmar cambios (commit)**:

COMMIT;

* **Revertir cambios (rollback)**:

ROLLBACK;