

## • Tipos de Consultas

### **Estructura de la Base de Datos:**

- CREATE DATABASE -> Crear Base de Datos
- DROP DATABASE -> Eliminar Base de Datos
- CREATE TABLE -> Crear Tablas
- ALTER TABLE -> Modificar Tablas
- DROP TABLE -> Eliminar Tablas

### **Consultando y Modificando los Datos:**

- SELECT -> Buscar
- INSERT -> Agregar
- UPDATE -> Modificar
- DELETE -> Eliminar

Tipos de datos

## • Tipos de Datos

<b>Char (tamaño)</b>	Entre 0 y 256
<b>Varchar (tamaño)</b>	Entre 0 y 65.535 caracteres
<b>Text</b>	Textos
<b>TinyInt (tamaño)</b>	-128 a 127
<b>Int (tamaño)</b>	-2147483648 a 2147483647
<b>BigInt (tamaño)</b>	-9223372036854775808 a 9223372036854775807
<b>Double (tamaño, d)</b>	Números grandes con coma
<b>Date</b>	2021-09-10
<b>DateTime</b>	2021-09-10 10:05:05
<b>TimeStamp</b>	2021-09-10 10:05:05

## • Creación de la Base de Datos

Agregar Base de Datos:

**CREATE DATABASE** sistema\_veterinaria;

Eliminar Base de Datos:

**DROP DATABASE IF EXISTS** sistema\_veterinaria;

Con IF EXISTS eliminaría la base de datos si existe, si no usamos éste comando devolvería un error

Si queremos crear la siguiente tabla

## • Estructura de la Base de Datos

ID	NOMBRE	APELLIDO	FECHA NACIMIENTO
1	Lucas	Moy	1998-04-02 18:00:00
2	Pepe	Perez	1999-05-03 18:00:00

El comando sería el siguiente

## • Estructura de la Base de Datos

Comando para la Creación de Tablas:

```
CREATE TABLE usuarios (
```

```
id int(11) NOT NULL AUTO_INCREMENT,
```

```
nombre varchar(80) NOT NULL,
```

```
apellido varchar(80) NOT NULL,
```

```
fecha_nacimiento DATETIME NULL,
```

```
PRIMARY KEY (id)
```

```
);
```

Hacemos que incremente el id en cada instancia

Llave primaria

Con NOT NULL, no permitiremos atributos vacíos, con NULL sí permitiremos

Comando para la Creación de Tablas:

```
DROP TABLE IF EXISTS usuarios;
```

Borrar tabla

Modificación, agregación y eliminación de columnas en una tabla

## • Estructura de la Base de Datos

Agregar Columnas:

```
ALTER TABLE usuarios ADD COLUMN telefono varchar(45) NULL;
```

Eliminar Columnas:

```
ALTER TABLE usuarios DROP telefono;
```

Modifica el tamaño del tipo de dato

Modificar Columnas:

```
ALTER TABLE usuarios ALTER COLUMN telefono varchar(35) NULL;
```



P.D: Si habian datos con un tamaño de 45 se recortarán!!!

## Agregar y eliminar filas en una tabla

## • Agregar y Eliminar Datos:

Comando para Agregar una Fila:

```
INSERT INTO usuarios (  
    nombre, apellido, fecha_nacimiento, telefono)  
VALUES ('Lucas', 'Moy', '1998-04-04 10:00:00', '1231234343');
```

Comando para Agregar una Fila:

```
DELETE FROM usuarios WHERE id = '1';
```

## Modificar datos en una fila en una tabla

## • Modificar Datos:

```
UPDATE usuarios  
SET nombre = 'Pepe',  
    apellido = 'Angelini'  
WHERE id = '1';
```



P.D: Al usuario con el id = 1, se cambiará el nombre y apellido a Pepe Angelini

## Asignación de la clave primaria y foránea

```
Query 1 x  
Limit to 1000 rows  
1 CREATE TABLE publicaciones (  
2     id int(11) NOT NULL AUTO_INCREMENT,  
3     autor_id int(11) NOT NULL,  
4     titulo varchar(80) NOT NULL,  
5     texto varchar(80) NOT NULL,  
6     PRIMARY KEY (id),  
7     FOREIGN KEY (autor_id) REFERENCES usuarios(id)  
8 );  
9
```

primary key

Creamos un atributo que hará como clave foránea para el id de la tabla con la que relacionaremos



Le decimos que autor\_id hará referencia al atributo id de la tabla usuarios

## Consultar datos en una tabla



## • CONSULTAR Datos:

Buscar todo los usuarios:

```
SELECT * FROM usuarios;
```

Buscar el Usuario con ID 1:

```
SELECT * FROM usuarios WHERE id = '1';
```

Buscar el nombre y apellido del Usuario con ID 1:

```
SELECT nombre, apellido  
FROM usuarios WHERE id = '1';
```

**SELECT:** Selecciona las columnas de una tabla

**FROM:** Selecciona las tablas de una base de datos

**WHERE:** Selecciona los datos de una o varias tuplas dependiendo de una condicional

Con \* traemos todas las columnas de una tabla o tablas

## Contamos la cantidad de filas de una consulta

The screenshot shows a database query editor with a toolbar at the top. The query text is as follows:

```
1 • SELECT COUNT(*) as 'Cantidad' FROM usuarios  
2 WHERE apellido = 'Moy'  
3 AND nombre = 'Lucas';
```

Below the query editor, the 'Result Grid' is displayed, showing a single row with the column header 'Cantidad' and the value '1'.

Cuenta la cantidad de filas que hay dependiendo de la consulta que hicimos

Con la clausula as renombramos el nombre de una columna pero no afecta al nombre real

## Consultas con el comando LIKE

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM usuarios
2 WHERE nombre LIKE 'Luc%'
```

En éste caso buscará todos los nombres que empiecen con Luc, como sigue el nombre no importará

Result Grid

	id	nombre	apellido	fecha_nacimiento	telefono
▶	3	Lucas	Moy	1998-04-04 10:00:00	1231234343

usuarios 26 x

Apply

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM usuarios
2 WHERE nombre LIKE '%uc%'
```

Nos buscará todos los nombres que contengan uc, no importará como empiece o finalice

Result Grid

	id	nombre	apellido	fecha_nacimiento	telefono
▶					

usuarios 27 x

Apply

## Consultas más complejas

# • Estructura de la Base de Datos

## Usuarios

ID	NOMBRE	APELLIDO	FECHA NACIMIENTO
1	Lucas	Moy	1998-04-02 18:00:00
2	Pepe	Perez	1999-05-03 18:00:00

## Publicaciones

ID	Autor_Id	Titulo	Texto
1	1	Curso de SQL	En este curso...
2	1	Curso de PHP	Curso para aprender...

## • CONSULTAR Datos:

```
SELECT p.*,  
CONCAT(u.nombre, ' ', u.apellido) as 'autor'  
FROM publicaciones p, usuarios u  
WHERE p.autor_id = u.id;
```

Selecciona todas las columnas de tabla publicaciones

Uniremos el nombre y el apellido en una sola columna con CONCAT

Acá consultamos a la tabla publicaciones con una abreviatura "p" y a usuarios con "u"

Nos traerá todas las publicaciones que hagan referencia y coincidencia con el id de la tabla usuarios

MALA PRÁCTICA, USAR INNER JOIN !!!

## Consultas con INNER JOIN

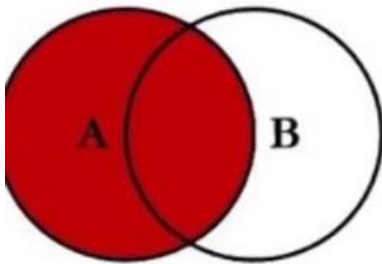
```
Query 1 x  
Limit to 1000 rows  
1 • SELECT p.id,  
2   p.titulo,  
3   p.texto,  
4   CONCAT(u.nombre, ' ', u.apellido) as 'autor'  
5 FROM publicaciones p INNER JOIN usuarios u  
6 ON p.autor_id = u.id;
```

Hacemos lo mismo que el anterior pero ésta consulta es mucho más eficiente

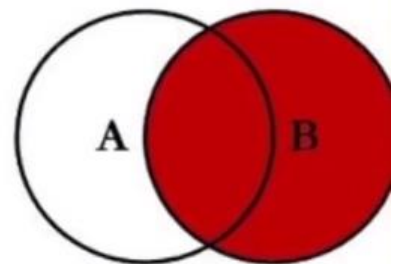
## Tipos de JOINS



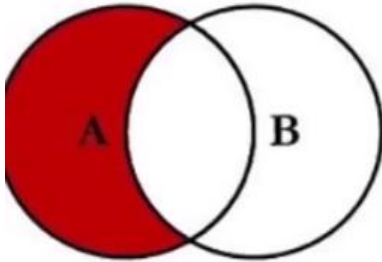
# SQL JOINS



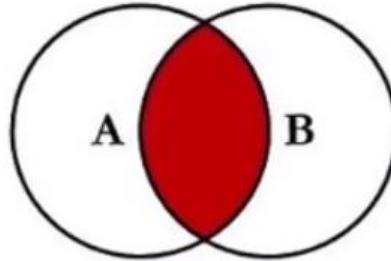
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



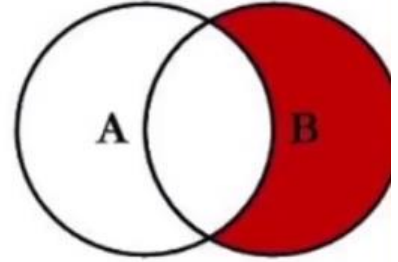
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



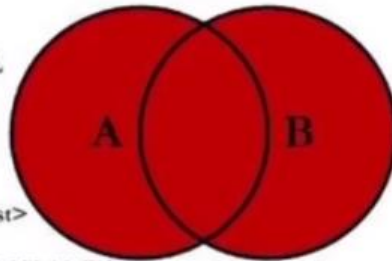
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



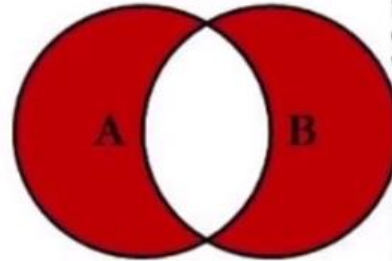
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

**FUNCIONES SQL PARA ESTADISTICA**

# • SQL para Estadísticas

FUNCIÓN	Descripción
<b>COUNT(*)</b>	Cuenta la cantidad de filas
<b>MAX(puntaje)</b>	Devuelve el máximo puntaje
<b>MIN(puntaje)</b>	Devuelve el mínimo puntaje
<b>AVG(puntaje)</b>	Devuelve el puntaje promedio
<b>SUM(precio)</b>	Devuelve una sumatoria de precios

## Uso de Group By

La sentencia GROUP BY identifica una columna seleccionada para utilizarla para agrupar resultados

The screenshot shows a SQL query editor interface. At the top, the query is: `1 • select * from curso.usuarios group by MONTH(fecha_nacimiento);`. Below the query, the results are displayed in a table with the following columns: `id`, `nombre`, `apellido`, `fecha_nacimiento`, and `telefono`. The first row of data shows: `2`, `Pepe`, `Moy`, `1998-04-04 10:00:00`, and `1231234343`. The interface includes a toolbar with various icons for editing and executing queries, and a status bar at the bottom showing the current table is `usuarios 29`.

id	nombre	apellido	fecha_nacimiento	telefono
2	Pepe	Moy	1998-04-04 10:00:00	1231234343