

CDMX, 07 de Abril de 2025.

*Actividad 4. Ejercicios*

*Base de datos relacionales.*

Miranda Velazquez Mariles  
Alfonso Roberto Gómez Rosales  
Jose Emiliano Jauregui Guzmán



## ACTIVIDAD IV:

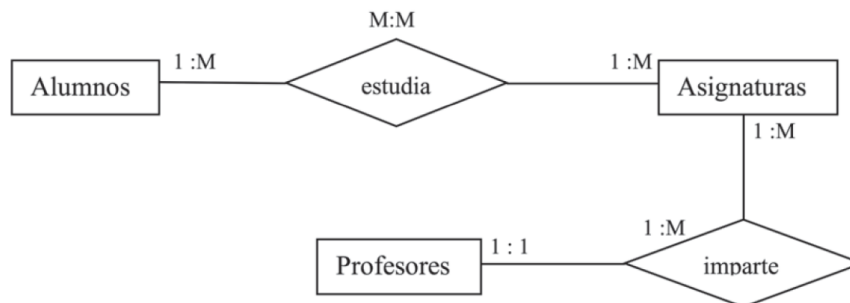
### EJERCICIO

1. Con base en el material consultado en la unidad resuelve el ejercicio que se plantea a continuación acerca de los siguientes temas:

- Lenguaje de Definición de Datos DDL
- Modelo Entidad-Relación
- MySQL Workbench

#### Ejercicio. Generación de estructura de base de datos mediante DDL

2. Analiza con detenimiento el siguiente diagrama donde se representan diversas entidades y la relación entre ellas.



Fuente: López, M., Gallegos, F., (2017). *Programación de bases de datos relacionales*.

3. Utilizando tu ambiente de trabajo MySQL Workbench, el cual instalaste en la primera parte del proyecto integrador:

a) Crea una base de datos Universidad y en ella genera las siguientes tablas mediante código SQL, pon especial atención en las relaciones mostradas en el diagrama anterior, para que establezcas tus llaves primarias y foráneas correctamente:

- Alumnos: Número de control, Nombre, Apellido, Edad
- Asignaturas: ID, Nombre, Número de créditos, Profesor, área
- estudia: Debe registrar las materias asociadas a los alumnos.



### Base de datos MySQL

```
-- Crear la base de datos
CREATE DATABASE Universidad;
USE Universidad;

-- Alumnos
CREATE TABLE Alumnos (
    num_control INT PRIMARY KEY,
    nombre VARCHAR(50),
    apellido VARCHAR(50),
    edad INT
);

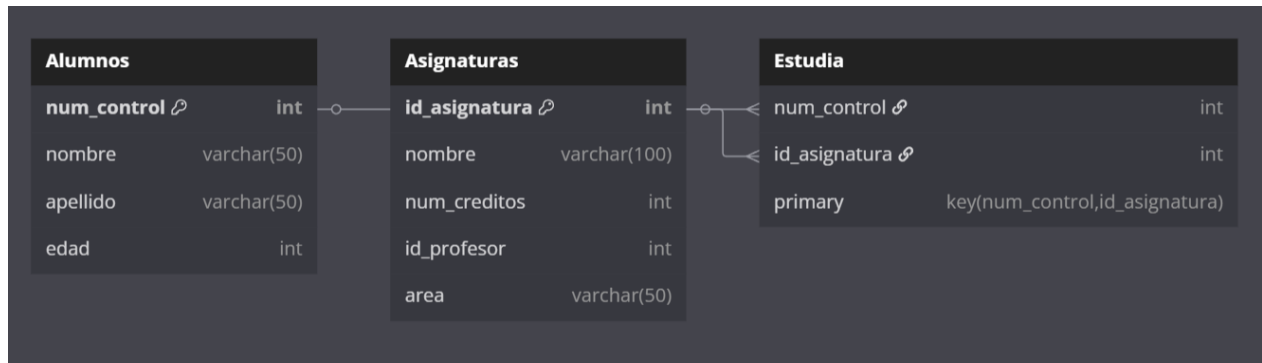
-- Profesores
CREATE TABLE Profesores (
    id_profesor INT PRIMARY KEY,
    nombre VARCHAR(50),
    apellido VARCHAR(50)
);

-- Asignaturas
CREATE TABLE Asignaturas (
    id_asignatura INT PRIMARY KEY,
    nombre VARCHAR(100),
    num_creditos INT,
    id_profesor INT,
    area VARCHAR(50),
    FOREIGN KEY (id_profesor) REFERENCES Profesores(id_profesor) ON DELETE SET
    NULL
);

CREATE TABLE estudia (
    num_control INT,
    id_asignatura INT,
    PRIMARY KEY (num_control, id_asignatura),
    FOREIGN KEY (num_control) REFERENCES Alumnos(num_control) ON DELETE
    CASCADE,
    FOREIGN KEY (id_asignatura) REFERENCES Asignaturas(id_asignatura) ON DELETE
    CASCADE
);
```



- b) Define los campos (atributos) de cada una de las tablas (entidades) y el tipo de dato correspondiente, para representarlos genera un diagrama Entidad-Relación en Lucidchart. Únicamente con las tres entidades solicitadas.

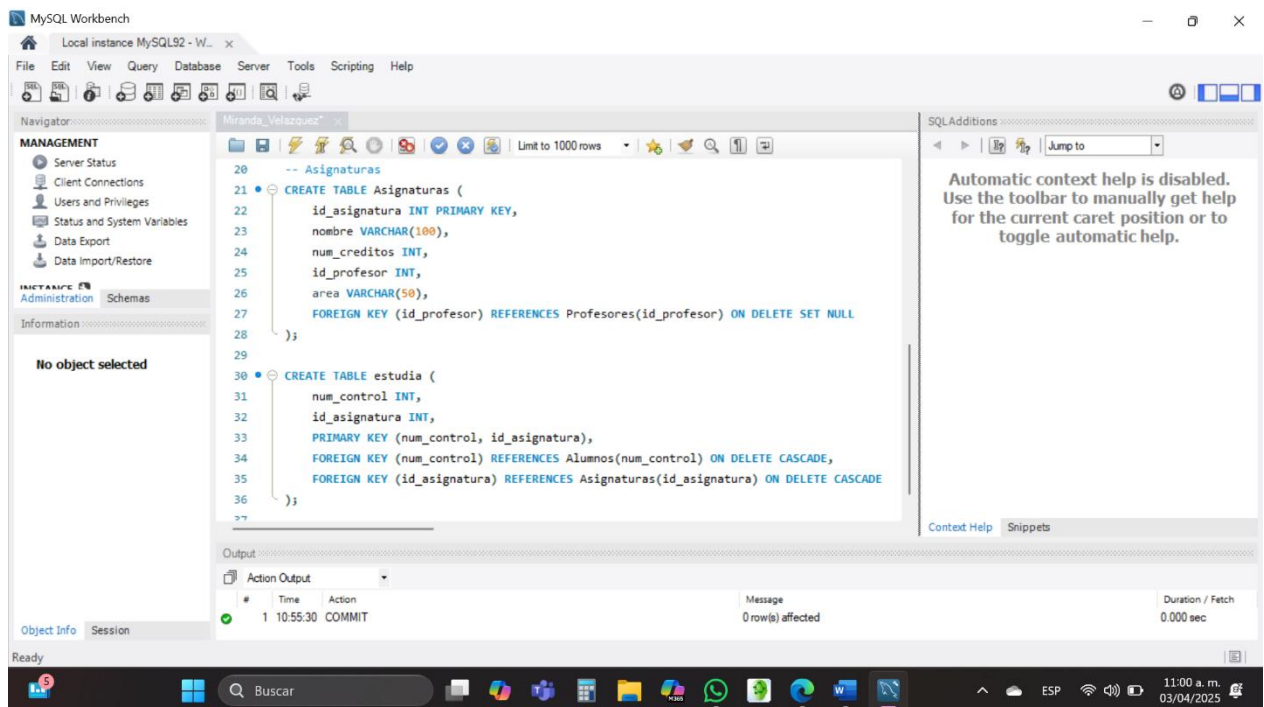
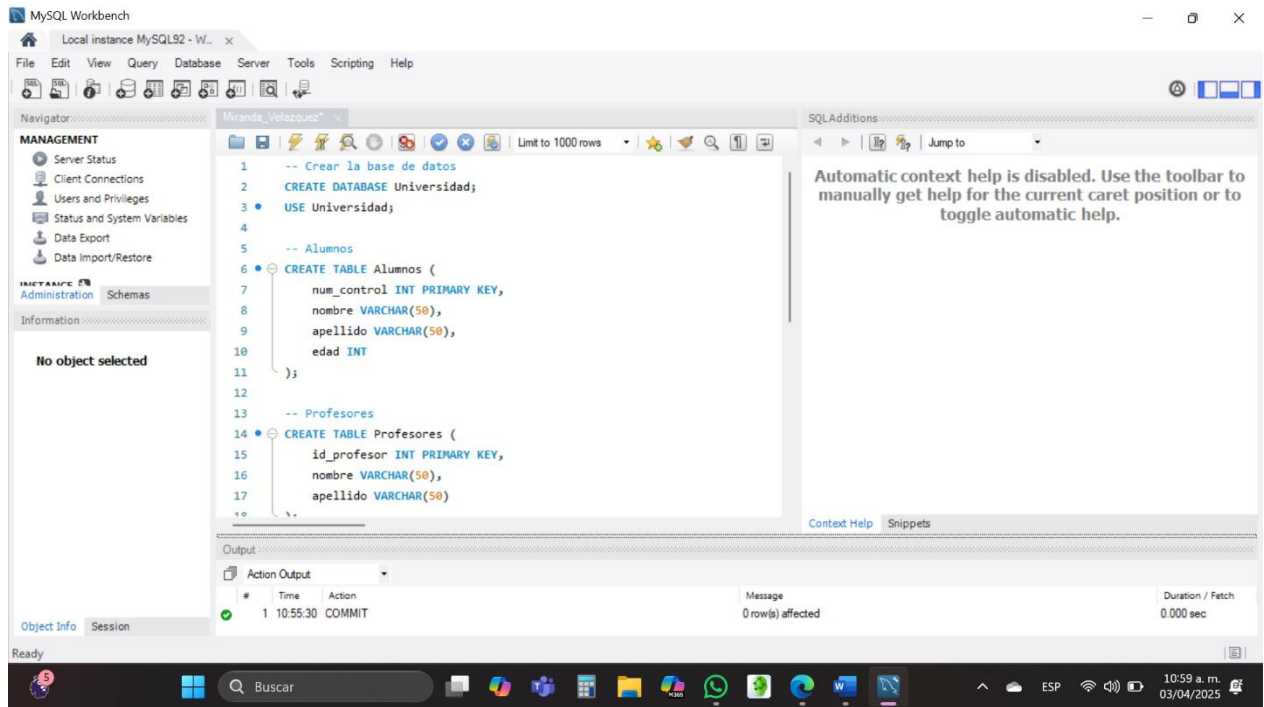


El diagrama representa un modelo simplificado de base de datos, compuesto por tres entidades principales:

1. **Alumnos:** Contiene los datos básicos de cada estudiante, como número de control (clave primaria), nombre, apellido y edad.
2. **Asignaturas:** Representa las materias disponibles en la universidad. Cada asignatura tiene un ID único (clave primaria), nombre, número de créditos, un campo que identifica al profesor encargado (id\_profesor) y el área a la que pertenece.
3. **Estudia:** Es una tabla intermedia que modela la relación muchos-a-muchos entre los alumnos y las asignaturas. Cada registro indica que un alumno cursa una asignatura específica. Su clave primaria está compuesta por las dos llaves foráneas: num\_control (de Alumnos) e id\_asignatura (de Asignaturas).



- c) Genera el *script.sql* y ejecutalo para la creación de las tablas. Toma capturas de pantalla de tu proceso.



- d) Realiza una consulta a tu base de datos para listar las tablas y validar que se hayan creado correctamente, para ello puedes utilizar los siguientes comandos:

**USE 'nombre\_database';**

**SHOW TABLES;**

The screenshot shows a SQL IDE interface with a dark theme. The top toolbar includes icons for file operations and a 'Limit to 1000 rows' dropdown. The left sidebar shows a tree view of the database schema, with 'Universidad' expanded to show 'Tables' containing 'Alumnos', 'Asignaturas', 'estudia', and 'Profesores'. The main editor displays a SQL script with the following commands:

```

1 -- Crear la base de datos
2 CREATE DATABASE Universidad;
3 USE Universidad;
4
5 -- Alumnos
6 CREATE TABLE Alumnos (
7
8
9
10
11
12

```

Below the editor, the 'Result Grid' shows the output of the 'SHOW TABLES' command. The table lists the tables in the 'Universidad' database:

Tables_in_universid...
Alumnos
Asignaturas
estudia
Profesores

At the bottom, the 'Object Info' panel shows details for the 'estudia' table, including its columns and primary keys:

Column	PK
num_control	int PK
id_asignatura	int PK

The 'Action Output' panel shows the execution log of the commands:

Time	Action	Response	Duration / Fetch Time
21:08:32	USE Universidad	0 row(s) affected	0.00072 sec
21:08:52	SHOW TABLES	4 row(s) returned	0.0029 sec / 0.00000...
21:08:52	SHOW TABLES	4 row(s) returned	0.0017 sec / 0.00000...
21:08:56	SHOW TABLES	4 row(s) returned	0.0021 sec / 0.00000...

The status bar at the bottom indicates 'Query Completed'.



- e) Utiliza el comando DESCRIBE para cada una de las tablas para validar que se crearon de acuerdo a tu diagrama

The screenshot shows the SQL Developer interface with the following components:

- Left Panel (SCHEMAS):** Displays the database structure, including the 'Universidad' schema and its tables: 'Alumnos', 'Asignaturas', 'estudia', 'Profesores', 'Views', 'Stored Procedures', and 'Functions'.
- SQL Editor:** Contains the following SQL commands:
 

```

      1 -- Crear la base de datos
      2 CREATE DATABASE Universidad;
      3 USE Universidad;
      4
      5 DESCRIBE Alumnos;
      6 -- Alumnos
      7 CREATE TABLE Alumnos (
      8     num_control INT PRIMARY KEY,
      9     nombre VARCHAR(50),
      10    apellido VARCHAR(50),
      11    edad INT
      
```
- Result Grid:** Displays the results of the DESCRIBE command for the 'Alumnos' table. The table has the following structure:
 

Field	Type	Null	Key	Default	Extra
num_control	int	NO	PRI		
nombre	varchar(50)	YES			
apellido	varchar(50)	YES			
edad	int	YES			
- Object Info:** Shows the table 'estudia' with columns 'num\_control' (int PK) and 'id\_asignatura' (int PK).
- Action Output:** Displays the execution results of the SQL commands. The table shows the following actions:
 

Time	Action	Response	Duration / Fetch Time
21:08:32	USE Universidad	0 row(s) affected	0.00072 sec
21:08:52	SHOW TABLES	4 row(s) returned	0.0029 sec / 0.00000...
21:08:52	SHOW TABLES	4 row(s) returned	0.0017 sec / 0.00000...
21:08:56	SHOW TABLES	4 row(s) returned	0.0021 sec / 0.00000...
21:08:56	SHOW TABLES	4 row(s) returned	0.0021 sec / 0.00000...
23:19:00	DESCRIBE Alumnos	4 row(s) returned	0.066 sec / 0.00092...
23:19:34	DESCRIBE Profesores	3 row(s) returned	0.0044 sec / 0.00001...



Local Instance 3306 - Warning - not supported

Administration Schemas Query 1 SQL File 3\*

SCHEMAS

Filter objects

- Jose\_Jauregui
- sys
- Universidad
  - Tables
    - Alumnos
    - Asignaturas
    - estudia
    - Profesores
    - Views
    - Stored Procedures
    - Functions

```

21
22 DESCRIBE Asignaturas;
23 -- Asignaturas
24 CREATE TABLE Asignaturas (
25   id_asignatura INT PRIMARY KEY,
26   nombre VARCHAR(100),
27   num_credits INT,
28   id_profesor INT,
29   area VARCHAR(50),
30   FOREIGN KEY (id_profesor) REFERENCES Profesores(id_profesor) ON DELETE SET NULL
31 );
32

```

100% 23:27

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
id_asignatura	int	NO	PRI		
nombre	varchar(100)	YES			
num_credits	int	YES			
id_profesor	int	YES	MUL		
area	varchar(50)	YES			

Result 6 Read Only

Object Info Session

Table: **estudia**

Columns:

- num\_control int PK
- id\_asignatura int PK

Action Output

	Time	Action	Response	Duration / Fetch Time
1	21:08:32	USE Universidad	0 row(s) affected	0.00072 sec
2	21:08:52	SHOW TABLES	4 row(s) returned	0.0029 sec / 0.00000...
3	21:08:52	SHOW TABLES	4 row(s) returned	0.0017 sec / 0.00000...
4	21:08:56	SHOW TABLES	4 row(s) returned	0.0021 sec / 0.00000...
5	23:19:00	DESCRIBE Alumnos	4 row(s) returned	0.066 sec / 0.00092...
6	23:19:34	DESCRIBE Profesores	3 row(s) returned	0.0044 sec / 0.00001...
7	23:20:06	DESCRIBE Asignaturas	5 row(s) returned	0.0032 sec / 0.00001...

Query Completed

Local Instance 3306 - Warning - not supported

Administration Schemas Query 1 SQL File 3\*

SCHEMAS

Filter objects

- Jose\_Jauregui
- sys
- Universidad
  - Tables
    - Alumnos
    - Asignaturas
    - estudia
    - Profesores
    - Views
    - Stored Procedures
    - Functions

```

30 FOREIGN KEY (id_profesor) REFERENCES Profesores(id_profesor) ON DELETE SET NULL
31 );
32
33 DESCRIBE Estudia;
34 CREATE TABLE Estudia (
35   num_control INT,
36   id_asignatura INT,
37   PRIMARY KEY (num_control, id_asignatura),
38   FOREIGN KEY (num_control) REFERENCES Alumnos(num_control) ON DELETE CASCADE,
39   FOREIGN KEY (id_asignatura) REFERENCES Asignaturas(id_asignatura) ON DELETE CASCADE
40 );
41 SHOW TABLES;
42
43

```

100% 19:33

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
num_control	int	NO	PRI		
id_asignatura	int	NO	PRI		

Result 7 Result 8 Read Only

Object Info Session

Table: **estudia**

Columns:

- num\_control int PK
- id\_asignatura int PK

Action Output

	Time	Action	Response	Duration / Fetch Time
1	21:08:32	USE Universidad	0 row(s) affected	0.00072 sec
2	21:08:52	SHOW TABLES	4 row(s) returned	0.0029 sec / 0.00000...
3	21:08:52	SHOW TABLES	4 row(s) returned	0.0017 sec / 0.00000...
4	21:08:56	SHOW TABLES	4 row(s) returned	0.0021 sec / 0.00000...
5	23:19:00	DESCRIBE Alumnos	4 row(s) returned	0.066 sec / 0.00092...
6	23:19:34	DESCRIBE Profesores	3 row(s) returned	0.0044 sec / 0.00001...
7	23:20:06	DESCRIBE Asignaturas	5 row(s) returned	0.0032 sec / 0.00001...
8	23:20:49	DESCRIBE Estudia	2 row(s) returned	0.0032 sec / 0.00001...

Query Completed





Si tienes algún problema con los comandos, investiga un poco sobre ellos para poder obtener el listado de tablas, en el **Tutorial Básico de MySQL** que se muestra en los enlaces puedes encontrar información sobre los comandos, puedes complementar con la revisión del documento **MySQL 8.0 Reference Manual**.

MySQL, (2020). *MySQL 8.0 Reference Manual* [Sitio Web] Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/information-schema.html>

MySQL AB (s.f.). *Tutorial Básico MySQL* [Archivo PDF]. Recuperado de <http://www.v-espino.com/~chema/daw1/tutoriales/tutorial-mysql.pdf>

4. Desarrolla un reporte en el que incluyas los siguientes puntos:

- Planteamiento: Explica la funcionalidad de tu base de datos.
- Diagrama relacional: incluye la imagen del diagrama generado el Lucidchart.
- Procedimiento: Documenta tu procedimiento de creación de tablas y consultas realizadas, toma capturas de pantalla para validar la correcta ejecución de tu *scripts SQL*.

5. Redacta una conclusión en la que describas la utilidad de las llaves primarias y foráneas en la base de datos.

Las llaves primarias y foráneas son fundamentales para el diseño y funcionamiento correcto de una base de datos relacional. La llave primaria permite identificar de manera única cada registro dentro de una tabla, lo que garantiza la integridad y evita duplicidades. Gracias a ella, es posible realizar búsquedas eficientes y mantener la coherencia en los datos almacenados.

Por otro lado, las llaves foráneas permiten establecer relaciones entre distintas tablas, asegurando que los datos estén conectados de forma lógica y estructurada. Esto facilita la organización, evita la redundancia de información y permite que la base de datos sea escalable y más fácil de mantener. En conjunto, ambas llaves permiten construir un modelo relacional sólido y confiable.

6. Elabora el reporte en este mismo documento.

7. Genera una carpeta .ZIP donde integres tu reporte y tu *script.sql* que contenga todo el código desarrollado.

8. Al finalizar esta actividad, vuelve a la plataforma y sigue los pasos que se indican para enviar tu trabajo.

\* \* \*



## Conclusión

### Miranda Velázquez Mariles:

La creación de una base de datos para una Universidad permite organizar de manera eficiente la información relacionada con estudiantes, profesores, materias y sus relaciones, mediante el uso de llaves primarias y foráneas es posible establecer vínculos lógicos entre las distintas entidades asegurando la integridad de los datos y facilitando consultas complejas y reportes útiles para la administración académica.

Las llaves primarias identifican de forma única a cada registro dentro de una tabla, evitando duplicidades y permitiendo un acceso rápido y preciso, por otro lado, las llaves foráneas permiten relacionar registros entre tablas creando una estructura de datos relacional que mantiene la consistencia de la información al garantizar que los valores referenciados existan previamente en la tabla principal.

### Jose Emiliano Jauregui Guzman:

Para esta actividad pudimos tener práctica de la creación de una base de datos ya en MySQL y poder explorar las llaves primarias y las llaves foráneas que nos permiten tener un registro preciso y poder identificar con mayor exactitud la información ya que estas llaves nos indican su registro único y la foránea nos ayuda a relacionar las tablas para darle mas cuerpo a nuestra base de datos al momento de meterlos en un contexto.

Me gusto poder tener mucha guía al momento de crear cada linea de código ya que tuvimos bastantes fuentes que consultar y con ayuda del profesor especificando las sentencias y las cláusulas para poder verificar las tablas creadas, con esta actividad me quedo satisfecto por el conocimiento que otorgo.

