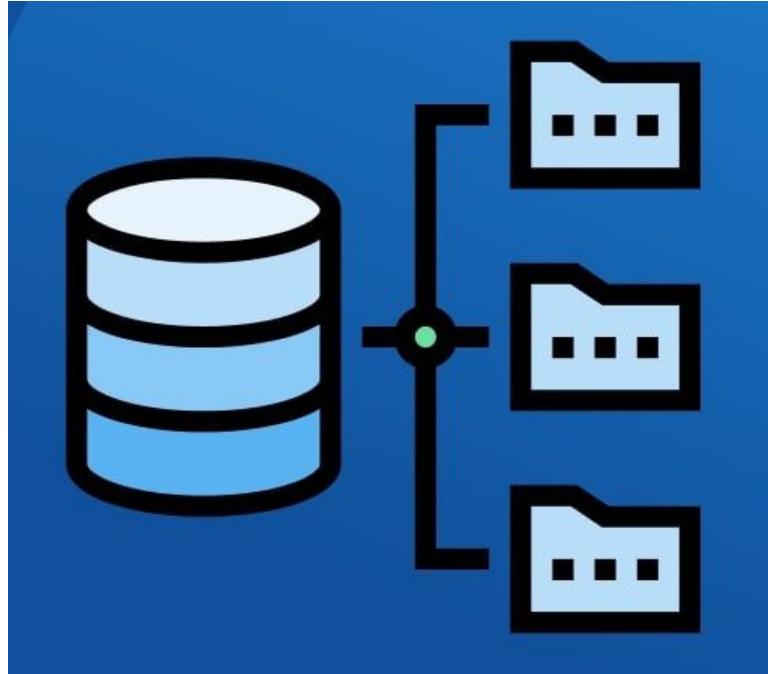


Normalización

¿Qué es la normalización?

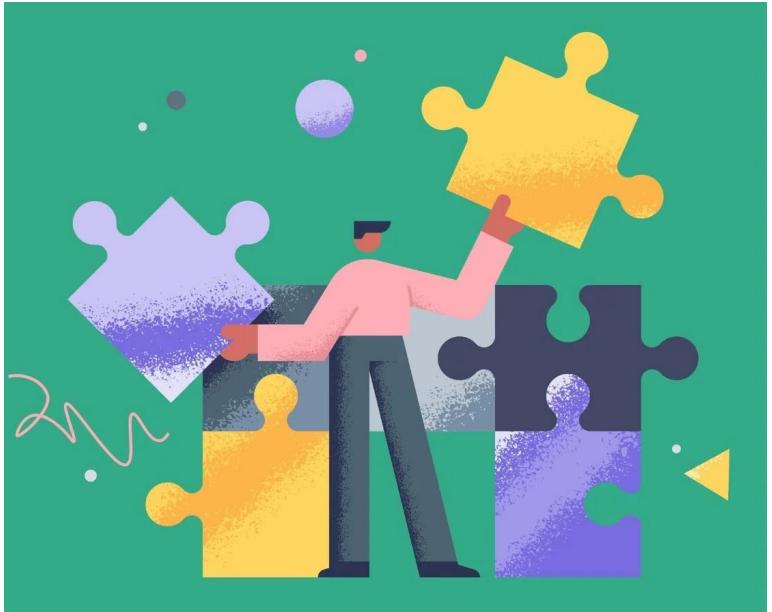
La normalización es la transformación de las vistas de usuario complejas y del almacén de datos a un juego de estructuras de datos más pequeñas y estables. Además de ser más simples y estables, las estructuras de datos son más fáciles de mantener que otras estructuras de datos.

(Kendall, 2005)



¿Para qué sirve la normalización?

Para mejorar el desempeño de una base de datos, así como evitar redundancia en la información que contiene y, en consecuencia, generar condiciones para un mejor diseño



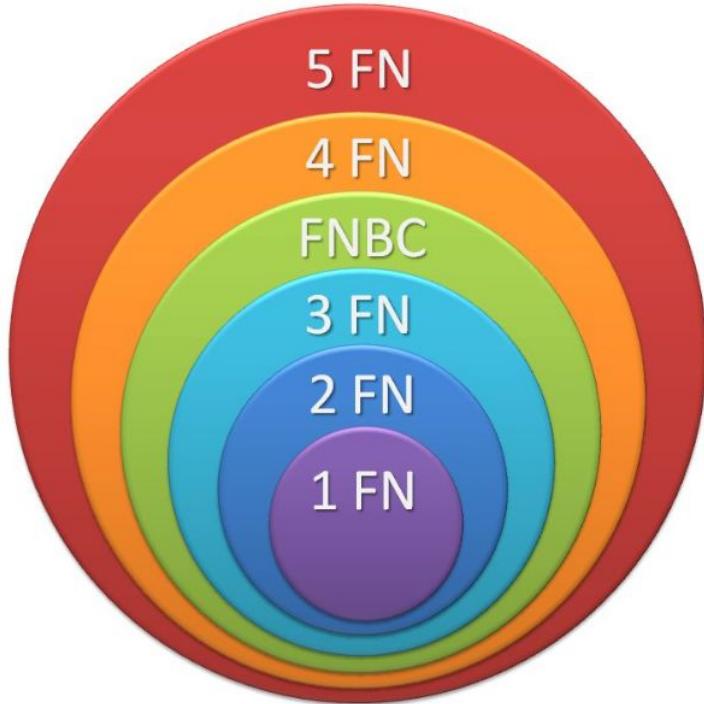
¿Cuándo se utiliza?

- Se crea una nueva estructura
- Para analizar las relaciones entre atributos
- Implementar mejoras en el diseño
- Modificaciones en una estructura



¿Cómo alcanzar la normalización?

La normalización de una base de datos persigue varios objetivos, principalmente reducir la redundancia de datos y simplificar las dependencias entre columnas, aplicándose de manera acumulativa. Lo anterior quiere decir que la segunda forma normal incluye a la primera, la tercera a la segunda y así sucesivamente.

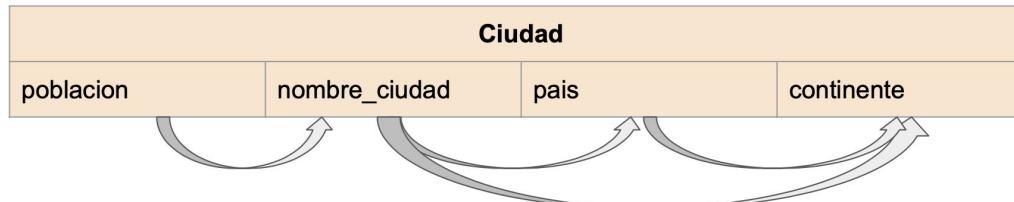
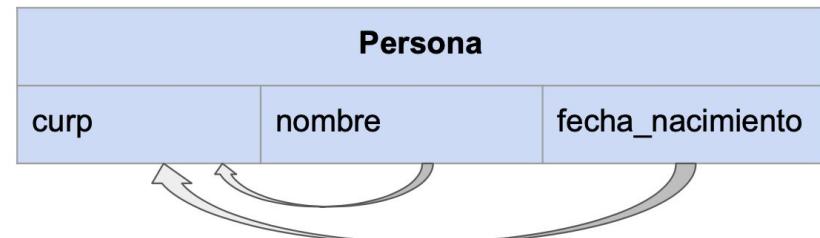


Transitividad

Dependencia
transitiva



Dependencia
funcional



Primera Forma Normal 1NF

Se deben seguir 2 indicaciones

1. Todos los atributos, valores almacenados en las columnas, deben ser indivisibles
2. No deben existir grupos de valores repetidos

Para esto la 1NF debe de:

- Tener todos los atributos llave definidos
- No hay grupos repetidos en la tabla (en cada intersección de fila y columna contiene sólo un valor, no varios)
- Todos los atributos dependen de la llave primaria

Ejemplo

Paso 1: Analizar la base y los grupos repetidores

Paciente	Raza	Año_de_nacimiento	Descripción	Juguete	Problema	Edad	Sexo	Comida
Pandora	Terrier	2009	Grande blanco	Carnaza	Jadeo, Dolor	12	Hembra	Campeón
Fry	Schnauzer	2016	Mediano gris	Pelota	Indigestión	5	Macho	Campeón
Honey	Schnauzer	2008	Mediano gris	Carnaza	Cansancio, Dolor	13	Hembra	Pedigree
Tana	Schnauzer	2016	Mediano gris	Revista	Lloriqueo	5	Hembra	Beneful
Nausy	Bulldog	2016	Grande blanco	Hueso	Indigestión	5	Macho	Campeón
Dory	Chihuahua	2016	Mini blanco	Cuerda	Cirugía, Indigestión	5	Hembra	Pedigree
Mila	Chihuahua	2010	Mini blanco	Pelota	Dolor	11	Macho	ProPlan
Bolillo	Husky	2020	Mediano gris	Hueso	Jadeo	1	Macho	ProPlan
Ragnar	Husky	2009	Mediano gris	Cuerda	Lloriqueo, Jadeo	12	Hembra	Campeón
Pandora	Terrier	2009	Grande blanco	Carnaza	Dolor, Cirugía	12	Hembra	Campeón
Mono	Salchicha	2011	Mediano café	Pelota	Agresividad	10	Macho	ProPlan
Nala	Salchicha	2012	Mediano Café	Hueso	Dolor	9	Hembra	Dog Chow
Dory	Chihuahua	2016	Mini blanco	Cuerda	Agresividad	5	Hembra	Pedigree

Ejemplo

Paso 2: Identificar columnas PK

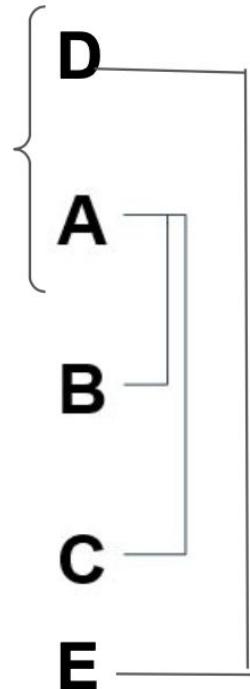
Paso 3: Identificar dependencias

Id	Nombre	Raza	Año_de_nacimiento	Color	Juguete	Tamaño	Problema	Edad	Sexo	Comida
1	Pandora	Terrier	2009	Blanco	Carnaza	Grande	Jadeo	12	Hembra	Campeón
2	Fry	Schnauzer	2016	Gris	Pelota	Mediano	Indigestión	5	Macho	Campeón
3	Honey	Schnauzer	2008	Gris	Carnaza	Mediano	Cansancio	13	Hembra	Pedigree
4	Honey	Schnauzer	2008	Gris	Carnaza	Mediano	Dolor	13	Hembra	Pedigree
5	Tana	Schnauzer	2016	Gris	Revista	Mediano	Lloriqueo	5	Hembra	Beneful
6	Nausy	Bulldog	2016	Blanco	Hueso	Grande	Indigestión	5	Macho	Campeón
7	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	Cirugía	5	Hembra	Pedigree
8	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	Indigestión	5	Hembra	Pedigree
9	Mila	Chihuahua	2010	Blanco	Pelota	Mini	Dolor	11	Macho	ProPlan
10	Bolillo	Husky	2020	Gris	Hueso	Mediano	Jadeo	1	Macho	ProPlan
11	Ragnar	Husky	2009	Gris	Cuerda	Grande	Lloriqueo	12	Hembra	Campeón
12	Ragnar	Husky	2009	Gris	Cuerda	Grande	Jadeo	12	Hembra	Campeón
13	Pandora	Terrier	2009	Blanco	Carnaza	Grande	Dolor	12	Hembra	Campeón
14	Pandora	Terrier	2009	Blanco	Carnaza	Grande	Cirugía	12	Hembra	Campeón
15	Mono	Salchicha	2011	Café	Pelota	Mediano	Agresividad	10	Macho	ProPlan
16	Nala	Salchicha	2012	Café	Hueso	Mediano	Dolor	9	Hembra	Dog Chow
17	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	Agresividad	5	Hembra	Pedigree

Segunda Forma Normal 2NF

- Se debe cumplir la 1NF
- Todos los atributos que no sean una llave primaria, deben depender de una
- No puede haber más de una llave primaria en la tabla
- La PK debe ser un solo atributo con valor indivisible

Llave
compuesta



Ejemplo

— — —

Id	Nombre	Raza	Año_de_nacimiento	Color	Juguete	Tamaño	Edad	Sexo	Comida
1	Pandora	Terrier	2009	Blanco	Carnaza	Grande	12	Hembra	Campeón
2	Fry	Schnauzer	2016	Gris	Pelota	Mediano	5	Macho	Campeón
3	Honey	Schnauzer	2008	Gris	Carnaza	Mediano	13	Hembra	Pedigree
4	Tana	Schnauzer	2016	Gris	Revista	Mediano	5	Hembra	Beneful
5	Nausy	Bulldog	2016	Blanco	Hueso	Grande	5	Macho	Campeón
6	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	5	Hembra	Pedigree
7	Mila	Chihuahua	2010	Blanco	Pelota	Mini	11	Macho	ProPlan
8	Bolillo	Husky	2020	Gris	Hueso	Mediano	1	Macho	ProPlan
9	Ragnar	Husky	2009	Gris	Cuerda	Grande	12	Hembra	Campeón
10	Pandora	Terrier	2009	Blanco	Carnaza	Grande	12	Hembra	Campeón
11	Mono	Salchicha	2011	Café	Pelota	Mediano	10	Macho	ProPlan
12	Nala	Salchicha	2012	Café	Hueso	Mediano	9	Hembra	Dog Chow
13	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	5	Hembra	Pedigree

Cita #	Nombre	Problema
1	Pandora	Jadeo
2	Fry	Indigestión
3	Honey	Cansancio
4	Honey	Dolor
5	Tana	Lloriqueo
6	Nausy	Indigestión
7	Dory	Cirugía
8	Dory	Indigestión
9	Mila	Dolor
10	Bolillo	Jadeo
11	Ragnar	Lloriqueo
12	Ragnar	Jadeo
13	Pandora	Dolor
14	Pandora	Cirugía
15	Mono	Agresividad
16	Nala	Dolor
17	Dory	Agresividad

Tercera Forma Normal 3NF

- No deben existir **dependencias transitivas** entre las columnas de una tabla
 - *Las columnas que no forman parte de la clave primaria deben depender sólo de la clave*
- Para eliminar las dependencias se tienen que crear nuevas tablas, y reasignar los atributos

Ejemplo: Nombre y color

Id	Nombre	Raza	Año_de_nacimiento	Color	Juguete	Tamaño	Edad	Sexo	Comida
1	Pandora	Terrier	2009	Blanco	Carnaza	Grande	12	Hembra	Campeón
2	Fry	Schnauzer	2016	Gris	Pelota	Mediano	5	Macho	Campeón
3	Honey	Schnauzer	2008	Gris	Carnaza	Mediano	13	Hembra	Pedigree
4	Tana	Schnauzer	2016	Gris	Revista	Mediano	5	Hembra	Beneful
5	Nausy	Bulldog	2016	Blanco	Hueso	Grande	5	Macho	Campeón
6	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	5	Hembra	Pedigree
7	Mila	Chihuahua	2010	Blanco	Pelota	Mini	11	Macho	ProPlan
8	Bolillo	Husky	2020	Gris	Hueso	Mediano	1	Macho	ProPlan
9	Ragnar	Husky	2009	Gris	Cuerda	Grande	12	Hembra	Campeón
10	Pandora	Terrier	2009	Blanco	Carnaza	Grande	12	Hembra	Campeón
11	Mono	Salchicha	2011	Café	Pelota	Mediano	10	Macho	ProPlan
12	Nala	Salchicha	2012	Café	Hueso	Mediano	9	Hembra	Dog Chow
13	Dory	Chihuahua	2016	Blanco	Cuerda	Mini	5	Hembra	Pedigree

Id_paciente	Nombre	Id_raza	Año_de_nacimiento	Id_juguete	Edad	Id_sexo	Id_comida
1	Pandora	5	2009	1	12	2	5
2	Dory	3	2016	3	5	2	1
3	Honey	1	2008	1	13	2	1
4	Mono	6	2011	5	8	1	3
5	Nausy	2	2016	4	5	1	5
6	Ragnar	4	2009	3	12	1	5
7	Mila	3	2010	5	11	2	3
8	Bolillo	4	2020	4	1	1	3
9	Tana	1	2016	2	5	2	2
10	Fry	1	2016	5	5	1	5
11	Nala	6	2012	4	9	2	4

Id_comida	Comida	Id_Raza	Raza	Color	Tamaño
1	Pedigree	1	Schnauzer	Gris	Mediano
2	Beneful	2	Bulldog	Blanco	Grande
3	ProPlan	3	Chihuahua	Blanco	Mini
4	Dog Chow	4	Husky	Gris	Mediano
5	Campeón	5	Terrier	Blanco	Grande
		6	Salchicha	Café	Mediano

Id_sexo	Sexo
1	Macho
2	Hembra

Id_juguete	Juguete
1	Carnaza
2	Revista
3	Cuerda
4	Hueso
5	Pelota

Forma final

Id_problema	Problema
1	Jadeo
2	Dolor
3	Cirugía
4	Lloriqueo
5	Agresividad
6	Indigestión
7	Cansancio

Cita #	Id_paciente	Id_Problema
1	1	1
2	10	6
3	3	7
4	3	2
5	9	4
6	5	6
7	2	3
8	2	6
9	7	2
10	8	1
11	6	4
12	6	1
13	1	2
14	1	3
15	4	5
16	11	2
17	2	5

Aprendizaje colaborativo: Checklist de Normalización

Checklist

Actividad 5. Una escuela (a)normal

Han sido seleccionados para realizar su servicio social en la poderosísima prepa 6.

Nos encargaron entregarles una base de datos con bastantes registros de alumnos y profesores, y nos encargaron que ustedes la normalizaran hasta 3NF.

https://docs.google.com/spreadsheets/d/1G_h6vVuaRBBHmBhoEhjz5wx02t0FdIjttebdZIw0AE/edit?usp=sharing

En equipos de 3 analicen la base que se les entregarán. Para evaluar la actividad, tendrán que crear una nueva hoja de cálculo donde se aprecien las tablas creadas al normalizar y los registros correspondientes a cada una.

Sólo un integrante del equipo entregará el link del commit con el archivo en pdf dentro de la carpeta de la actividad. No olviden incluir el nombre de los integrantes en el archivo.

¿Qué es un SGBD?

- Conjunto de programas que permiten el
 - Almacenamiento
 - Modificación
 - Extracción de
- información en una base de datos



Funciones de un SGBD

- Control de concurrencia
- Cumplir reglas de integridad y redundancias
- Realizar copias de seguridad y recuperación de datos
- Restricción de acceso no autorizados
- Suministrar múltiples interfaces de usuario
- Representar relaciones complejas entre datos



¿Qué es SQL?

Es el lenguaje estandarizado que se usa para operar una base de datos relacional



¿Por qué MySQL?

Es un manejador de bases de datos muy popular y que

- Es gratis
- Es extensible
- Es robusto
- Es rápido
- No requiere de muchos recursos
- Es fácil de administrar



Tipos de dato

Numéricos

Nombre	Ejemplo
BOOL o BOOLEAN	1
BIT	63
TINYINT	255
SMALLINT	65535
MEDIUMINT	-8388608
INT o INTEGER	2146472648
BIGINT	-9223372036854775807
DECIMAL	999.99
FLOAT	-3.402823466 × 1038

Texto

Nombre	Ejemplo
CHAR	TengoEspacios
VARCHAR	NoTengoEspacios
TINYBLOB	[Soy un BLOB]
TEXT	Soy un texto largo

Fecha

Nombre	Ejemplo
TIME	12:59:59
DATE	2021-05-04
DATETIME	2021-05-04 11:00:03
YEAR	2021

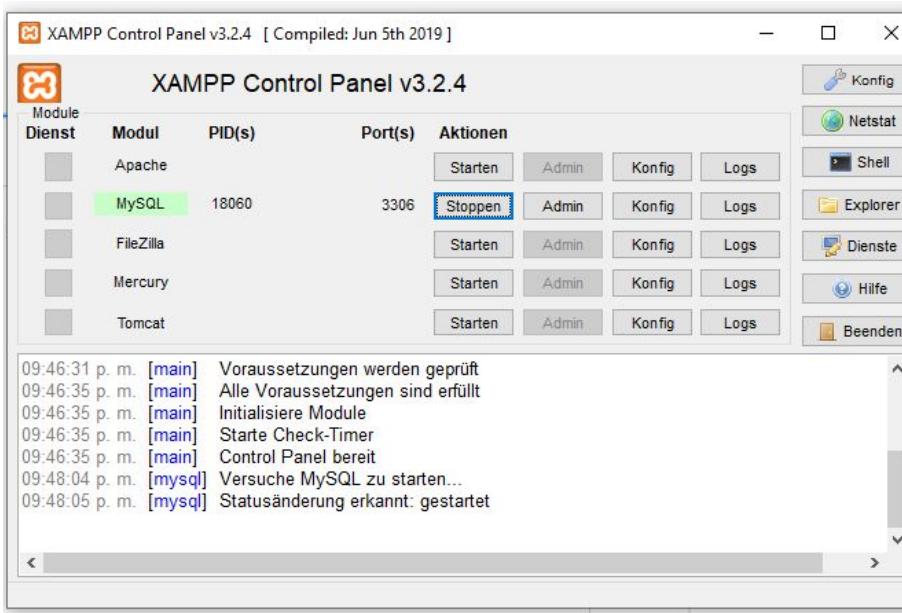
Actividad 4.2 Diccionario de datos

Como te fue tan bien en la parte anterior, tu amigue te pidió que también le hicieras el diccionario de datos y como quieres ir a su casa para ver a sus gatitos le vas a hacer el favor. Entrega el diccionario de datos junto con los diagramas E-R.



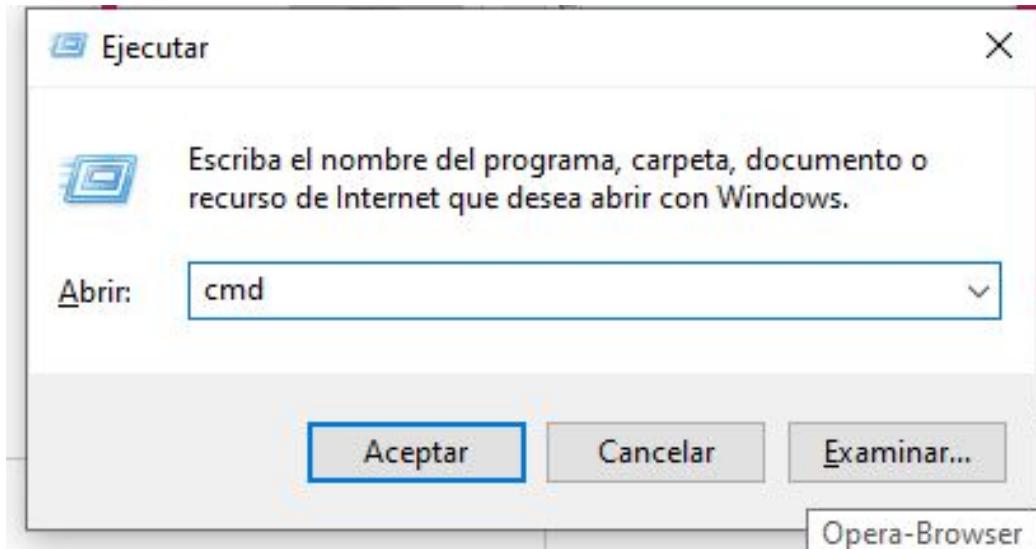
Implementación de bases de datos

Cómo iniciar tu servidor sql con XAMPP



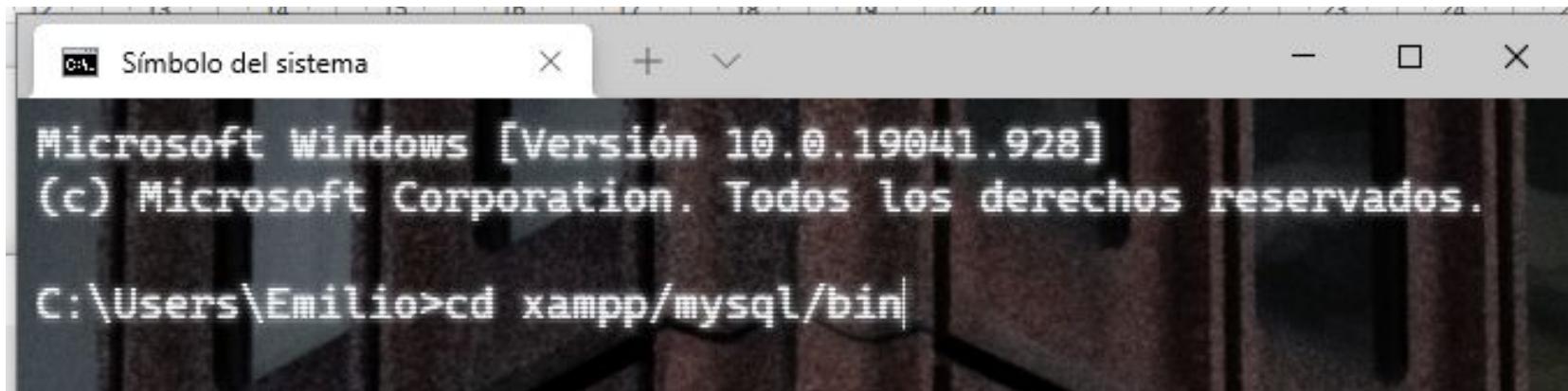
1. Abre tu consola de XAMPP e inicia el servidor MySQL

Cómo iniciar tu servidor sql con XAMPP



2. Abre tu consola con WINDOWS+R y escribiendo 'cmd'.

Cómo iniciar tu servidor sql con XAMPP



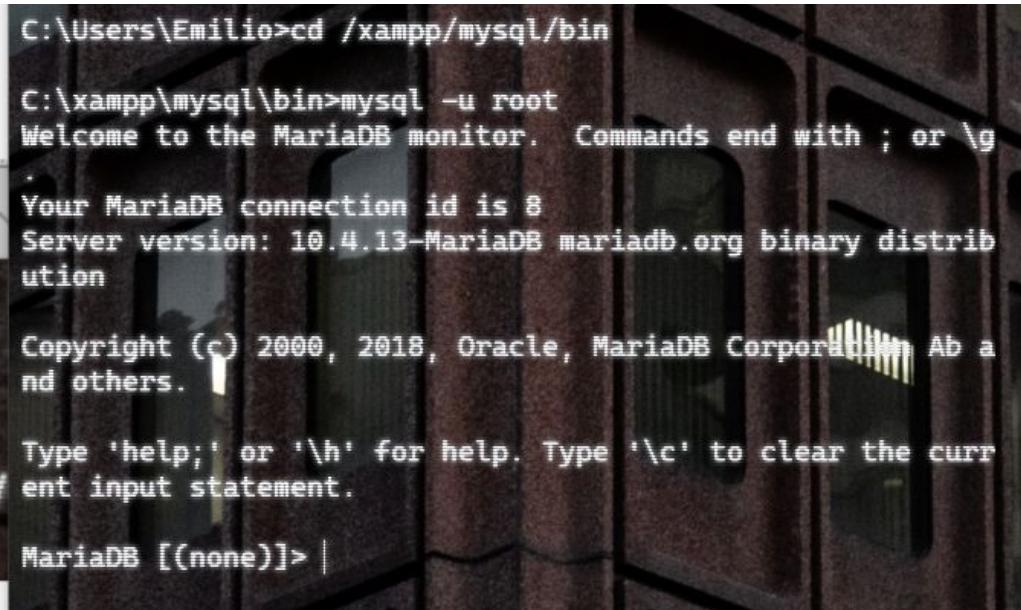
Símbolo del sistema

Microsoft Windows [Versión 10.0.19041.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Emilio>cd xampp/mysql/bin|

3. Entra a la carpeta de mysql: Si estas en C:, puedes hacerlo con el comando ‘cd xampp/mysql/bin’

Cómo iniciar tu servidor sql con XAMPP



```
C:\Users\Emilio>cd /xampp/mysql/bin

C:\xampp\mysql\bin>mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g
.
Your MariaDB connection id is 8
Server version: 10.4.13-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> |
```

4. Dentro de bin escribe el comando ‘mysql -u root’ para iniciar el servidor MariaDB

Clasificación de las sentencias SQL: DML y DDL

DDL (Lenguaje de definición de datos):

Las sentencias DDL **se utilizan para crear y modificar la estructura de las tablas así como otros objetos** de la base de datos.



Ver y acceder a bases de datos

SHOW DATABASES;

USE <nombre de la base de datos>;

Elementos utilizados por DDL

- **CREATE:** para crear objetos en la base de datos.
- **ALTER:** modifica la estructura de la base de datos.
- **DROP:** borra objetos de la base de datos.
- **TRUNCATE:** elimina todos los registros de la tabla, incluyendo todos los espacios asignados a los registros.

Creación de una base de datos (db):

CREATE DATABASE <nombre de la base de datos> **CHARACTER SET** utf8mb4 **COLLATE** utf8mb4_general_ci

CONSTRAINTS (Restricciones)

Las CONSTRAINTS son restricciones que se utilizan para limitar el tipo de dato que puede recibir una columna de una tabla.

- **NOT NULL:** Se asegura de que una columna no pueda tener un valor NULL.
- **UNIQUE:** Se asegura de que todos los valores en una columna sean diferentes.
- **PRIMARY KEY:** Una combinación entre NOT NULL y UNIQUE. Identifica de forma única cada registro(tupla).
- **FOREIGN KEY:** Identifica de forma única los renglones o registros de una tabla enlazada a otra con llave primaria.

CONSTRAINTS (Restricciones)

- **CHECK:** Se asegura de que todos los valores en una columna cumplan una condición específica.
- **DEFAULT:** Da un valor por defecto cuando no especifica alguno.
- **AUTO_INCREMENT:** Genera automáticamente un número único cuando se inserta un nuevo registro en una tabla (incremento unitario).

NOT NULL y UNIQUE

```
CREATE TABLE <nombreTabla>(  
    <col1> <tipo>() NOT NULL,  
    <col2> <tipo>());
```

```
CREATE TABLE <nombreTabla>(  
    <col1> <tipo>() UNIQUE,  
    <col2> <tipo>(),  
    <col3> <tipo>(),  
    <col4> <tipo>(),  
    UNIQUE (<col2>),  
    CONSTRAINT <nombreEspecial> UNIQUE (<col3>, <col4>));
```

CONSTRAINT nos ayuda a referencias a un grupo de atributos con una misma restricción.

CHECK && DEFAULT && AUTO_INCREMENT

```
CREATE TABLE <nombreTabla>(  
    <col1> <tipo>(),  
    <col2> <tipo>,  
    CHECK (<col2> *=20));
```

```
CREATE TABLE <nombreTabla>(  
    <col1> <tipo>() DEFAULT “<valor>”,  
    <col2> <tipo>());
```

```
CREATE TABLE <nombreTabla>(  
    <col1> <tipo>() AUTO_INCREMENT,  
    <col2> <tipo>());
```

RENAME && ADD && DROP (Modificación de Tablas)

ALTER TABLE <nombreTabla> **RENAME** <nuevoNombreTabla>;

ALTER TABLE <nombreTabla> **ADD** <col> <tipo>(<longitud>);

ALTER TABLE <nombreTabla> **DROP** <col>;

CHANGE && MODIFY

Sustituir atributos

ALTER TABLE <nombreTabla> **CHANGE** <colActual> <colNueva> <tipo>(<longitud>);

Cambiar tipo de dato y restringir que sea null

ALTER TABLE <nombreTabla> **MODIFY** <col> <nuevoTipo>(nueva longitud);

ALTER TABLE <nombreTabla> **MODIFY** <col> <tipo>(long) **NOT NULL**;

PRIMARY KEY

```
ALTER TABLE <nombreTabla> ADD PRIMARY KEY (<col>);
```

```
ALTER TABLE <nombreTabla> DROP PRIMARY KEY;
```

FOREIGN KEY

```
ALTER TABLE <nombreTabla1> ADD FOREIGN KEY (<col>) REFERENCES  
<nombreTabla2>(<col>);
```

```
ALTER TABLE <nombreTabla1> ADD CONSTRAINT <nombreEspecial> FOREIGN KEY  
(<col>) REFERENCES <nombreTabla2>(<col>);
```

```
ALTER TABLE <nombreTabla1> DROP FOREIGN KEY <nombreEspecial>;
```

UNIQUE

```
ALTER TABLE <nombreTabla> ADD UNIQUE (<col>);
```

```
ALTER TABLE <nombreTabla> ADD CONSTRAINT <indice> UNIQUE (<col1>, ...);
```

```
ALTER TABLE <nombreTabla> DROP INDEX <indice>;
```

CHECK && DEFAULT

ALTER TABLE <nombreTabla> **ADD CONSTRAINT** <indice> **CHECK** (<atributo> <= *valor*);

ALTER TABLE <nombreTabla> **DROP CHECK** <indice>;

ALTER TABLE <nombreTabla> **ALTER** <col> **SET DEFAULT** <valor>;

ALTER TABLE <nombreTabla> **ALTER COLUMN** <col> **DROP DEFAULT**;

Ver mis tablas

SHOW TABLES;

DESCRIBE <nombre de la tabla>;

¿Cómo eliminar una base de datos (DB)?

```
DROP TABLE <nombreTabla>;
```

```
DROP DATABASE <nombreBD>;
```



¡Hagamos un ejemplo!

Actividad 6.1 Mi primera base de datos

Recién egresado del Curso Web, en Facebook te piden crear una base de datos con tu apellido y nombre (PavonLeninInd) para que cada usuario pueda expresar su individualidad. Te ordenan hacerla de la siguiente forma:

- Vas a crear 4 entidades: música, películas, videojuegos y libros
- En cada entidad debe de poderse saber su nombre, creador, y año (atributos obligatorios)
 - En música se debe también guardar el álbum
 - El álbum predeterminado es The Wall
 - En películas se debe guardar el actor principal y su clasificación (AA, A, B, B15, C)
 - La clasificación predeterminada es AA
 - En videojuegos se debe guardar el protagonista
 - El protagonista predeterminado es Sans
 - En libros debe guardarse editorial y lugar
- Todos los datos no obligatorios pueden ser nulos

DML (Lenguaje de manipulación de datos):

Las sentencias DML son aquellas **utilizadas para insertar, borrar, modificar y consultar los datos** de una base de datos.

DML da **órdenes que afectan al contenido** de los archivos de datos.



Elementos utilizados por DML

Los elementos que se utilizan para manipular los datos, son los siguientes:

- **INSERT:** instrucción que permite insertar los valores deseados en una base de datos.
- **SELECT:** sentencia para realizar consultas sobre los datos.
- **UPDATE:** modifica los valores de uno o varios registros, es decir, actualiza la base.
- **DELETE:** se utiliza para eliminar los registros de una tabla.

Inserción de datos

— — —

```
INSERT INTO <nombrertabla> (col1 , col2, col3, etc) VALUES (“VARCHAR”, INT,  
“TEXT”, etc.)
```

```
INSERT INTO Alumno (Nombre, Edad, Descripción) VALUES (“Juan Christian”, 17,  
“Es ... ”)
```

```
INSERT INTO Alumno VALUES (“Juan Christian”, 17, “Es ... ”)
```

Consulta de registros

SELECT * FROM Alumno

SELECT * FROM Alumno WHERE Nombre=“Enrique”

Consulta de registros

SELECT <col1>, <col2>, <col3>, etc FROM <nombretabla>

SELECT Nombre, Edad, Descripción FROM Alumno

**SELECT Nombre, Edad, Descripción FROM Alumno WHERE
Nombre=“Enrique”**

¡Hagamos otro
ejemplo!

Cláusula WHERE

Es la instrucción que **nos permite filtrar el resultado de una sentencia de consulta, actualización y eliminación de datos** de una tabla.

Estructura:

...**WHERE** condiciones;

...**WHERE** col = “valor”;

...**WHERE** col = “valor” AND col (<,>, <>, !=) número;

...**WHERE** (col =“valor” AND col = “valor”) OR (col > 5000);

Actualización de registros

**UPDATE <nombreretabla> SET col=“valor” WHERE
col=“valor”**

UPDATE Alumno SET Edad=“18” WHERE Nombre=“Luis”

Eliminación de registros

Un registro o varios:

DELETE FROM <nombretabla> WHERE col=“valor”

DELETE FROM Alumno WHERE Edad=“15”

Eliminación de registros

Eliminar todos los registros de una tabla:

DELETE FROM <nombrertabla>;

TRUNCATE <nombrertabla>;

Actividad 6.2 Mi primera base de datos

- Con la base de datos que creaste debes insertar al menos 5 registros a cada tabla y tomarles captura de pantalla.
- Selecciona todos los valores de música, y en otra consulta selecciona todos los valores de libros
- Selecciona el atributo actor principal de todos los registros con clasificación predeterminada
- Selecciona el año de los videojuegos donde el protagonista no sea el predeterminado
- Como ahora Tom Hanks está en todas las películas que han existido, actualiza el actor principal de todas las películas a Tom Hanks
- Falsa alarma, Tom Hanks no existe (como Tlaxcala), elimina todos los registros que hayan existido de él de tu base de datos

Consultas simples

DISTINCT

SELECT DISTINCT <atributo> FROM <tabla>

SELECT DISTINCT nombre FROM alumnos

Comparaciones

Igualdad

=

Mayor que

>

Menor que

<

Mayor o igual que

\geq

Menor o igual que

\leq

Diferente

$(\neq) \text{ || } (<>)$

DISTINCT

```
mysql> SELECT id_pais FROM atletas;
+-----+
| id_pais |
+-----+
|      1 |
|      1 |
|      2 |
|      3 |
|      3 |
|      3 |
|      3 |
|      3 |
|      5 |
|      6 |
|      6 |
|      6 |
|      7 |
|      8 |
|      8 |
|      9 |
|      9 |
+-----+
```

```
mysql> SELECT DISTINCT id_pais FROM atletas;
+-----+
| id_pais |
+-----+
|      1 |
|      2 |
|      3 |
|      5 |
|      6 |
|      7 |
|      8 |
|      9 |
+-----+
8 rows in set (0.03 sec)
```

Igual (=)

```
mysql> SELECT 1 = 0;
+-----+
| 1 = 0 |
+-----+
|      0 |
+-----+
1 row in set (0.01 sec)

mysql> SELECT '0' = 0;
+-----+
| '0' = 0 |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT '0.0' = 0;
+-----+
| '0.0' = 0 |
+-----+
|      1 |
+-----+
```

```
mysql> SELECT '0.01' = 0;
+-----+
| '0.01' = 0 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT '0.01' = 0.01;
+-----+
| '0.01' = 0.01 |
+-----+
|          1 |
+-----+
1 row in set (0.01 sec)
```

Mayor que (>)

```
mysql> SELECT 2 > 2;
+-----+
| 2 > 2 |
+-----+
|      0 |
+-----+
```

Menor que (<)

```
mysql> SELECT 2 < 2;
+-----+
| 2 < 2 |
+-----+
|      0 |
+-----+
```

Menor o igual que (\leq)

```
mysql> SELECT 2 <= 2;
+-----+
| 2 <= 2 |
+-----+
|      1 |
+-----+
```

Mayor o igual que (\geq)

```
mysql> SELECT 2 >= 2;
+-----+
| 2 >= 2 |
+-----+
|      1 |
+-----+
```

Diferente (\neq) || (\neq)

```
mysql> SELECT '.01' <> '0.01';
+-----+
| '.01' <> '0.01' |
+-----+
|      1 |
+-----+
1 row in set (0.01 sec)

mysql> SELECT .01 <> '0.01';
+-----+
| .01 <> '0.01' |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT 'zapp' <> 'zappp';
+-----+
| 'zapp' <> 'zappp' |
+-----+
|      1 |
+-----+
1 row in set (0.00 sec)
```

BETWEEN

```
[mysql] > SELECT nombre, id_atleta FROM atletas WHERE id_atleta BETWEEN 3 AND 5;  
+-----+-----+  
| nombre | id_atleta |  
+-----+-----+  
| Bjørn | 3 |  
| Christopher | 4 |  
| Edoardo | 5 |  
+-----+-----+  
3 rows in set (0.02 sec)
```

NOT BETWEEN

```
mysql> SELECT nombre, id_atleta FROM atletas WHERE id_atleta NOT BETWEEN 3 AND 5;
+-----+-----+
| nombre | id_atleta |
+-----+-----+
| Amy    |      1 |
| Birgit |      2 |
| Eric   |      6 |
| Jennifer |    7 |
| Krishna |     8 |
| Kristin |     9 |
| Larisa |    10 |
| Marit  |    11 |
| Matthew |   12 |
| Nedo   |   13 |
| Nikolái |  14 |
| Ole    |  15 |
| Ryan   |  17 |
| Sawao  |  18 |
| Vitali |  20 |
+-----+-----+
15 rows in set (0.01 sec)
```

IN

```
[mysql]> SELECT nombre, id_atleta FROM atletas WHERE id_atleta IN(3,5,7,8);
+-----+-----+
| nombre | id_atleta |
+-----+-----+
| Bjørn | 3 |
| Edoardo | 5 |
| Jennifer | 7 |
| Krishna | 8 |
+-----+-----+
4 rows in set (0.01 sec)
```

NOT IN

```
[mysql]> SELECT nombre, id_atleta FROM atletas WHERE id_atleta NOT IN(3,5,7,8);
+-----+-----+
| nombre | id_atleta |
+-----+-----+
| Amy    |      1 |
| Birgit |      2 |
| Christopher | 4 |
| Eric   |      6 |
| Kristin |      9 |
| Larisa |     10 |
| Marit  |     11 |
| Matthew |     12 |
| Nedo   |     13 |
| Nikolái |     14 |
| Ole    |     15 |
| Ryan   |     17 |
| Sawao  |     18 |
| Vitali |     20 |
+-----+-----+
14 rows in set (0.00 sec)
```

Wildard (comodín)

'%' - Cero, uno, o múltiples valores

'_' - un valor

Ejemplos uso de Comodín	Descripción
'%a'	Cualquier valor que termina con 'a'
'%a%'	Cualquier valor que tenga 'a' en cualquier posición
'_a%'	Cualquier valor que tiene 'a' en su segunda posición
'a_%'	Cualquier valor que comienza con 'a' y tiene por lo menos 2 caracteres

LIKE

```
mysql> SELECT nombre,apellidoPat,apellidoMat FROM atletas WHERE nombre LIKE 'Amy';
+-----+-----+-----+
| nombre | apellidoPat | apellidoMat |
+-----+-----+-----+
| Amy    | Van        | Dyken      |
+-----+-----+-----+
1 row in set (0.01 sec)

mysql> SELECT nombre,apellidoPat,apellidoMat FROM atletas WHERE nombre LIKE 'A';
Empty set (0.00 sec)
```

NOT LIKE

```
mysql> SELECT nombre, apellidoPat, apellidoMat FROM atletas WHERE nombre NOT LIKE 'B%';
```

nombre	apellidoPat	apellidoMat
Amy	Van	Dyken
Christopher	Hoy	NULL
Edoardo	Mangiarotti	NULL
Eric	Heiden	NULL
Jennifer	Thompson	NULL
Krishna	Ramayana	Smith
Kristin	Otto	NULL
Larisa	Semiónovna	Latýnina
Marit	Bjørgen	NULL
Matthew	Biondi	NULL
Nedo	Nadi	NULL
Nikolái	Yefímovich	Andriánov
Ole	Einar	Bjørndalen
Ryan	Steven	Lochte
Sawao	Kato	NULL
Vitali	Shcherbo	NULL

```
16 rows in set (0.02 sec)
```

Comparación con valores nulos

```
mysql> SELECT nombre, apellidoPat, apellidoMat FROM atletas WHERE apellidoMat IS NULL;
+-----+-----+-----+
| nombre | apellidoPat | apellidoMat |
+-----+-----+-----+
| Christopher | Hoy | NULL |
| Edoardo | Mangiarotti | NULL |
| Eric | Heiden | NULL |
| Jennifer | Thompson | NULL |
| Kristin | Otto | NULL |
| Marit | Bjørgen | NULL |
| Matthew | Biondi | NULL |
| Nedo | Nadi | NULL |
| Sawao | Kato | NULL |
| Vitali | Shcherbo | NULL |
+-----+-----+-----+
```

Comparación con valores nulos

```
mysql> SELECT nombre, apellidoPat, apellidoMat FROM atletas WHERE apellidoMat IS NOT NULL;
+-----+-----+-----+
| nombre | apellidoPat | apellidoMat |
+-----+-----+-----+
| Amy    | Van        | Dyken      |
| Birgit | Fischer    | Schmidt    |
| Bjørn  | Erlend     | Dæhlie    |
| Krishna | Ramayana   | Smith      |
| Larisa | Semiónovna | Latýnina  |
| Nikolái | Yefímovich | Andriánov |
| Ole    | Einar       | Bjørndalen|
| Ryan   | Steven      | Lochte    |
+-----+-----+-----+
8 rows in set (0.00 sec)
```

Funciones de agrupamiento

```
MariaDB [prueba]> SELECT * FROM numero;
+-----+
| numero |
+-----+
|      1 |
|      3 |
|      7 |
|     10 |
|     12 |
|      4 |
|      8 |
+-----+
7 rows in set (0.000 sec)
```

AVG

SELECT AVG(col) FROM <tabla>;

Devuelve el promedio de un grupo de valores.

```
MariaDB [prueba]> SELECT AVG(numero) FROM numero;
+-----+
| AVG(numero) |
+-----+
|      6.4286 |
+-----+
1 row in set (0.000 sec)
```

COUNT

SELECT COUNT(col) FROM <tabla>;

Devuelve el número de registros con respecto a los valores de un atributo.

```
MariaDB [prueba]> SELECT COUNT(numero) FROM numero;
+-----+
| COUNT(numero) |
+-----+
|          7   |
+-----+
1 row in set (0.000 sec)
```

```
MariaDB [prueba]> SELECT * FROM instructor;
```

id_mentor	nombre	edad
0	Jaasiel	18
1	Carlos	17
2	Lenin	17
3	Melo	18
4	Luis	18
5	Abraham	18
6	Christian	17
7	Emiliano	17
8	KIKI	66

```
9 rows in set (0.000 sec)
```

MIN

SELECT MIN(col) FROM <tabla>;

Devuelve el o los registros con los valores más pequeños.

```
MariaDB [prueba]> SELECT MIN(edad) FROM instructor;
+-----+
| MIN(edad) |
+-----+
|      17   |
+-----+
1 row in set (0.010 sec)
```

MAX

SELECT MAX(col) FROM <tabla>;

Devuelve el o los registros con los valores más grandes.

```
MariaDB [prueba]> SELECT MAX(edad) FROM instructor;
+-----+
| MAX(edad) |
+-----+
|      66   |
+-----+
1 row in set (0.000 sec)
```

SUM

SELECT SUM(COL) FROM <tabla>;

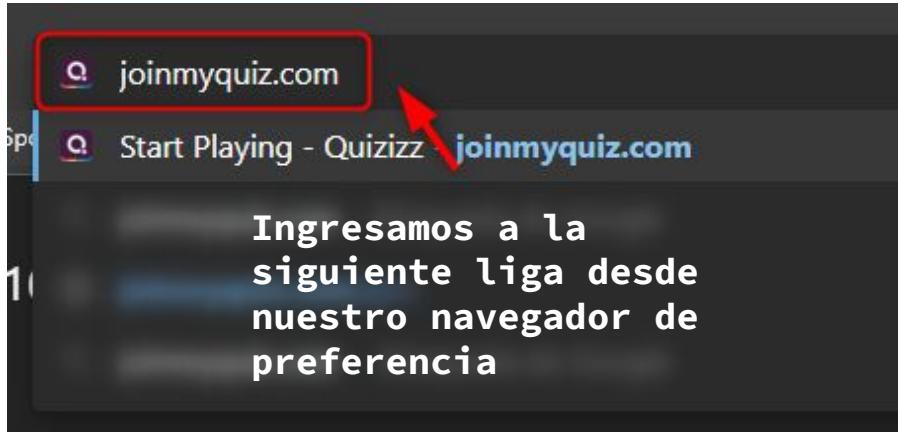
Devuelve el resultado de la suma de los registros de una columna.

```
MariaDB [prueba]> SELECT SUM(edad) FROM instructor;
+-----+
| SUM(edad) |
+-----+
|      206 |
+-----+
1 row in set (0.010 sec)
```

QUIZIZZ



Ahora un pequeño desafío
por equipos :D



Breve tutorial previo al desafío

A screenshot of the Quizizz website. The address bar at the top shows the URL "https://quizizz.com/join". Below the address bar, the page content includes the Quizizz logo, a search bar, and navigation links for "Inicio" and "Actividad". On the left, there is text: "En este apartado se introducirá el código correspondiente a la sala de juego :D". On the right, there is text: "La liga anterior nos redireccionará al enlace que aparece en pantalla". At the bottom, there is a large input field with the placeholder "Código de sala" and a purple button labeled "ÚNETE". A red arrow points from the text "joinmyquiz.com" in the slide's instructions up to the URL in the browser's address bar. Another red arrow points from the text "Código de sala" in the slide's instructions down to the input field on the website.

En este apartado se introducirá el código correspondiente a la sala de juego :D

La liga anterior nos redireccionará al enlace que aparece en pantalla

Código de sala

ÚNETE

Algunas notas importantes y reglas de juego:

1. Deberás ingresar al juego con un nombre que sea fácilmente identificable por los mentores (**Por ejemplo: Cristian Umaña**)
2. La actividad se llevará a cabo por medio de los grupos reducidos, por lo que es de vital importancia que colabores con tu equipo para conseguir la mayor cantidad de puntos
3. Un mentor será asignado a cada una de las salas para supervisar la actividad, mas no participará en esta (**NO responderán dudas que interfieran con tu puntuación de juego**)
4. Los primeros 2 equipos ganadores serán acreedores a 2 y 1 décimas, respectivamente, sobre su calificación final en el proyecto del módulo 1 y 2

Creación de Usuarios

Ver usuarios

```
USE mysql;  
SHOW TABLES;  
SELECT User FROM user;
```

CREATE USER

```
CREATE USER '<usr>'@'localhost' IDENTIFIED BY  
'<contraseña>';
```

```
CREATE USER 'xochitl'@'localhost' IDENTIFIED BY  
'CW2021';
```

*no tiene permisos para nada

¿Y mi usuario?

```
mysql -u <usr> -p
```

```
mysql -u xochitl -p
```

Tipos de permisos

Tipos de permisos

ALL PRIVILEGES: permite a un usuario acceder a todas las bases de datos asignadas en el sistema

CREATE: permite crear nuevas tablas o bases de datos

DROP: permite eliminar tablas o bases de datos

DELETE: permite eliminar registros de tablas

INSERT: permite insertar registros en tablas

SELECT: permite leer registros en tablas

UPDATE: permite actualizar registros seleccionados en tablas

GRANT OPTION: permite remover privilegios de los usuarios

Proporcionar permisos

```
GRANT ALL PRIVILEGES ON *.* TO  
'<usr>'@'localhost';
```

```
GRANT <permiso> ON <base(s) de datos>.<tabla(s)>  
TO '<usr>'@'localhost';
```

Los asteriscos son un comodín que indica que los permisos serán asignados a todas las bases de datos y a todas las tablas, en ese orden

Proporcionar permisos

También podemos asignar varios permisos a la vez como en el siguiente ejemplo

```
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON
Base.* TO '<usr>'@'localhost';
```

Revocar permisos

```
REVOKE <permiso> ON <base(s) de datos>.<tabla(s)>
    FROM '<usr>'@'localhost';
```

Eliminar usuarios

```
DROP USER '<usr>'@'localhost';
```

Actividad 6.3

Para ya terminar, deben crear un usuario personalizado llamado ‘axolotl’ de nombre como su base que tenga todos los privilegios en su base y en la tabla música, pero que no pueda borrar en películas ni en videojuegos, y que tampoco sea capaz de agregar ni mucho menos modificar en la entidad libros.

Además del respaldo de tu base añadir capturas de pantalla describiendo los pasos para crear al usuario.

Respaldos

Creación de respaldos

```
mysqldump -u '<usr>' <nombre de la base de datos>  
          > respaldo.sql;
```

Creación de respaldos

```
C:\xampp\mysql\bin>mysqldump -u root enp6 > enp62.sql  
C:\xampp\mysql\bin>
```

Este equipo > Windows (C:) > xampp > mysql > bin				
Nombre	Fecha de modifica...	Tipo	Tamaño	
aria_chk	14/06/2019 08:19 ...	Aplicación	3,918 KB	
aria_dump_log	14/06/2019 08:19 ...	Aplicación	3,486 KB	
aria_ftdump	14/06/2019 08:19 ...	Aplicación	3,712 KB	
aria_pack	14/06/2019 08:19 ...	Aplicación	3,758 KB	
aria_read_log	14/06/2019 08:19 ...	Aplicación	3,895 KB	
enp62	12/05/2020 08:25 a...	Archivo SQL	1,139 KB	

Macintosh HD > Aplicaciones > MAMP > Library > bin > enp6resp.sql

Restauración de respaldos (si está en la carpeta bin)

- En CMD

```
mysql -u '<usr>' <nombre de la base de datos> <  
respaldo.sql;
```

- En MySQL

```
source respaldo.sql;
```