

Ejercicios prácticos

# **MySQL: Introducción a bases de datos relacionales**

---

## Actividades

1.

Mirá atentamente la estructura de esta tabla que contiene información de mensajes de texto.

Mensaje	Hora	Destinatario
Feliz cumpleaños	15:21	Paula
Buenos días	08:20	Francisco
Bienvenido a Puerto Rico	17:30	Alejandra
¿A qué hora nos encontramos?	14:00	Marcos

Responde las siguientes preguntas:

- ¿A qué hora se envió el mensaje “Feliz cumpleaños”? ¿A quién se le envió ese mensaje?
- ¿A quién se le envió un mensaje a las 14:00? ¿Qué contenido tiene ese mensaje?
- ¿Qué cambio harías en la tabla para agregar la fecha del mensaje?
- ¿Qué cambios harías en la tabla para agregar un mensaje nuevo?

2.

Una veterinaria quiere almacenar la información de algunos perros. De cada uno de ellos se quiere almacenar su nombre, su raza y su dueño.

Hasta ahora se tiene información de los siguientes perros:

- Rolly, un Labrador. Su dueño se llama Lucas.
- Tommy, un Beagle. Su dueña se llama Ana.
- Piky, un Caniche. Su dueña se llama Natalia.

Pensá cómo sería la estructura de la tabla que permita persistir la información de estos perros y cuál sería su contenido.

3.

Esta tabla almacena información de estudiantes de una universidad. Tiene la siguiente estructura:

Legajo	Dirección	Fecha de nacimiento	Mail	Número de Documento
-	-	-	-	-
-	-	-	-	-

¿Cuáles columnas podrían ser claves primarias de esta tabla? ¿Son claves naturales o subrogadas?

4.

Nos pasaron el siguiente diagrama entidad-relación que almacena información de los clientes de un negocio. La **primary key** de esta tabla es la fecha de nacimiento de los clientes. ¿Qué problemas puede traer esa clave primaria? ¿Cómo los solucionarías?

Cliente
nombre apellido fecha_nacimiento (PK) documento telefono

5.

El siguiente diagrama entidad-relación representa la estructura de una tabla que almacena la información de funciones de cine. Identificá el tipo de dato de cada atributo de esta tabla.

Función
id (PK) película sala fecha_y_hora precio cantidad_entradas

- **id:** debe ser un número entero.
- **película:** debe ser una cadena que pueda tener hasta 70 caracteres.
- **sala:** el número de sala debe ser un número entero.
- **fecha\_y\_hora:** en este campo se va a almacenar el día y horario de la función.
- **precio:** el precio puede tener hasta 3 dígitos enteros y hasta 2 dígitos decimales. Se debe poder representar, por ejemplo, 125,75 o 92,50.
- **cantidad\_entradas:** un número entero que representa la cantidad de entradas que se pueden vender.

6.

Una cancha de fútbol va a comenzar a almacenar la información de los partidos que se van a jugar. Para esto, va a utilizar una base de datos. La información que se quiere almacenar es la siguiente:

- Un identificador para cada partido que debe ser un número entero.
- La fecha y hora en la que se va a jugar el partido.
- El nombre del equipo 1. La cadena puede tener hasta 50 caracteres.
- El nombre del equipo 2. La cadena puede tener hasta 50 caracteres.
- El precio de la entrada del partido. Puede tener hasta 4 dígitos enteros y hasta 2 dígitos decimales. Se debe poder representar, por ejemplo, 1450,50 o 375,45.

Realiza un diagrama entidad-relación donde representes la entidad que va a almacenar esta información. Especificá los atributos de esta entidad, su tipo de dato y la clave primaria.

7.

Escribí el comando para crear una base de datos. Llamala reportes.

8.

Escribí el comando para crear la siguiente tabla:

alumno
id INT (PK) nombre VARCHAR (30) apellido VARCHAR (30) fecha_nacimiento DATE

- Todos los campos de esta tabla son obligatorios, no pueden ser nulos.
- El campo id debe ser auto incremental.

9.

Escribí el comando para crear la siguiente tabla:

producto
id INT (PK) descripcion VARCHAR (50) precio_mayorista DECIMAL(8,2) precio_minorista DECIMAL(8,2) stock INT

- El campo id debe ser auto incremental.
- Los campos id, descripcion, stock y precio\_minorista son campos obligatorios, es decir, no pueden ser nulos.
- El campo precio\_mayorista no es obligatorio.

10.

En la consola de MySQL, realizá los siguientes pasos:

- a) Creá una base de datos. Llamala cine.
- b) Ingresá a esa base de datos.
- c) Dentro de la base de datos cine, creá la siguiente tabla:

pelicula
id INT (PK) titulo VARCHAR (70) duracion INT director VARCHAR(70)

- Todos los campos de esta tabla son obligatorios, no pueden ser nulos.
- El campo id debe ser auto incremental.

11.

- a) Completá la siguiente consulta para que obtenga el campo nombre de la tabla estudiante:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_;
- b) Completá la siguiente consulta para que obtenga los campos nombre y dueño de la tabla mascota:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_;
- c) Completá la siguiente consulta par que obtenga los campos nombre, poblacion y continente de la tabla pais:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_;
- d) Completá la siguiente consulta para que obtenga todos los campos de la tabla frutas:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_;

12.

Este es el modelo de una tabla que almacena la información de los corredores de una carrera.

corredor
id INT (PK) kilometros INT nombre VARCHAR(70) apellido VARCHAR(70) edad INT

Realizá las siguientes consultas:

- a) Obtené todos los datos de los corredores.
- b) Obtené el nombre de todos los corredores.
- c) Obtené el nombre y apellido de todos los corredores.
- d) Obtené el nombre de cada corredor y la cantidad de kilómetros que va a correr.
- e) Obtené el nombre, apellido y edad de cada corredor.

13.

Este es el modelo de una tabla que almacena la información de exámenes de una escuela.

examen
id INT (PK) nota INT alumno VARCHAR(70) materia VARCHAR(70) fecha DATE

Realizá las siguientes consultas:

- a) Obtené todos los datos de los exámenes.
- b) Obtené la nota de todos los exámenes.
- c) Obtené la materia de todos los exámenes.
- d) Obtené el alumno y la nota de cada examen.
- e) Obtené la nota y la materia de cada examen.

- f) Obtené la materia y la fecha de cada exámen.
- g) Obtené la materia, fecha y nota de cada exámen.
- h) Obtené el nombre del alumno, la nota, la materia y la fecha de cada exámen.

14.

Creá un script de SQL que realice las siguientes acciones:

- a) Crear una base de datos llamada clases.
- b) Ingresar a la base de datos creada.
- c) Crear una tabla llamada clase con la siguiente estructura:

clase
id INT <b>(PK)</b> cantidad_alumnos INT profesor VARCHAR(70) materia VARCHAR(70)

- Todos los campos de esta tabla son obligatorios, no pueden ser nulos.
- El campo id debe ser auto incremental.

Una vez que crees el script, guardalo como script.sql. Ingresá a la consola de MySQL y ejecutalo.

15.

Este es el modelo de una tabla que almacena la información sobre los trabajadores de una empresa multinacional.

empleado
documento INT <b>(PK)</b> apellido VARCHAR(70) nombre VARCHAR(70) area VARCHAR(40) pais VARCHAR(50) fecha_ingreso DATE edad INT

Realizá las siguientes consultas:



- a) Obtené todos los datos de los empleados cuya área es Marketing.
- b) Obtené el área y el apellido de los empleados cuya área es Finanzas.
- c) Obtené todos los datos de los empleados cuyo país es Alemania.
- d) Obtené el documento, nombre y país de los empleados cuyo país es Chile.
- e) Obtené el documento y la edad de los empleados que tienen 20 años.
- f) Obtené nombre, apellido y fecha de ingreso de los empleados que ingresaron después del 2008.
- g) Obtené nombre, apellido y edad de los empleados que tienen más de 30 años.
- h) Obtené nombre, apellido y edad de los empleados que tienen menos de 60 años.

16.

- a) Completá la siguiente consulta para que obtenga los campos nombre, apellido y de edad de una tabla llamada estudiante, de todos aquellos que tengan más de 20 años:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_ WHERE \_\_\_\_\_ ;
- b) Completá la siguiente consulta para que obtenga todos los campos de la tabla factura, de todas aquellas que tienen un precio total mayor a 1000:  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_ WHERE \_\_\_\_\_ ;
- c) Completá la siguiente consulta para que obtenga el campo descripción de la tabla producto, de todos aquellos que tengan un precio igual a 300.  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_ WHERE \_\_\_\_\_ ;
- d) Completá la siguiente consulta para que obtenga todos los campos de una tabla llamada ciudad, de aquellas ciudades que tienen una población mayor a 10.000.000.  
SELECT \_\_\_\_\_ FROM \_\_\_\_\_ WHERE \_\_\_\_\_ ;

17.

En esta tabla se almacena información de los socios de un gimnasio.

socio
numero_socio INT (PK) nombre_completo VARCHAR(100) actividad VARCHAR(70) fecha_ingreso DATE valor_cuota INT edad INT

Realizá las siguientes consultas:

- a) Obtené el número de socio, la fecha de ingreso y el valor de la cuota de los socios que ingresaron al club antes del 2005 y que pagan una cuota mayor a 500.
- b) Obtené la actividad y la edad de los socios que realizan “baile” y tienen más de 50 años.
- c) Obtené el numero y nombre de los socios que realizan la actividad “gimnasia artística” y pagan una cuota menor a 700.
- d) Obtené el nombre completo y la actividad que realizan los socios que tienen menos de 15 años e ingresaron al club antes del 2012.
- e) Obtené el nombre completo, la edad y el valor de la cuota de los socios que realizan “boxeo”, tienen más de 28 años y pagan una cuota menor a 540.
- f) Obtené la fecha de ingreso, el número y el nombre completo de los socios que tienen más de 30 años, ingresaron después del 2008 y pagan más de 1000.
- g) Obtené el nombre completo de los socios que realizan actividad “tenis”, ingresaron luego del 2014 y pagan una cuota menor a 800.

18.

En esta tabla se almacena la información de los juegos de un parque de diversiones.

juego
id INT (PK) nombre VARCHAR(100) altura_minima DECIMAL(4,2) cantidad_personas INT minutos_espera INT habilitado BIT

**Aclaración:** el campo “disponible” es de tipo BIT. Esto quiere decir que si el juego está habilitado, el campo va a tomar valor 1. Si en cambio, el juego no se encuentra habilitado, va a tomar el valor 0.

Realiza las consultas que correspondan para responder las siguientes preguntas:

- ¿Hay algún juego cuya altura mínima sea 1,5 y la cantidad de personas que pueden ingresar sea mayor o igual a 70?
- ¿Hay algún juego habilitado en el que se pueda ingresar esperando 15 minutos o menos?
- ¿Hay algún juego en el que pueda entrar un niño que mide 1,25 esperando 30 minutos o menos?
- ¿Hay algún juego habilitado al que puedan ingresar 10 personas a la vez?
- ¿Hay algún juego en el que puedan entrar a la vez 15 niños que miden 1,5 en 30 minutos o menos?
- ¿Hay algún juego habilitado al que puedan ingresar más de 15 personas a la vez en 10 minutos o menos?
- ¿Hay algún juego habilitado al que pueda ingresar un niño de 1,3 de altura en 20 minutos o menos?

## Soluciones

1.

- a) El mensaje se envió a Paula a las 15:21.
- b) El mensaje se le envió a Marcos. El contenido del mensaje es “¿A qué hora nos encontramos?”.
- c) Para agregar la fecha del mensaje, se debería agregar una nueva columna en la tabla, cuyo nombre podría ser **Fecha**.
- d) Para agregar un nuevo mensaje, se debería agregar una nueva fila en la tabla con la información del mensaje.

2.

Las columnas de esta tabla serían nombre, raza y dueño. En cada fila de esta tabla, se va a almacenar la información de cada uno de los perros.

Nombre	Raza	Dueño
Rolly	Labrador	Lucas
Tommy	Beagle	Ana
Piky	Caniche	Natalia

3.

- Legajo: cada estudiante tiene su propio legajo, no van a existir en la misma universidad, dos alumnos con el mismo. Por lo tanto, este campo puede ser clave primaria de la tabla.
- Dirección: pensá en dos hermanos que asisten a la misma universidad y viven en la misma casa. Son distintas personas que tienen la misma dirección. Este campo podría repetirse en más de un estudiante, por lo tanto, no puede ser clave primaria de la tabla.
- Fecha de nacimiento: dos estudiantes pueden haber nacido el mismo día. Este campo no puede ser clave primaria de la tabla.

- Mail: los correos electrónicos son propios de cada persona, no pueden existir dos dirección de mail iguales. Este campo puede ser clave primaria de la tabla.
- Número de Documento: cada persona tiene el propio, no existen dos personas con el mismo número de documento. Este campo puede ser clave primaria de la tabla.

Estos tipos de claves son naturales, porque forman parte de la tabla, no fueron agregadas para identificar a cada estudiante.

#### 4.

En el caso de que exista más de un cliente con la misma fecha de nacimiento puede haber problemas. Imaginemos por ejemplo que queremos cambiar el número de teléfono de un cliente que nació en una determinada fecha. Vamos a buscar en la tabla a la persona con esa fecha de nacimiento y actualizar su teléfono. ¿Y si había 10 clientes con la misma fecha de nacimiento? Estaríamos cambiando el teléfono de todos ellos, lo que nos llevaría a tener datos inconsistentes.

Para solucionar este problema, debemos seleccionar como clave primaria un atributo de la tabla que identifique unívocamente a cada cliente. En este caso podemos utilizar el número de documento, ya que no existen dos personas que tengan el mismo. Otra opción sería agregar a la tabla un nuevo campo que podríamos llamar **id**. Este campo sería autoincremental, es decir, la base de datos se encargaría de generarlo para cada cliente.

#### 5.

- **id:** INT
- **pelicula:** VARCHAR(70)
- **sala:** INT
- **fecha\_y\_hora:** DATETIME
- **precio:** DECIMAL(5,2)
- **cantidad\_entradas:** INT

6.

Partido
id INT (PK) fecha_hora DATETIME equipo_1 VARCHAR(50) equipo_2 VARCHAR(50) precio DECIMAL(6,2)

7.

CREATE DATABASE reportes;

8.

```
CREATE TABLE alumno(  
    id INT NOT NULL AUTO_INCREMENT,  
    nombre VARCHAR(30) NOT NULL,  
    apellido VARCHAR(30) NOT NULL,  
    fecha_nacimiento DATE NOT NULL,  
    PRIMARY KEY (id)  
);
```

9.

```
CREATE TABLE producto(  
    id INT NOT NULL AUTO_INCREMENT,  
    descripcion VARCHAR(50) NOT NULL,  
    precio_mayorista DECIMAL(8,2),  
    precio_minorista DECIMAL(8,2) NOT NULL,  
    stock INT NOT NULL,  
    PRIMARY KEY (id)  
);
```

10.

- a) CREATE DATABASE cine;
- b) USE cine;
- c) CREATE TABLE pelicula(  
    id INT NOT NULL AUTO\_INCREMENT,  
    titulo VARCHAR(70) NOT NULL,  
    duracion INT NOT NULL,  
    director VARCHAR(70) NOT NULL,  
    PRIMARY KEY(id)  
);

11.

- a) SELECT nombre FROM estudiante;
- b) SELECT nombre, dueño FROM mascota;
- c) SELECT nombre, poblacion, continente FROM pais;
- d) SELECT \* FROM frutas;

12.

- a) SELECT \* FROM corredor;
- b) SELECT nombre FROM corredor;
- c) SELECT nombre, apellido FROM corredor;
- d) SELECT nombre, kilometros FROM corredor;
- e) SELECT nombre, apellido, edad FROM corredor;

13.

- a) SELECT \* FROM examen;
- b) SELECT nota FROM examen;
- c) SELECT materia FROM examen;
- d) SELECT alumno, nota FROM examen;
- e) SELECT nota, materia FROM examen;
- f) SELECT materia, fecha FROM examen;
- g) SELECT materia, fecha, nota FROM examen;
- h) SELECT alumno, nota, materia, fecha FROM examen;

14.

En tu archivo script.sql deberías tener el siguiente código:

```
CREATE DATABASE clases;
USE clases;
CREATE TABLE clase(
    id INT NOT NULL AUTO_INCREMENT,
    cantidad_alumnos INT NOT NULL,
    profesor VARCHAR(70) NOT NULL,
    materia VARCHAR(70) NOT NULL,
    PRIMARY KEY(id)
);
```

15.

- a) SELECT \* FROM empleado WHERE area = 'Marketing';
- b) SELECT area, apellido FROM empleado WHERE area = 'Finanzas';
- c) SELECT \* FROM empleado WHERE pais = 'Alemania';
- d) SELECT documento, nombre, pais FROM empleado WHERE pais = 'Chile';
- e) SELECT documento, edad FROM empleado WHERE edad = 20;
- f) SELECT nombre, apellido, fecha\_ingreso FROM empleado WHERE fecha\_ingreso > '2008-01-01'
- g) SELECT nombre, apellido, edad FROM empleado WHERE edad > 30;
- h) SELECT documento, edad FROM empleado WHERE edad < 60;

16.

- a) SELECT nombre, apellido, edad FROM estudiante WHERE edad > 20;
- b) SELECT \* FROM factura WHERE precio\_total > 1000;
- c) SELECT descripcion FROM producto WHERE precio = 300;
- d) SELECT \* FROM ciudad WHERE poblacion > 10.000.000;

17.

- a) SELECT numero\_socio, fecha\_ingreso, valor\_cuota FROM socio WHERE fecha\_ingreso < '2005-01-01' AND valor\_cuota > 500;
- b) SELECT edad, actividad FROM socio WHERE actividad = 'baile' AND edad > 50;



- c) `SELECT numero_socio, nombre_completo FROM socio WHERE actividad = 'gimnasia artistica' AND cuota < 700;`
- d) `SELECT nombre_completo, actividad FROM socio WHERE edad < 15 AND fecha_ingreso < '2012-01-01';`
- e) `SELECT nombre_completo, edad, cuota FROM socio WHERE actividad = 'boxeo' AND edad > 28 AND cuota < 540.`
- f) `SELECT fecha_ingreso, numero_socio, nombre_completo FROM socio WHERE edad > 30 AND fecha_ingreso > '2008-01-01' AND cuota > 1000.`
- g) `SELECT nombre_completo FROM socio WHERE actividad = 'tenis' AND fecha_ingreso > '2014-01-01' AND cuota < 800;`

18.

- a) `SELECT * FROM juego WHERE altura_minima = 1,5 AND cantidad_personas >= 70;`
- b) `SELECT * FROM juego WHERE habilitado = 1 AND minutos_espera <= 15;`
- c) `SELECT * FROM juego WHERE altura_minima <= 1,25 AND minutos_espera <= 30;`
- d) `SELECT * FROM juego WHERE habilitado = 1 AND cantidad_personas >= 10`
- e) `SELECT * FROM juego WHERE cantidad_personas >= 15 AND altura_minima <= 1,5 AND minutos_espera <= 30;`
- f) `SELECT * FROM juego WHERE habilitado = 1 AND cantidad_personas >= 15 and minutos_espera <= 10;`
- g) `SELECT * FROM juego WHERE habilitado = 1 AND altura_minima <= 1,3 AND minutos_espera <= 20;`