

Creación de las filas: se cargan los registros

## Terminología

### Citas

Pedro	20/12/20
Karen	21/12/21
Andrea	22/12/22

Creación de las columnas: tipo de datos que tiene la tabla.

## Terminología

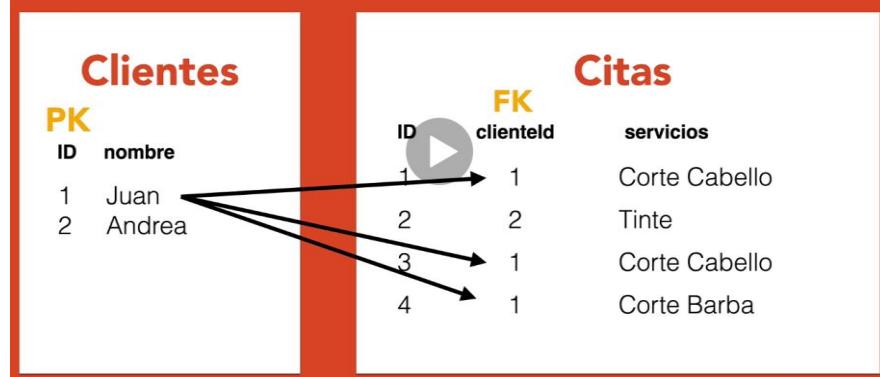
### Citas

Nombre	Teléfono	correo			
--------	----------	--------	--	--	--

Tipo de relaciones en una base de datos: 1 a muchos

Ejemplo: un cliente puede tener muchas citas.

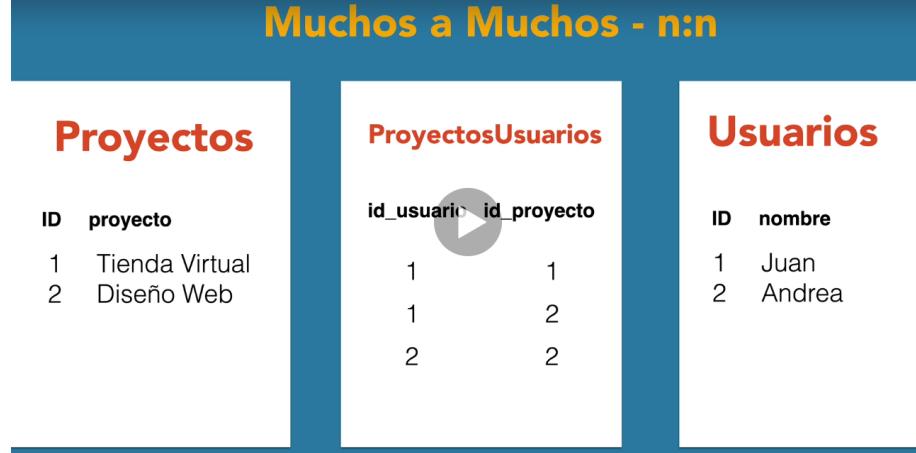
## 1 a Muchos - 1:n



Tipo de relación de: Muchos a Muchos.

Ejemplo: Muchos usuario pueden crear muchos productos.

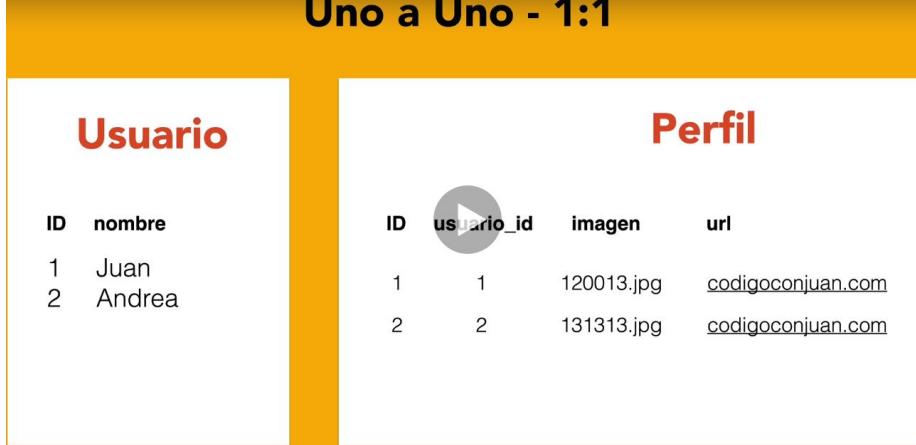
## Muchos a Muchos - n:n



Tipo de relación de: 1 a 1.

Ejemplo: un usuario puede tener un perfil.

## Uno a Uno - 1:1



## Los tipos de datos más comunes son.

Números (Enteros y Decimales)  
Cadenas de Texto (Desde Nombres hasta Noticias)  
Fechas y Horas

### Números y Enteros

TIPO	RANGO	UNSIGNED
INT	- 2100 Millones a 2100 Millones	0 a 4200+ Millones
TINYINT	-128 a 127	0 a 255
SMALLINT	-32,768 a 32,767	0 a 65,535
MEDIUMINT	-8.3 millones a 8.3 millones	0 a 16.7 millones
BIGINT	-2 <sup>63</sup> a 2 <sup>63</sup> -1	0 a 2 <sup>64</sup> -1

### Números con decimales

TIPO	USOS
FLOAT	Valores Decimales
DOUBLE	Valores Decimales muy grandes
DECIMAL	Valores donde no se acepta redondos (dinero)

## Texto y Datos Binarios

TIPO	USOS
VARCHAR	Texto corto, o texto que varia en su extensión (hasta 255 caracteres)
CHAR	Corto, extensión fija, (como un password encriptado)
TEXT	Texto Largo (como una entrada de blog)
ENUM	Un valor de una lista enumerada
BLOB	Imágenes, sonidos, y archivos comprimidos

## Fecha y Hora

TIPO	USOS
DATE	AAAA-MM-DD
TIME	hh:mm:ss
DATETIME	AAAA-MM-DD hh:mm:ss
TIMESTAMP	AAAA-MM-DD hh:mm:ss
YEAR	AAAA

# Los más comunes

TIPO	USOS
INT	Números Enteros
TINYINT	Números pequeños como edades
DECIMAL	Dinero y Medidas
VARCHAR	Textos Cortos
TEXT	Textos Largos
DATETIME	Fechas

Comandos Importantes en SQL:

- SHOW DATABASES;
- CREATE DATABASE servicios;
- USE servicios;
- CREATE TABLE servicios (
- id INT (11) NOT NULL AUTO\_INCREMENT,
- nombre VARCHAR(60) NOT NULL,
- precio DECIMAL(6,2) NOT NULL,
- PRIMARY KEY (id)
- );
- DESCRIBE servicios; ver la tabla formateada.

**Deberás dominar  
las 4 operaciones  
del CRUD**

- Gracias a SQL es posible realizar estas 4 operaciones.
- Casi todas las aplicaciones web o sitios web que realices deberán tener un CRUD.
- Primero deberás aprender a realizar el CRUD en SQL y después conectarlo con un lenguaje de backend.

## INSERTAR SERVICIOS

```
|mysql> INSERT INTO servicios (nombre, precio) VALUES ("Corte de Cabello de Adulto"mysql);
|      INSERT INTO servicios (nombre, precio) VALUES ("Corte de Cabello de Adulto", 80);
|Query OK, 1 row affected (0.00 sec)

|mysql> INSERT INTO servicios (nombre, precio) VALUES
|      -> ('Peinado Mujer',80),
|      -> ('Peinado Hombre', 60);
|Query OK, 2 rows affected (0.01 sec)
|Records: 2  Duplicates: 0  Warnings: 0
```

## LISTAR LOS SERVICIOS

### TRAE TODOS LOS SERVICIOS

```
mysql> SELECT * FROM servicios;
```

### TRAE SOLO LOS NOMBRES

```
|mysql> SELECT nombre FROM servicios;
|
|mysql> SELECT nombre, precio FROM servicios;
|
|mysql> SELECT id, nombre, precio FROM servicios;
|
|mysql> SELECT id, nombre, precio FROM servicios ORDER BY precio;
|
|mysql> SELECT id, nombre, precio FROM servicios ORDER BY precio ASC;
|
|mysql> SELECT id, nombre, precio FROM servicios ORDER BY precio DESC;
|
|mysql> SELECT id, nombre, precio FROM servicios ORDER BY id DESC;
|
|mysql> SELECT id, nombre, precio FROM servicios LIMIT 2;
|
|mysql> SELECT id, nombre, precio FROM SERVICIOS WHERE id = 3;
```

## ACTUALIZAR

```
|mysql> UPDATE servicios SET precio = 70 WHERE id = 2;
|
|mysql> UPDATE servicios SET nombre = "Corte de Cabello de Niño Actualizado" WHERE id = 2;
|
|mysql> UPDATE servicios SET nombre = "Corte de Cabello de Adulto ACTUALIZADO", precio = 120 WHERE id = 1;
```

## DELETE

```
|mysql> DELETE FROM servicios WHERE id = 1;
|
|mysql> DELETE FROM servicios WHERE id = 4;
```

## MODIFICAR LA BASE DE DATOS (TABLAS)

```
|mysql> ALTER TABLE servicios ADD descripcion VARCHAR(100) NOT NULL;
|
|mysql> ALTER TABLE servicios CHANGE descripcion nuevonombre VARCHAR(11) NOT NULL
|;
```

```
mysql> ALTER TABLE servicios CHANGE nuevonombre descripcion VARCHAR(11) NOT NULL;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE servicios DROP COLUMN descripcion;
Query OK, 0 rows affected (0.15 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

ELIMINAR TABLES Y AGREGAR OTRA TABLA

```
mysql> DROP TABLE servicios;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> CREATE TABLE reservaciones (
->     id INT(11) NOT NULL AUTO_INCREMENT,
->     nombre VARCHAR(60) NOT NULL,
->     apellido VARCHAR(60) NOT NULL,
->     hora TIME DEFAULT NULL,
->     fecha DATE DEFAULT NULL,
->     servicios VARCHAR(255) NOT NULL,
->     PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (0.03 sec)
```

SELECCIONAR SERVICIOS

```
mysql> SELECT * FROM servicios WHERE precio > 90;
mysql> SELECT * FROM servicios WHERE precio <= 80;
mysql> SELECT * FROM servicios WHERE precio = 80;
mysql> SELECT * FROM servicios WHERE precio BETWEEN 100 AND 200;
```

SELECTORES

```
mysql> SELECT COUNT(id), fecha
->     FROM reservaciones
->     GROUP BY fecha
->     ORDER BY COUNT(id) DESC;

mysql> SELECT SUM(precio) AS totalServicios FROM servicios;
mysql> SELECT MIN(precio) AS precioMenor FROM servicios;
mysql> SELECT MAX(precio) AS precioMenor FROM servicios;
```

BUSCAR EN SQL

```
mysql> SELECT * FROM servicios WHERE nombre LIKE 'Corte%';
mysql> SELECT * FROM servicios WHERE nombre LIKE 'Lavado%';
mysql> SELECT * FROM servicios WHERE nombre LIKE '%Cabello%';
```

```
|mysql> SELECT * FROM servicios WHERE nombre LIKE '%Lavado%';
```

COMO UNIR COLUMNAS FILTRADOS CON AND

```
|mysql> SELECT * FROM reservaciones WHERE fecha = "2021-06-28";
```

```
|mysql> SELECT * FROM reservaciones WHERE fecha = "2021-06-28" AND id = 1;
```

```
|mysql> SELECT * FROM reservaciones WHERE fecha = "2021-06-28" AND id IN(1,10);
```

```
|mysql> SELECT * FROM reservaciones WHERE fecha = "2021-06-28" AND id = 1 AND nombre = "Juan";
```

CONCATENAR UNIR DOS COLUMNAS

```
|mysql> SELECT CONCAT(nombre, ' ', apellido) AS nombreCompleto FROM reservaciones;
```

```
|mysql> SELECT * FROM reservaciones  
|      -> WHERE CONCAT(nombre, " ", apellido) LIKE '%Ana Preciado%';
```

```
mysql> SELECT hora, fecha, CONCAT(nombre, " ", apellido) as 'Nombre Completo'  
|      -> FROM reservaciones  
|      -> WHERE CONCAT(nombre, " ", apellido)  
|      -> LIKE '%Ana Preciado%';
```

Normalizacion de base de datos

**Primera Regla de Normalización: 1NF**

- Cada columna debe tener solo 1 valor y no debe haber grupos repetidos.

Problema

Id	nombre	servicios
1	Juan	Corte de Cabello
2	Pedro	Corte de Cabello, Barba
3	Karen	Peinado
4	Alejandra	Uñas
5	Raquel	Uñas, Tinte, Peinado

Solucion

Clientes		Citas	
Id	nombre	servicios	clientelId
1	Juan	Corte de Cabello	1
2	Pedro	Corte de Cabello	2
3	Karen	Corte de Barba	2
4	Alejandra	Peinado	3
5	Raquel	Uñas	4
		Uñas	5
		Tinte	5
		Peinado	5

## Segunda Regla de Normalización: 2NF

[www.codigoconjuan.com](http://www.codigoconjuan.com)

- Una vez aplicada la primer regla, pasamos a la segunda.
- Mientras que la primera nos dice que no debemos tener múltiples valores en una celda, la 2 y 3 se enfocan más en la relación con otras columnas.
- 2NF se utiliza en Llaves primarias compuestas.

## Citas

clienteld	fecha	hora	nombre
129019	2/10/2021	10:30	Juan
1313	3/10/2021	12:30	...
129019	5/10/2021	15:30	Juan
13114	8/10/2021	10:00	...
129019	12/10/2021	12:00	Juan
5112	22/10/2021	17:00	...
1414	28/10/2021	14:00	...
129019	24/10/2021	13:00	Juan

Llave primaria compuesta

Solucion

Clientes		Citas		
PK	FK	clienteld	fecha	hora
Id	nombre	129019	2/10/2021	10:30
129019	Juan	1313	3/10/2021	12:30
...	...	129019	5/10/2021	15:30
...	...	13114	8/10/2021	10:00
...	...	129019	12/10/2021	12:00
...	...	5112	22/10/2021	17:00
...	...	1414	28/10/2021	14:00
...	...	129019	24/10/2021	13:00

## Tercera Regla de Normalización:

3NF

- Al igual que 2NF tiene que ver con la relación de datos.
- Mientras que 2NF se enfoca en la llave compuesta, la 3NF se enfoca en los demás datos que no forman parte de la llave compuesta

Clientes		Citas		
Id	nombre	servicios	clienteld	hora
1	Juan	Corte de Cabello	1	10:30
2	Pedro	Corte de Cabello	2	12:30
3	Karen	Corte de Barba	2	12:30
4	Alejandra	Peinado	3	18:00
5	Raquel	Uñas	4	16:00
		Uñas	5	10:00
		Tinte	5	10:00
		Peinado	5	10:00

Solucion

Clientes		Citas		
Id	nombre	servicios	clienteld	hora
1	Juan	1	1	10:30
2	Pedro	1	2	12:30
3	Karen	2	2	12:30
4	Alejandra	3	3	18:00
5	Raquel	4	4	16:00
		4	5	10:00
		5	5	10:00
		3	5	10:00

Servicios	
Id	nombre
1	Corte de Cabello
2	Corte de Barba
3	Peinado
4	Uñas
5	Tinte

Hay que tener cuidado con las tablas de cobros

## Ciencia de Datos

### Clientes

Id	nombre
1	Juan
2	Pedro
3	Karen
4	Alejandra
5	Raquel

### Citas

servicios	clientelid	hora	total
1	1	10:30	\$50
1	2	12:30	\$200
2	2	12:30	\$300
3	3	18:00	\$240
4	4	16:00	\$220
4	5	10:00	\$230
5	5	10:00	\$120
3	5	10:00	\$80

### Servicios

Id	nombre
1	Corte de Cabello
2	Corte de Barba
3	Peinado
4	Uñas
5	Tinte

## Denormalización

- Algunas veces te encontrarás bases de datos que rompen las reglas de la normalización, no por eso significa que está mal.

```
mysql> SELECT * FROM reservaciones;
+----+-----+-----+-----+-----+-----+
| id | nombre | apellido | hora | fecha | servicios |
+----+-----+-----+-----+-----+-----+
| 1 | Juan | De la torre | 10:30:00 | 2021-06-28 | Corte de Cabello Adulto, Corte de Barba |
| 2 | Antonio | Hernandez | 14:00:00 | 2021-07-30 | Corte de Cabello Niño |
| 3 | Pedro | Juarez | 20:00:00 | 2021-06-25 | Corte de Cabello Adulto |
| 4 | Mireya | Perez | 19:00:00 | 2021-06-25 | Peinado Mujer |
| 5 | Jose | Castillo | 14:00:00 | 2021-07-30 | Peinado Hombre |
| 6 | Maria | Diaz | 14:30:00 | 2021-06-25 | Tinte |
| 7 | Clara | Duran | 10:00:00 | 2021-07-01 | Uñas, Tinte, Corte de Cabello Mujer |
| 8 | Miriam | Ibañez | 09:00:00 | 2021-07-01 | Tinte |
| 9 | Samuel | Reyes | 10:00:00 | 2021-07-02 | Tratamiento Capilar |
| 10 | Joaquin | Muñoz | 19:00:00 | 2021-06-28 | Tratamiento Capilar |
| 11 | Julia | Lopez | 08:00:00 | 2021-06-25 | Tinte |
| 12 | Carmen | Ruiz | 20:00:00 | 2021-07-01 | Uñas |
| 13 | Isaac | Sala | 09:00:00 | 2021-07-30 | Corte de Cabello Adulto |
| 14 | Ana | Preciado | 14:30:00 | 2021-06-28 | Corte de Cabello Mujer |
| 15 | Sergio | Iglesias | 10:00:00 | 2021-07-02 | Corte de Cabello Adulto |
| 16 | Aina | Acosta | 14:00:00 | 2021-07-30 | Uñas |
| 17 | Carlos | Ortiz | 20:00:00 | 2021-06-25 | Corte de Cabello Niño |
| 18 | Roberto | Serrano | 10:00:00 | 2021-07-30 | Corte de Cabello Niño |
```

```
mysql> SELECT * FROM servicios;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1  | Corte de Cabello Niño | 60.00 |
| 2  | Corte de Cabello Hombre | 80.00 |
| 3  | Corte de Barba        | 60.00 |
| 4  | Peinado Mujer         | 80.00 |
| 5  | Peinado Hombre         | 60.00 |
| 6  | Tinte                 | 300.00 |
| 7  | Uñas                  | 400.00 |
| 8  | Lavado de Cabello     | 50.00 |
| 9  | Tratamiento Capilar   | 150.00 |
+----+-----+-----+
```

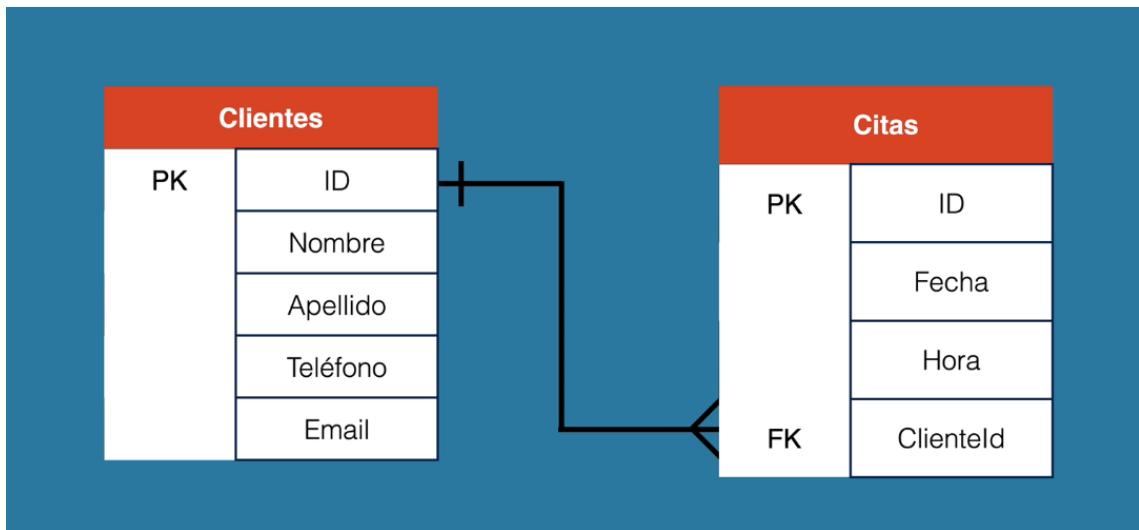
## Diagramas ER

Diagramas Entidad Relación

¿Para que sirven?

[www.codigoconjuan.com](http://www.codigoconjuan.com)

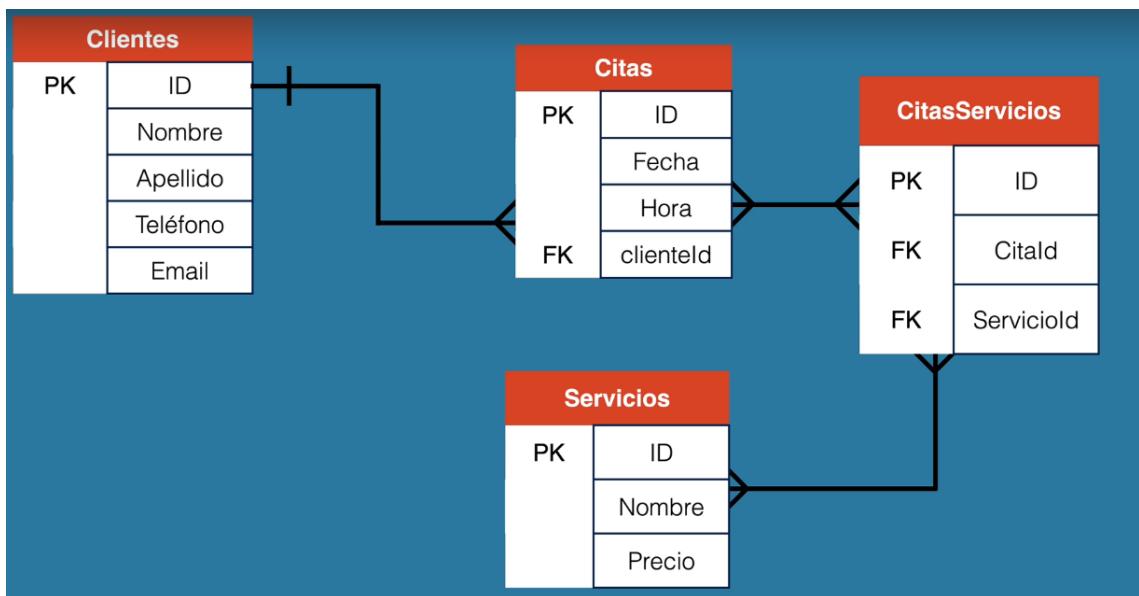
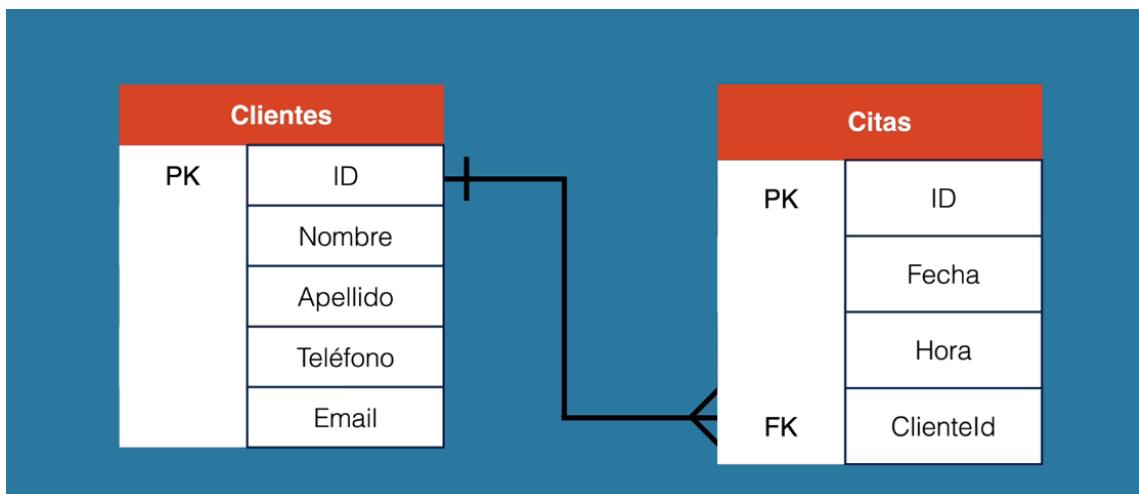
- Te darán una idea de forma gráfica de las Entidades (Tablas) y sus atributos.
- También ayudan a conocer como se relacionan los datos.
- Existen muchas herramientas de pago y gratis para generar estos diagramas.



## Cardinalidad

- La cardinalidad se refiere al número máximo de veces que una instancia se relaciona con otra.
- No pienses en esto como un número obligatorio, sino el máximo de veces que 2 tablas se pueden relacionar.

- |  |                    |
|--|--------------------|
|  | • Una              |
|  | • Muchas           |
|  | • Una (y solo una) |
|  | • Cero o Una       |
|  | • Una o Muchas     |
|  | • Cero o Muchas    |



```
mysql> CREATE TABLE clientes (
    -> id INT(11) NOT NULL AUTO_INCREMENT,
    -> nombre VARCHAR(60) NOT NULL,
    -> apellido VARCHAR(60) NOT NULL,
    -> telefono VARCHAR(10) NOT NULL,
    -> email VARCHAR(30) NOT NULL UNIQUE,
    -> PRIMARY KEY (id)
    -> );
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> SHOW TABLES;
+-----+
| Tables_in_appsalon |
+-----+
| clientes           |
| servicios          |
+-----+
2 rows in set (0.00 sec)
```

```
mysql> INSERT INTO clientes (nombre, apellido, telefono, email) VALUES
-> ("Juan", "De la torre", "1839813910", "correo@correo.com");
Query OK, 1 row affected (0.01 sec)
```

#### Relación tablas

```
mysql> CREATE TABLE citas (
-> id INT(11) NOT NULL AUTO_INCREMENT,
-> fecha DATE NOT NULL,
-> hora TIME NOT NULL,
-> clienteId INT(11) NOT NULL,
-> PRIMARY KEY (id),
-> KEY clienteId (clienteId),
-> CONSTRAINT cliente_FK
-> FOREIGN KEY (clienteId)
-> REFERENCES clientes (id)
-> );
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> DESCRIBE clientes;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| id    | int(11) | NO   | PRI | NULL    | auto_increment |
| nombre | varchar(60) | NO   |     | NULL    |               |
| apellido | varchar(60) | NO   |     | NULL    |               |
| telefono | varchar(10) | NO   |     | NULL    |               |
| email  | varchar(30) | NO   | UNI | NULL    |               |
+-----+-----+-----+-----+-----+-----+
```

```
mysql> INSERT INTO citas (fecha, hora, clienteId) VALUES
-> ('2021-06-28', '10:30:00', "HOLA");
ERROR 1366 (HY000): Incorrect integer value: 'HOLA' for column 'clienteId' at row 1
mysql> INSERT INTO citas (fecha, hora, clienteId) VALUES ('2021-06-28', '10:30:00', 2);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`appsalon`.`citas`, CONSTRAINT `cliente_FK` FOREIGN KEY (`clienteId`) REFERENCES `clientes` (`id`))
mysql> INSERT INTO citas (fecha, hora, clienteId) VALUES ('2021-06-28', '10:30:00', 1);
Query OK, 1 row affected (0.01 sec)
```

```
mysql>
```

```

mysql> SELECT * FROM clientes;
+----+-----+-----+-----+-----+
| id | nombre | apellido | telefono | email |
+----+-----+-----+-----+-----+
| 1  | Juan   | De la torre | 1839813910 | correo@correo.com |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM citas;
+----+-----+-----+-----+
| id | fecha      | hora      | clienteId |
+----+-----+-----+-----+
| 2  | 2021-06-28 | 10:30:00 |          1  |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM citas
    -> INNER JOIN clientes ON clientes.id = citas.clienteId;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | fecha      | hora      | clienteId | id | nombre | apellido | telefono | email |
+----+-----+-----+-----+-----+-----+-----+-----+
| 2  | 2021-06-28 | 10:30:00 |          1  | 1  | Juan   | De la torre | 1839813910 | correo@correo.com |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

mysql> SELECT * FROM citas LEFT JOIN clientes ON clientes.id = citas.clienteId;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | fecha      | hora      | clienteId | id | nombre | apellido | telefono | email |
+----+-----+-----+-----+-----+-----+-----+-----+
| 2  | 2021-06-28 | 10:30:00 |          1  | 1  | Juan   | De la torre | 1839813910 | correo@correo.com |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM citas RIGHT JOIN clientes ON clientes.id = citas.clienteId;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | fecha      | hora      | clienteId | id | nombre | apellido | telefono | email |
+----+-----+-----+-----+-----+-----+-----+-----+
| 2  | 2021-06-28 | 10:30:00 |          1  | 1  | Juan   | De la torre | 1839813910 | correo@correo.com |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO clientes (nombre, telefono, apellido, email) VALUES
    -> ("Karen", "Perez", "1093011391", "correo2@correo.com");

```

```

mysql> CREATE TABLE citasServicios (
    -> id INT(11) AUTO_INCREMENT,
    -> citaId INT(11) NOT NULL,
    -> servicioId INT(11) NOT NULL,
    -> PRIMARY KEY (id),
    -> KEY citaId (citaId),
    -> CONSTRAINT citas_FK
    -> FOREIGN KEY (citaId)
    -> REFERENCES citas (id),
    -> KEY servicioId (servicioId),
    -> CONSTRAINT servicios_FK
    -> FOREIGN KEY (servicioId)
    -> REFERENCES servicios (id)
    -> );

```

```

mysql> SELECT * FROM citas;
+----+-----+-----+-----+
| id | fecha | hora | clienteId |
+----+-----+-----+-----+
| 2 | 2021-06-28 | 10:30:00 | 1 |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM servicios;
+----+-----+-----+
| id | nombre | precio |
+----+-----+-----+
| 1 | Corte de Cabello Niño | 60.00 |
| 2 | Corte de Cabello Hombre | 80.00 |
| 3 | Corte de Barba | 60.00 |
| 4 | Peinado Mujer | 80.00 |
| 5 | Peinado Hombre | 60.00 |
| 6 | Tinte | 300.00 |
| 7 | Uñas | 400.00 |
| 8 | Lavado de Cabello | 50.00 |
| 9 | Tratamiento Capilar | 150.00 |
+----+-----+-----+
mysql> INSERT INTO citasServicios (citaId, servicioId) VALUES (2, 8);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM citasServicios;
+----+-----+-----+
| id | citaId | servicioId |
+----+-----+-----+
| 1 | 2 | 8 |
+----+-----+-----+
1 row in set (0.00 sec)

mysql> 
```

### Consultar tabla pivot

```

mysql> SELECT * FROM citasServicios
    -> LEFT JOIN citas ON citas.id = citasServicios.citaId
    -> LEFT JOIN servicios ON servicios.id = citasServicios.servicioId;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | citaId | servicioId | id | fecha | hora | clienteId | id | nombre | precio |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2 | 8 | 2 | 2021-06-28 | 10:30:00 | 1 | 8 | Lavado de Cabello | 50.00 |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO citasServicios (citaId, servicioId) VALUES (2, 3);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO citasServicios (citaId, servicioId) VALUES (2, 4);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM citasServicios;
+----+-----+-----+
| id | citaId | servicioId |
+----+-----+-----+
| 1 | 2 | 8 |
| 5 | 2 | 3 |
| 6 | 2 | 4 |
+----+-----+-----+
3 rows in set (0.00 sec)

mysql> 
```

```

mysql> SELECT * FROM citasServicios
    -> LEFT JOIN citas ON citas.id = citasServicios.citaId
    -> LEFT JOIN servicios ON servicios.id = citasServicios.servicioId;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | citaId | servicioId | id | fecha | hora | clienteId | id | nombre | precio |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2 | 8 | 2 | 2021-06-28 | 10:30:00 | 1 | 8 | Lavado de Cabello | 50.00 |
| 5 | 2 | 3 | 2 | 2021-06-28 | 10:30:00 | 1 | 3 | Corte de Barba | 60.00 |
| 6 | 2 | 4 | 2 | 2021-06-28 | 10:30:00 | 1 | 4 | Peinado Mujer | 80.00 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

```

```

mysql> SELECT * FROM citasServicios
    -> LEFT JOIN citas ON citas.id = citasServicios.citaId
    -> LEFT JOIN clientes ON citas.clienteId = clientes.id
    -> LEFT JOIN servicios ON servicios.id = citasServicios.servicioId;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | citaId | servicioId | id | fecha | hora | clienteId | id | nombre | apellido | telefono |
| email | id | nombre | precio | | | | | | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2 | 8 | 2 | 2021-06-28 | 10:30:00 | 1 | 1 | Juan | De la torre | 1839813910 |
| correo@correo.com | 8 | Lavado de Cabello | 50.00 | | | | | | |
| 5 | 2 | 3 | 2 | 2021-06-28 | 10:30:00 | 1 | 1 | Juan | De la torre | 1839813910 |
| correo@correo.com | 3 | Corte de Barba | 60.00 | | | | | | |
| 6 | 2 | 4 | 2 | 2021-06-28 | 10:30:00 | 1 | 1 | Juan | De la torre | 1839813910 |
| correo@correo.com | 4 | Peinado Mujer | 80.00 | | | | | | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

```

## Bases de Datos Relacionales (RDBMS)

- Microsoft SQL Server
- Oracle
- Microsoft Access
- MySQL
- MariaDB
- SAP Hana
- SQLite

## Bases de Datos de Escritorio

- Access
- Filemaker Pro

## Bases de Datos Empresariales

- SQL Server
- Oracle
- SAP Hana

**Cada BD se utiliza para una solución**

- MongoDB para transacciones bancarias y Big Data.
- MySQL y SQL Server para sitios y aplicaciones web.
- SQLite para aplicaciones móviles.